Support Operations

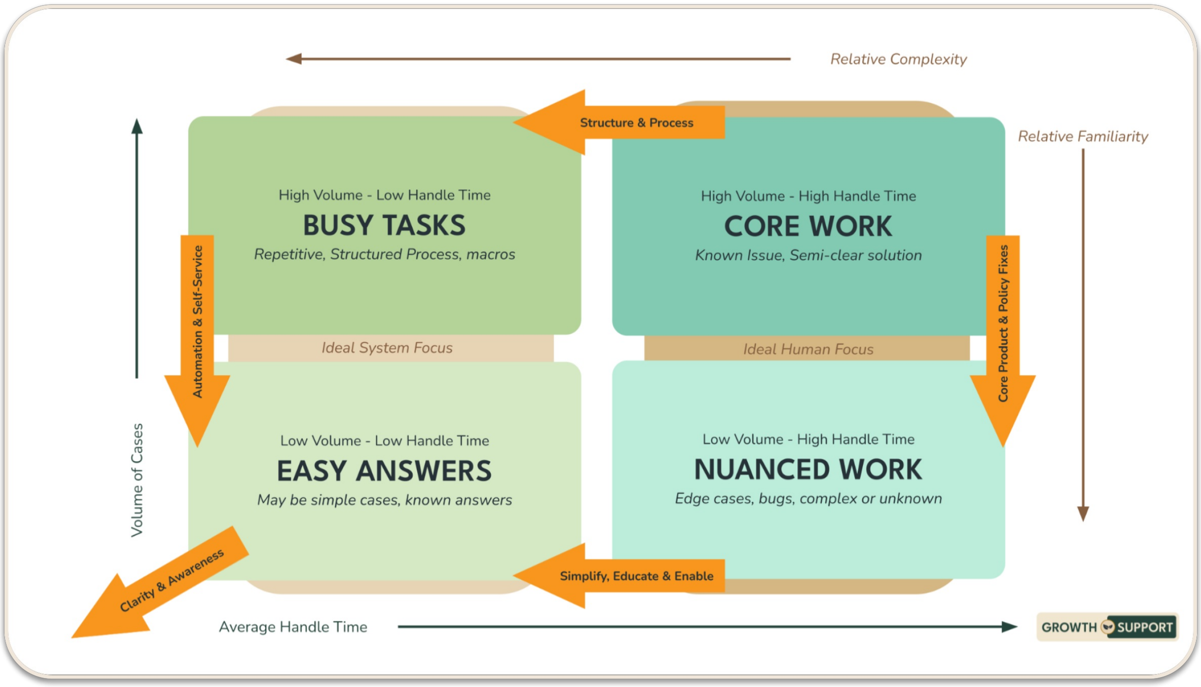
04/10/2024 12:21 am EDT

Help make your support more effective, and deliver better service to customers.

Navigating the Support Landscape

What is the 'Time-Spent' Method?

The 'Time-Spent' Method for Support Operations is a way to map where the time support efforts are spent and then reduce friction, automate manual work, and add value methodically and systematically to scale effectively.

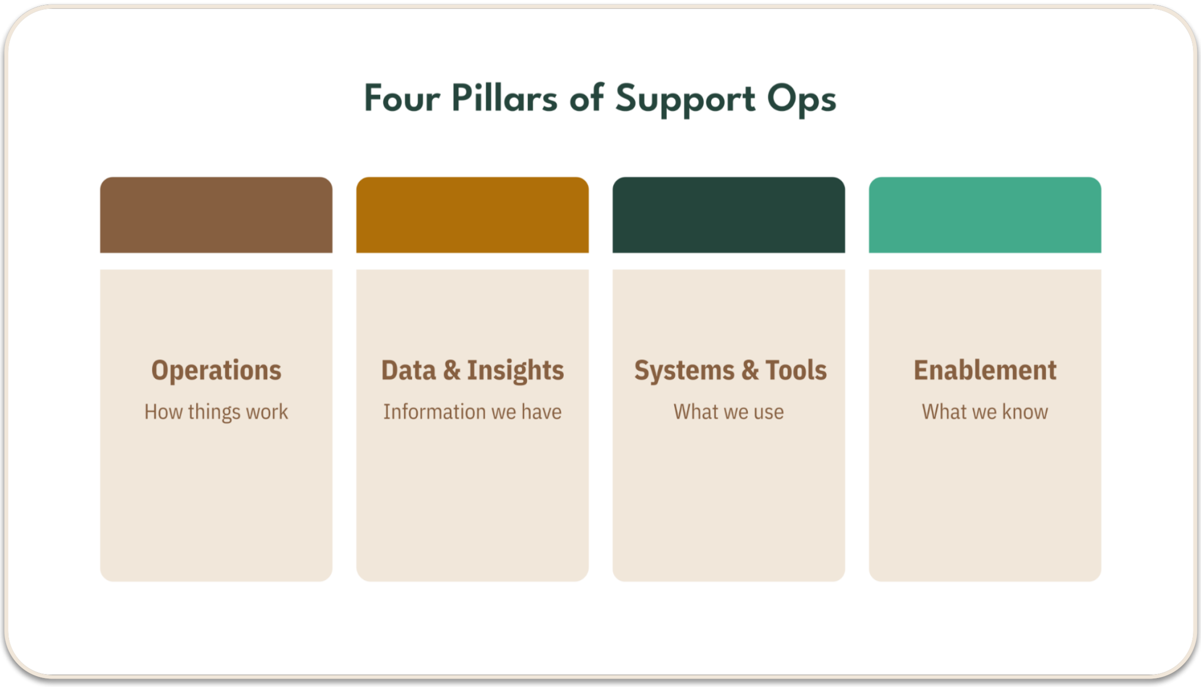

Relative Complexity
Relative Familiarity
Structure & Process
High Volume - Low Handle Time
BUSY TASKS
Repetitive, Structured Process, macros
High Volume - High Handle Time
CORE WORK
Known Issue, Semi-clear solution
Automation & Self-Service
Core Product & Policy Fixes
Ideal System Focus
Ideal Human Focus
Low Volume - Low Handle Time
EASY ANSWERS
May be simple cases, known answers
Low Volume - High Handle Time
NUANCED WORK
Edge cases, bugs, complex or unknown
Volume of Cases
Clarity & Awareness
Simplify, Educate & Enable
Average Handle Time
GROWTH SUPPORT

To apply this method, we will first need to understand support operations and the four pillars we will apply here.

Support Operations

Support Operations is responsible for helping the support team be more productive, work more effectively, and deliver better service to customers. It's all about supporting the supporters, reducing friction, and adding value across the customer journey.

The four pillars of Support Operations



1. **Operations** focuses on the processes and methods used to carry out tasks efficiently, ensuring a streamlined workflow for the support team. *How the work is done.*

2. **Data & Insights** involves leveraging information to gain valuable insights into customer needs and support performance, aiding informed decision-making towards actionable change. *What information we have about the work.*

3. **Tooling** focuses on the technology used to support the customer journey, aiming to optimize the team's capabilities through effective implementation of tools, systems and automation. *What we use to do the work.*

4. **Enablement** refers to the knowledge and skills of the support team, emphasizing training and development to ensure they possess the necessary skills to deliver high-quality and support. As well as stay informed about changes in product and process. *Our knowledge about the work itself.*

The Support Landscape

The Support Landscape is the essence of what your support team is doing. What are they spending their time on? How much time? What areas of the product or subjects are those conversations or cases about?

For the purpose of this method, the landscape is comprised of four components:

1. The categorization and identification of ticket data. (A tag taxonomy)

2. Volume of Cases (per type within this taxonomy)

3. Average Handle Time (AHT) of Cases (per type within this taxonomy)

4. Total Handle Time (THT) of Cases (per type within this taxonomy)

Mapping your Support Landscape using Time-Spent

1. Define your ticket identification & categorization methods. Have a Taxonomy in place.
   (It can be flat or hierarchical, but I would recommend a two or three-layer taxonomy for richer data)

2. Calculate your Total Time Spent Handling each type and your Average Handle Time (AHT) per type.

   - Total Handle Time = Sum of all cases' Handle Time (within type)

   - Average Handle Time = Sum of all Handle Time per Type / Number of Cases

3. Using these three Metrics, create a Bubble Chart (for Visual)

4. ID will be Type

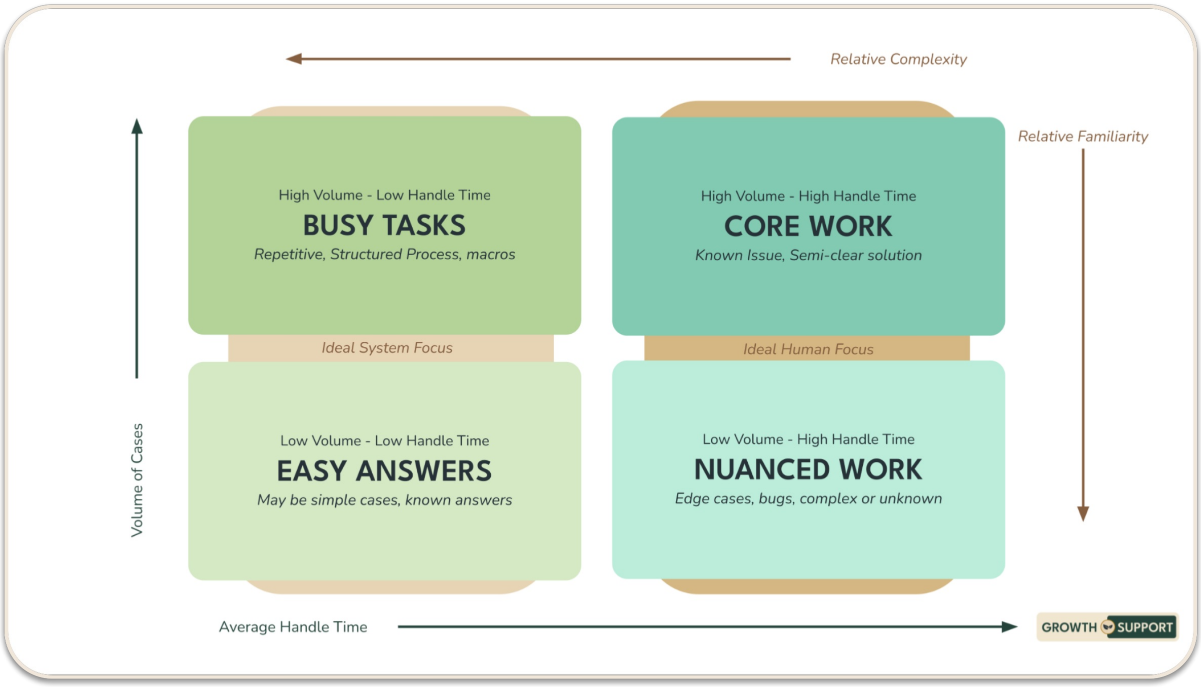
5. The series will be: Overlying Area

6. The horizontal (X) axis will be: Average Handle Time

7. The vertical (Y) Axis will be: Volume of Cases

   - Bubble size will be Total Handle Time (THT)

When you visualize and map your Support Landscape, we'll be looking at the distribution of cases, and their relevant clustering when compared to the medians of both volume of cases and Average Handle Time (AHT)


Relative Complexity
Relative Familiarity
High Volume - Low Handle Time
BUSY TASKS
Repetitive, Structured Process, macros
High Volume - High Handle Time
CORE WORK
Known Issue, Semi-clear solution
Ideal System Focus
Ideal Human Focus
Low Volume - Low Handle Time
EASY ANSWERS
May be simple cases, known answers
Low Volume - High Handle Time
NUANCED WORK
Edge cases, bugs, complex or unknown
Volume of Cases
Average Handle Time
GROWTH SUPPORT

The Support landscape is comprised of four areas:

Top Left: Busy Tasks High Volume, Low Handle Time - Repetitive, Administrative, structured process, administrative.	Top Right: Core Work High Volume, High Handle Time - Known issue areas, fairly unstructured, Team needs time to tackle
Bottom left: Easy Answers Low Volume, Low Handle Time - Quick questions, Informational, information is known well.	Bottom Right: Nuanced Work Low Volume, High Handle Time - Unstructured, unknown, or heavy process. Edge cases or bugs.
Ideally: Systems handle low handle-time requests.	Ideally: Human focus is on complex work

If your landscape is clustered together, you'll need to benchmark and define what a "healthy ticket" looks like for you, and then you can focus on overall improvements and goal setting to improve.

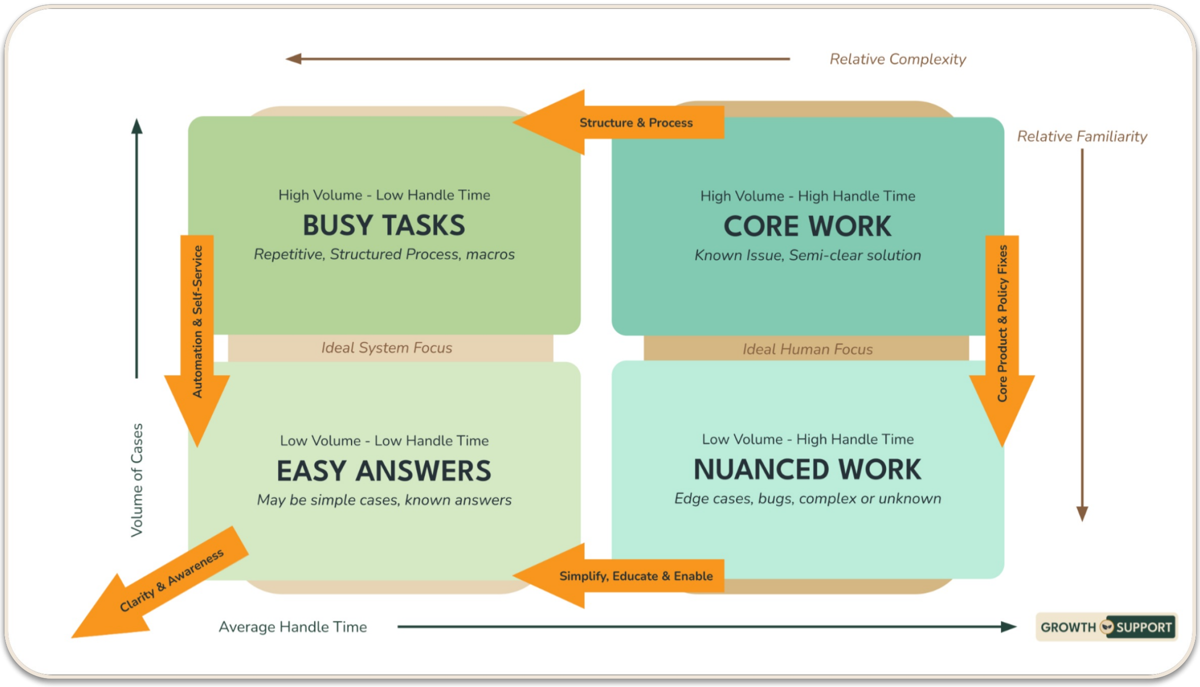
If you are already in your healthy range, this is good news! You'll never be able to reduce everything to zero, but you can identify larger opportunities and goals based on overall type distributions.

If you have large variations or outliers, or a large dispersion from the median, focus on prioritizing heavy time spenders first,

Navigating the Support Landscape

Shape your landscape by applying the four pillars of Support Operations.

The ideal goal would be to get cases from Core Work to Easy Answers, but bear in mind that this is relative to what you define as a "healthy ticket" per type - looking at the overlying areas of each type of request could also inform you of what product areas you will need overall support in for the team in terms of skills or tools.


Relative Complexity
Relative Familiarity
Structure & Process
High Volume - Low Handle Time
BUSY TASKS
Repetitive, Structured Process, macros
High Volume - High Handle Time
CORE WORK
Known Issue, Semi-clear solution
Automation & Self-Service
Core Product & Policy Fixes
Ideal System Focus
Ideal Human Focus
Low Volume - Low Handle Time
EASY ANSWERS
May be simple cases, known answers
Low Volume - High Handle Time
NUANCED WORK
Edge cases, bugs, complex or unknown
Volume of Cases
Clarity & Awareness
Simplify, Educate & Enable
Average Handle Time
GROWTH SUPPORT

When working to move from Core to Nuanced focus on Core Product & Policy Fixes. (Data & Insights)

Use Data & Insights to become the Voice of the Customer for your Product & Engineering Teams by bringing full usage data and recommendations to better meet customer needs. You can use the same approach to recommend any potential policy adjustments to create less friction for customers.

- "It might still take us a while to deal with it, but we've mitigated incoming volume and we get much less"
- "This issue is fixed, and now the cases we do get are actually the complex, bugs, and edge cases."

When working to move from Nuanced Work to Easy Tasks, focus on Simplifying, Educating, & Enabling. (Enablement)

If cases are taking the team a long time to handle, you can work to better equip your team with the knowledge and skills to investigate and execute more effectively. Training and enablement can help to provide a better experience for both customers and agents.

- "We've been trained and enabled on how to better navigate edge cases and better deal with these types of issues. Investigation & Decisions take less time, or we no longer need to escalate to other team members"

When working to move from Core Work to Busy Tasks focus on Structure & Process (Operations)

By better optimizing operations, you can support the team in handling more complex issues with clarity, cleaner processes, or clearer documentation. By structuring unstructured issues, you can make the solutions repeatable and more standard and reduce the time spent researching how to resolve an issue. Reducing the steps needed to process a specific solution will also go a long way in these improvements.

- "We've set clear steps and solutions to solve these issues. Clear documentation, or we've simplified the manual work needed to solve. Now we just have to take action or use the macro."

When working to move from Busy Tasks to Easy Answers focus on Automation & Self-service. (Systems & Tooling)

If issues are repeatable and consistently the same exact steps each time, you can look to better utilize your systems, automation, or self-service. You can mitigate the processing by providing the answers directly to customers, automate the steps needed to take action, or implement interactive self-service. This will help reduce the 'administrative' workload of the support team, and help them to focus efforts on more complex cases.

- "We've enabled our customers to help themselves solve this issue, automated all of the manual steps, or are helping customers through an automated service like a chatbot. Now the questions that we get are for additional information, clarification, or advice."

Eliminating Easy Answers - Clarity & Awareness

If you have questions that are about information that is already available, or in addition to self-service material that already exists, it may be time to look at raising awareness about the information, or clarifying the language. This can help to remove general frequent questions and take cases off the landscape entirely.

- "We've made the information more readily available and clearer for our customers, and they are fully educated and aware of how to use the product and services fully."

Driving Outcomes With the 'Time-Spent' Method

**Disclaimer: The key thing to note is that this works when taking a 'healthy approach' to the method. You can never reduce everything to zero, and you shouldn't aim to drill agents on handle times simply based on this method.*

Customer Experience Outcomes

When you take a healthy approach to navigating your Support Landscape, you are not only scaling your team effectively. You are also clarifying and simplifying your customer's experience to reduce friction and add value across the entire customer lifecycle. Making it faster and easier for them to use your product, get the information that they need, and find the value that they are looking for.

Agent Experience Outcomes

By freeing up the time of agents from Busy Tasks, or Enabling them through upskilling and development, you empower them to become more effective in their roles, and focus their efforts on more complex cases, and less tedious tasks. Keeping the agent experience top-of-mind is always important when supporting the supporters and building a more engaged team.

Staffing Outcomes

When utilizing this method effectively, you will be able to reduce friction, automate manual work, and add value. By doing this in a methodical and systematic way you will be able to scale more efficiently. This will help you to go further with the same, more effective team. With this you may not need to add additional members as soon as you may have needed to without this view.

Financial Outcomes

To take this method further than driving operational outcomes, you can also utilize this method to reflect financial outcomes with the business directly. How? Time is money.

You can measure the cost of running Support per hour/minute and map this to the cases that you have in each quadrant. This shows you how much it costs to run a process, or handle a specific case-type. You may find that your team spends a considerable amount of this spend on a specific busy task or area, which in turn, costs the business.

To drive the business case for change or reflect the outcomes of improvements you have made, you can also utilize the potential cost savings of resolutions.

A statement for this could be:

“*Currently this ‘Busy Task’ is costing us __ in time which equates to __ in revenue per month. By implementing this solution and reducing the volume by __ we would effectively save __.”*

These are just the initial outcomes after creating the methodology, thought there are many more that can be found! As this evolves, so will the view and depth of understanding of the Support Landscape.

Creating Customer Service Policies that Empower Reps

As a company grows, there will have more and more scenarios in which they need to standardize our customer-facing policies and how they approach providing exceptions to those policies.

In this article, I'll go through some guiding principles for: who to create policies for, levels of exceptions and when customers meet those levels, and what decision-making power each individual has versus requiring approvals to make different exceptions.

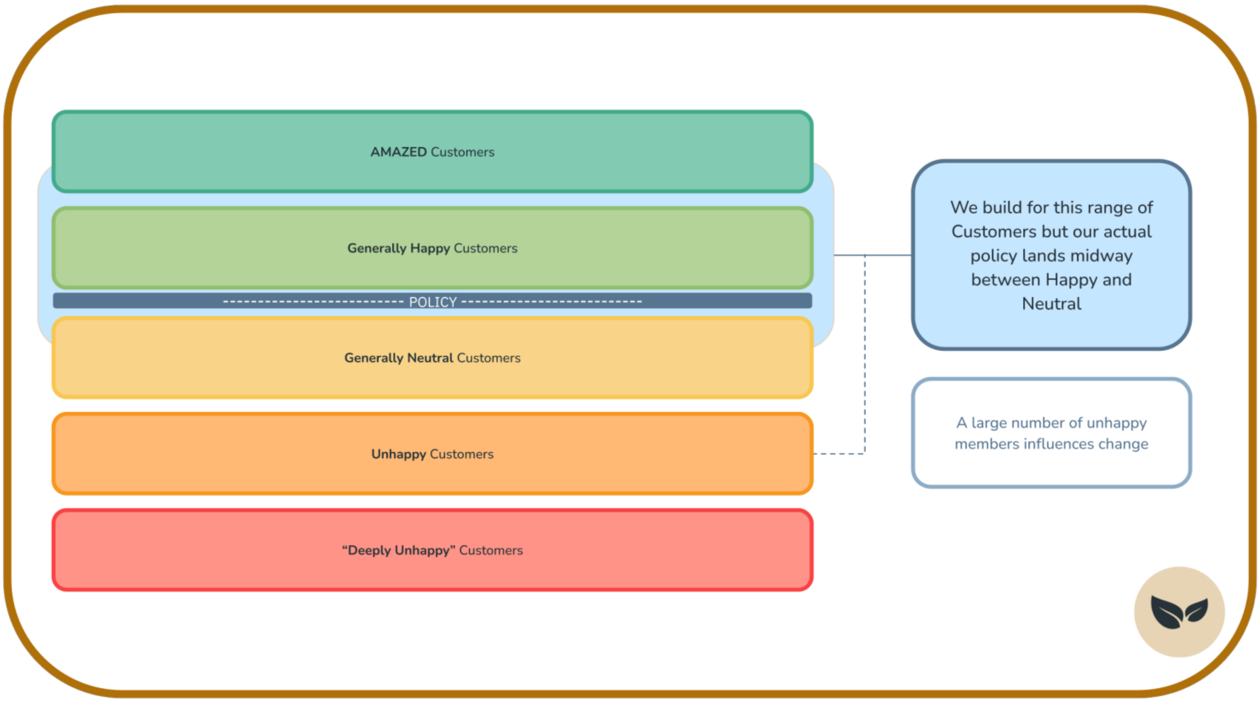
> It is our responsibility as service teams to be able to have constructive and positive communication. To be able to manage customer expectations and utilize the resources that we have to find solutions within policy and reason to de-escalate customer situations to create an amazing experience. We should not only rely on providing exceptions to create a memorable and great experience!

Who do we build policies for?

When building service policies, both sides need to be considered: our customers and our business.

We want to create the best possible experience for our customers, ensuring they have flexibility and positivity while ensuring that all customers get the same level of support from our team when needed.

For this reason, we build our policies for the largest population of our customer sentiment We create policies to make an amazing experience, but the actual policy will land between our 'generally happy' and 'generally neutral' customer sentiments. *(image below)*



AMAZED Customers

Generally Happy Customers

------------------------- POLICY -------------------------

Generally Neutral Customers

Unhappy Customers

"Deeply Unhappy" Customers

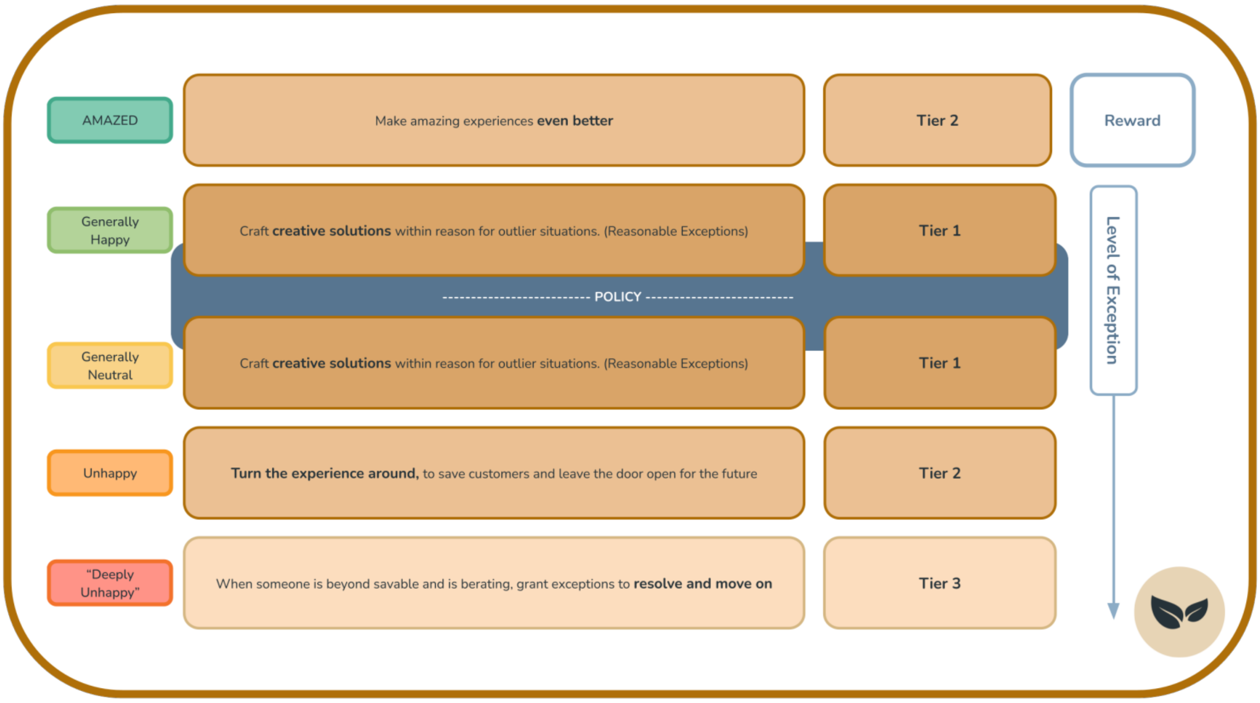
We build for this range of Customers but our actual policy lands midway between Happy and Neutral

A large number of unhappy members influences change

Customer Sentiments will be important when providing exceptions to policies.

What are the levels of Exception?

Every customer situation is unique, and while we need to standardize policies, we still want to make room to help customers with their situation so that we can make the best experience possible. For this reason, we have varying levels of exceptions based on the customer's situation, criteria, and experience/sentiment.



AMAZED	Make amazing experiences **even better**	Tier 2	Reward
Generally Happy	Craft **creative solutions** within reason for outlier situations. (Reasonable Exceptions)	Tier 1	
	------------------------- POLICY -------------------------		
Generally Neutral	Craft **creative solutions** within reason for outlier situations. (Reasonable Exceptions)	Tier 1	
Unhappy	**Turn the experience around,** to save customers and leave the door open for the future	Tier 2	
"Deeply Unhappy"	When someone is beyond savable and is berating, grant exceptions to **resolve and move on**	Tier 3	

Tier-One

These are creative and middle-ground solutions. Exceptions that are not very far off from the policy or a workaround/alternative solution that is still within policy.

Tier-Two

There are two types of tier-two exceptions: **Rewards and Exceptions.** These are clear exceptions to the policy used to turn the experience around for customers we can save and turn things around for. Or at least leave the door open for the future.

For Tier-two exceptions granted to 'Amazed' customers, these are considered rewards for being such a promoter and ambassador. *These cases are much rarer in which we will grant them,* but we should **look for moments that matter.**

Why would we give already amazed customers exceptions? They are already amazed...

To create an equal opportunity to make the experience EVEN better than it already is. *(Here's a funny movie clip with an analogy for this concept)* - why treat unhappy customers better than your superstar customers?

Tier-Three

This one goes against the concept above because these customers would be beyond savable, or their experience is not aligned with the usage/product guidelines. **The purpose of Tier 3 exceptions is to completely and fully resolve the issue no matter what.**

> Once a customer escalates from the 'Unhappy' to 'Deeply Unhappy' sentiment, and we make the decision that their frustration has reached a point in which we could lose the customer, we will move to what we consider a 'one-touch solve.' This means we would do everything we can to resolve the issue in one response rather than asking for alternative options or pushing back on policy to avoid back-and-forth.

What rights do reps have?

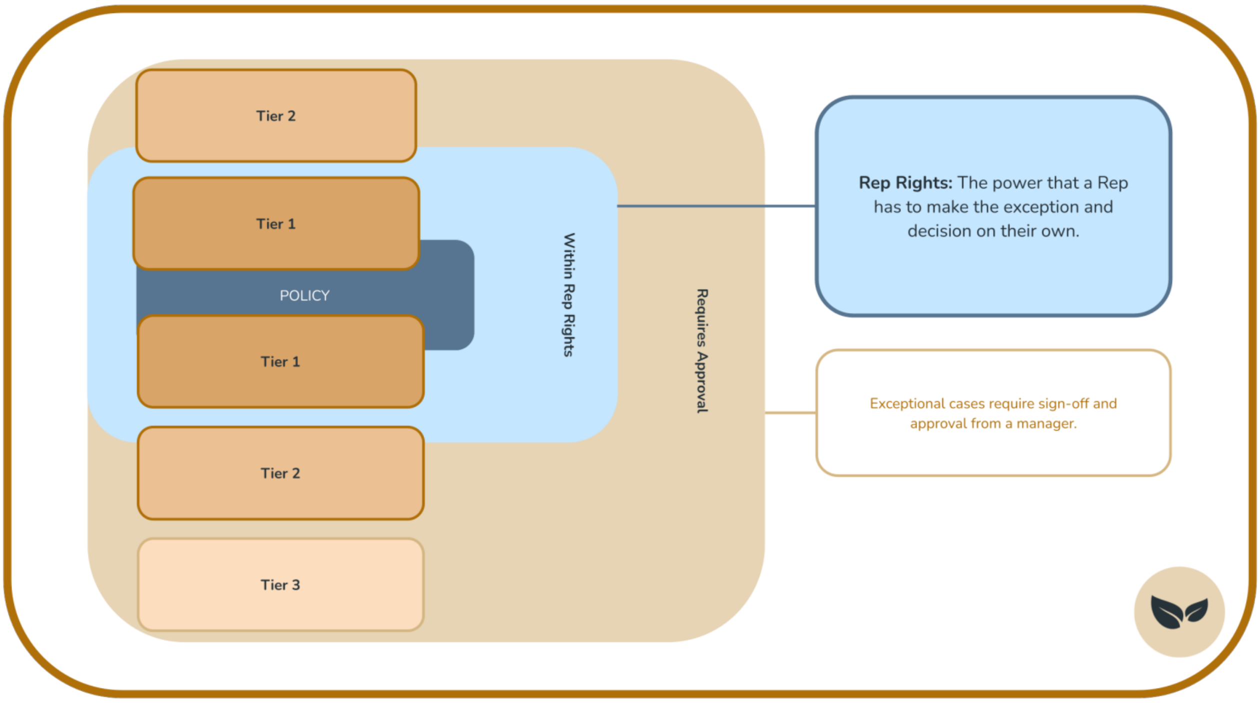
For exceptions to policy, we want to **empower reps to help customers** in their unique situations without a heavy process and slow down in approvals.

> For this reason, we give Reps full control over Tier-one exceptions without requesting approval.

Depending on the policy itself, they also have the power to grant Tier-Two exceptions without approval. These **Tier-Two level rep-rights are given on a per-policy basis**, not a blanket right.

Tiers of Exception

- Tier-One - Full rights to give exception
- Tier-Two - Mixed: Determined and given on a per-policy basis
- Tier-Three: Always requires approval



Tier 2
Tier 1
POLICY
Tier 1
Tier 2
Tier 3
Within Rep Rights
Requires Approval
Rep Rights: The power that a Rep has to make the exception and decision on their own.
Exceptional cases require sign-off and approval from a manager.

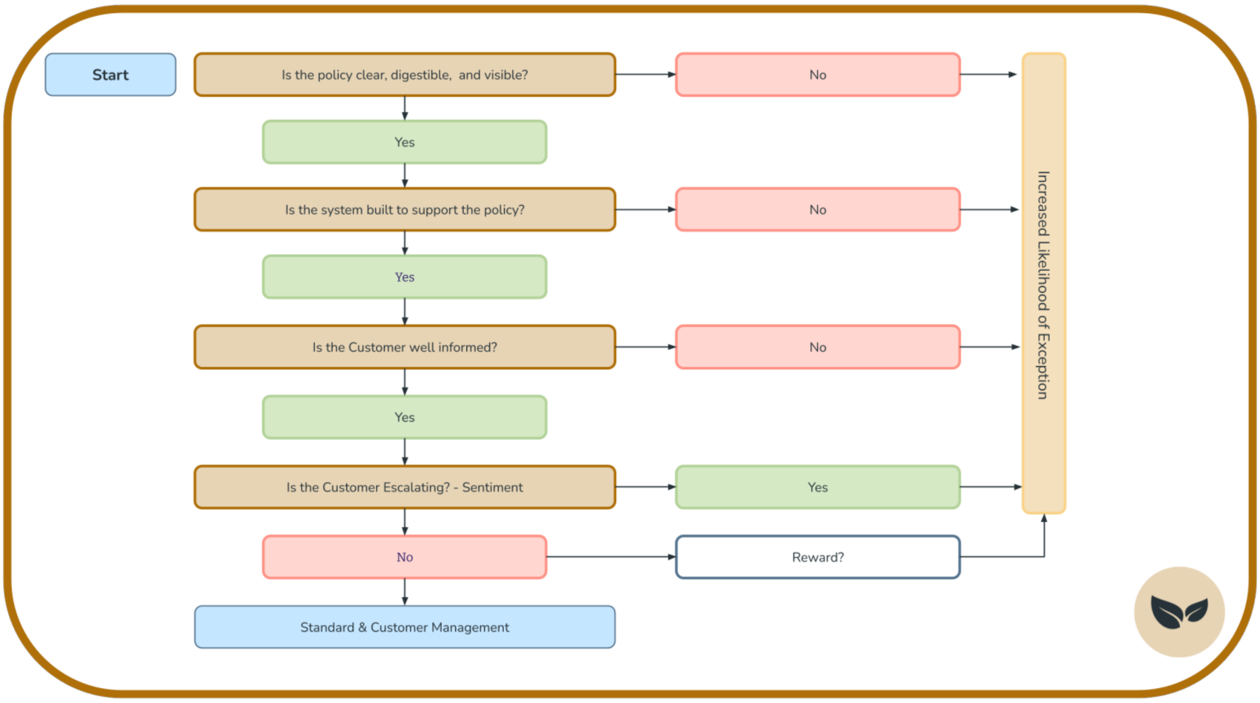
> We should add and consider these tiers and rights throughout the policy creation or policy adjustment process.

When can we grant exceptions?

When we create or adjust policy, we create the tiers and levels for exception. **During this time, we should also create the criteria needed to reach each level as a customer.** This is easily made in the form of a decision tree that reps can use to determine which level of exception is appropriate quickly and/or if they should move to get approval.

As an overlying decision tree, **we should ensure that policies are clear, simple, and visible to customers** (internal or external). If the policy is not clear and understandable, we are more likely to move towards exceptions.

Here is a basic overlying decision tree to see how we should move towards considering an exception agnostic of a policy:


Start
Is the policy clear, digestible, and visible?
No
Yes
Is the system built to support the policy?
No
Yes
Is the Customer well informed?
No
Yes
Is the Customer Escalating? - Sentiment
Yes
No
Reward?
Standard & Customer Management
Increased Likelihood of Exception

How do we communicate exceptions?

When providing exceptions, it is easy to set the wrong tone that these levels of exceptions will always be available to customers in every situation. This is why it is important to set the right expectation that this exception is for the specific scenario, and set the tone for what to expect in the future.

The standard format for delivering exceptions is E-E-E-E-E (Pronounced: ▯▯▯)

- **Empathy** - ALWAYS begin with empathy & understanding for the situation.
- **Explanation** - Explain the reasoning behind what caused a situation / the normal scenario.
- **Exception** - Provide the exception for their specific case.
- **Expectation** - Communicate what to expect for this exception and in the future. (What happens next)
- **Empathy** - End on a good note, and leave the door open. Reinforce that we understand.

How will we update exception tiers or policy in the future?

Through Policy Analysis!

Track and report on exceptions-per-problem type to see if there is a need to update , training, exceptions tiers, or ultimately the policy. This will be a consideration of customer and business needs as policies, again, should serve *both* sides, and ultimately be the best for the business.

In Closing

Policies are important for both customers, and the business alike, but we don't want them to get in the way of the customer experience, and cause friction to the value that a customer is trying to receive. Or the experience that the Rep is trying to create. Polices should grow with the business, and allow Reps to take ownership over creating an experience that lasts.