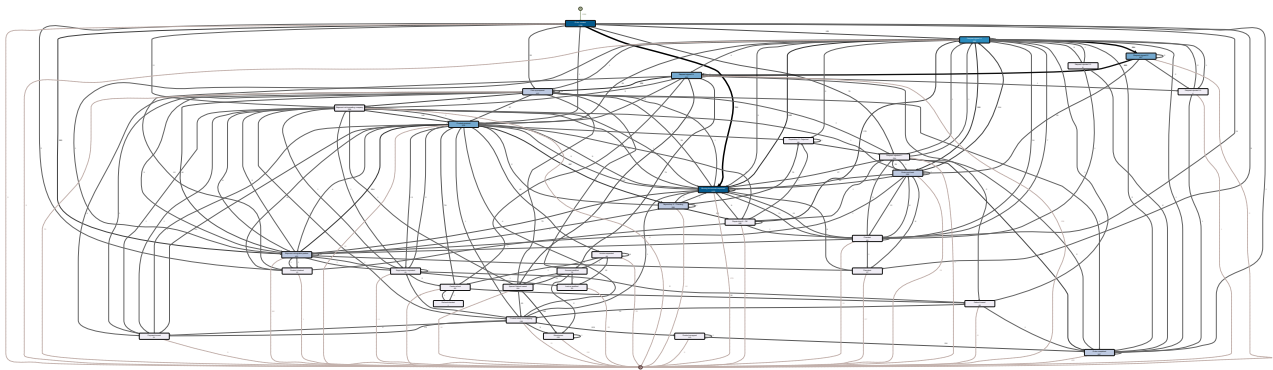


Managing Complexity in Process Mining

Have you ever imported a data set in your process mining tool and what you got was a complex “spaghetti” process? Often, real-life processes are so complex that the resulting process maps are too complicated to interpret and use.

For example, the process that you get might look like this:



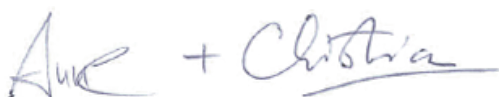
The problem with this picture is not that it is wrong, in fact this *is* the true process if you look at it in its entirety. The problem is that this process map is not useful, because it is too complicated to derive any useful insights or actionable information from it.

What we need to do is to break this up and to simplify the process map to get more manageable pieces.

In this article, you will learn nine simplification strategies for complex process maps that will help you get the analysis results that you need. We show you how you can apply these strategies in the process mining software [Disco](#) (download the [free demo version from the Disco website](#) to follow along with the instructions).

Let's get started!

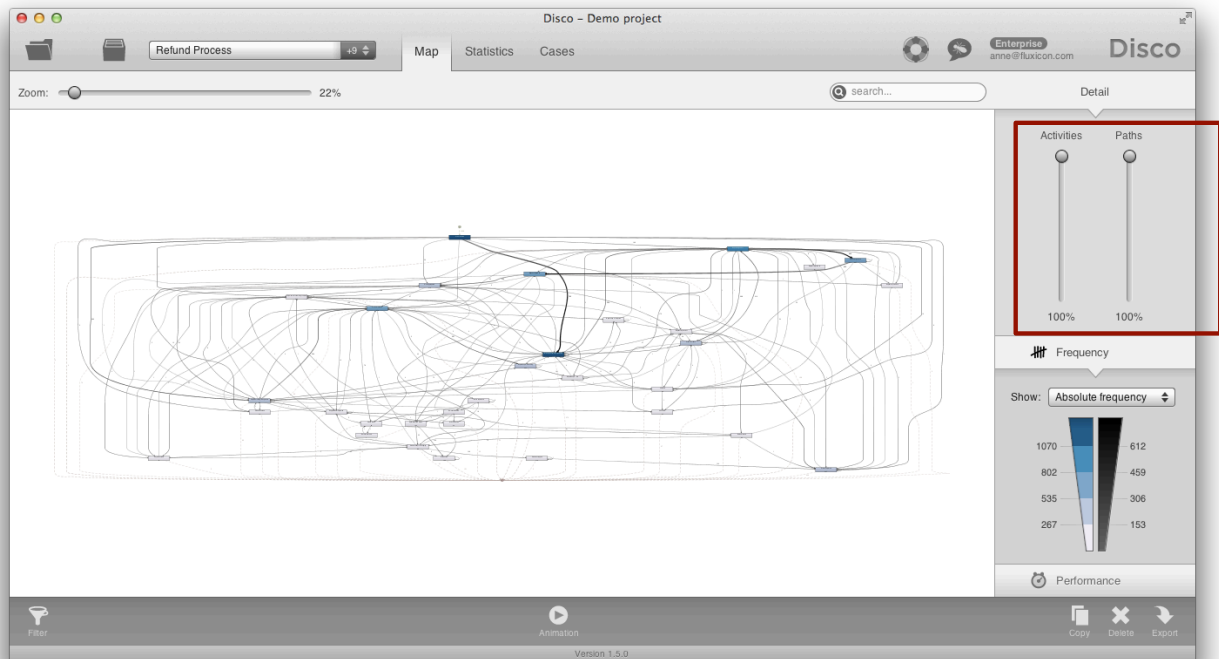
Your friends from Fluxicon,

A handwritten signature in blue ink that reads "Anne + Christa". The signature is written in a cursive, flowing style.

Part I: Quick Simplification Methods

1) Interactive Simplification Sliders

First, we look at two simplification methods that you can use to quickly get to a simpler process map. The first one is to use the interactive simplification sliders that are built in the map view in Disco (see below).



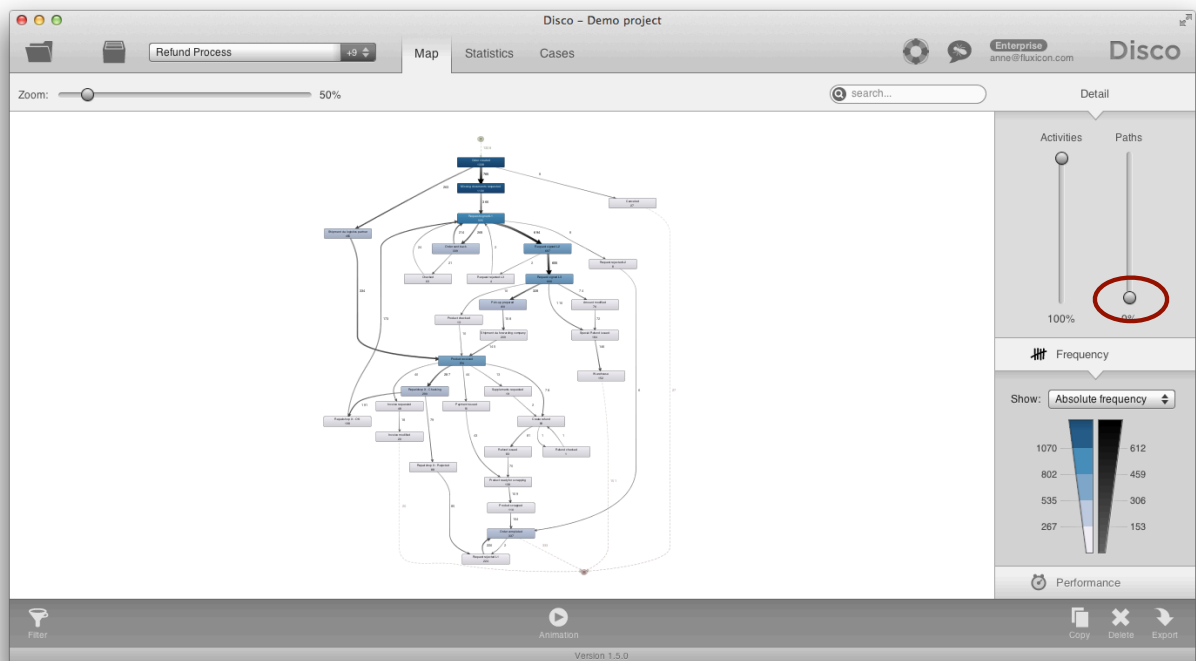
The Disco miner is based on Christian's Fuzzy Miner [1], which was the first mining algorithm to introduce the “map metaphor”, including advanced features like seamless process simplification and highlighting of frequent activities and paths. However, the Disco miner has been further developed in many ways.

One important difference is that if you pull both the Activities and the Paths sliders up to 100% then you see an exact representation of the process. The complete picture of the process is shown, exactly as it happened. This is very important as a reference point and one-on-one match of your data to understand the process map.

However, without applying any of the simplification strategies discussed later, the complete process is often too complex to look at on 100% detail.

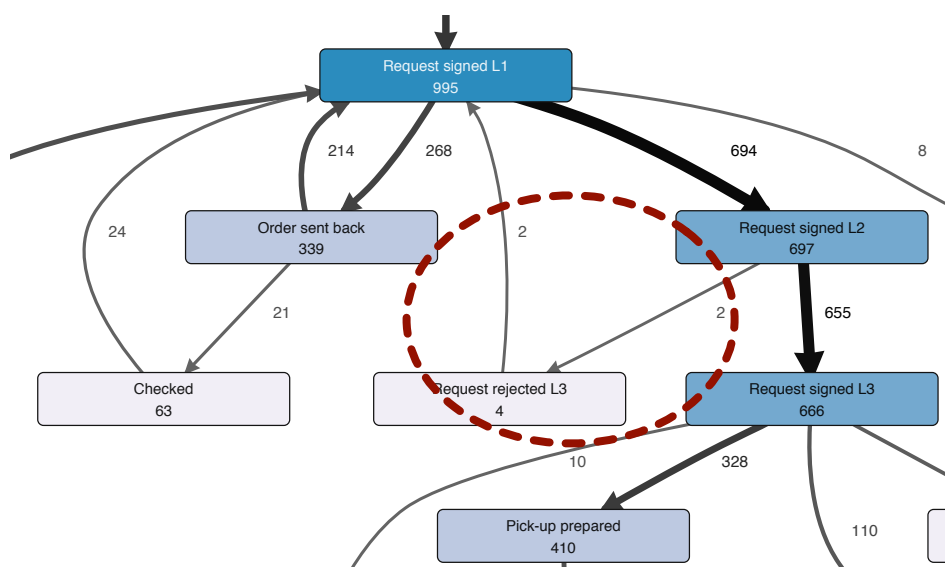
Here is where the interactive simplification sliders can give you a quick overview about the process. We recommend to start by pulling down the Paths slider, which gradually reduces the arcs in the process map by hiding less frequent transitions between activities.

At the lowest point, you only see the most frequent process flows, and you can see that the “spaghetti” process map from above has been simplified greatly, already yielding a very readable and understandable process map (see below).



What you will notice is that some of the paths that are shown can be still quite low-frequency. For example, in the following fragment you see that there are two paths with just the frequency 2 (see below). The reason is that the Paths simplification slider is smart enough to take the process context into account and sees that these paths connect the very low-frequency activity 'Request rejected L3', which just occurred 4 times (see below). It would not be very useful to have low-frequency activities "flying around", disconnected from the rest of the process.

The Paths slider is very important, because it allows you to see *everything that has happened* in your process (*all the activities* that were performed), but still get a readable process map with the main flows between them.

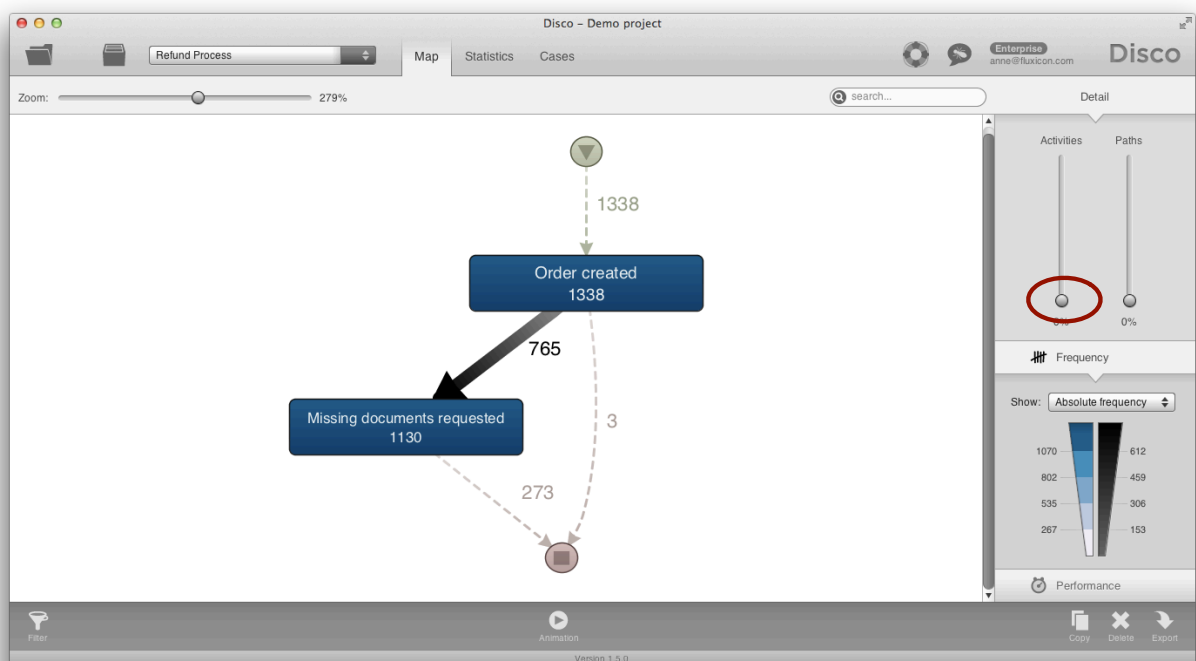


Often, you will find that getting a quick process map with all the activities shown (Activities slider up at 100%) and only the main process flows (Paths slider down at lowest point, or slightly up, depending on the complexity of the process) will give you the best results.

However, if you have many activities, or if you want to further simplify the process map, you can also reduce the number of activities by pulling down the Activities slider (see below).

At the lowest point, the Activities slider shows you only the activities from the most frequent process variant (see also next section). This means that only the activities that were performed on the most frequent path from the very beginning to the very end of the process are shown. So, this shows you really the main flow of the process (now also abstracting from less frequent activities, not only less frequent paths).

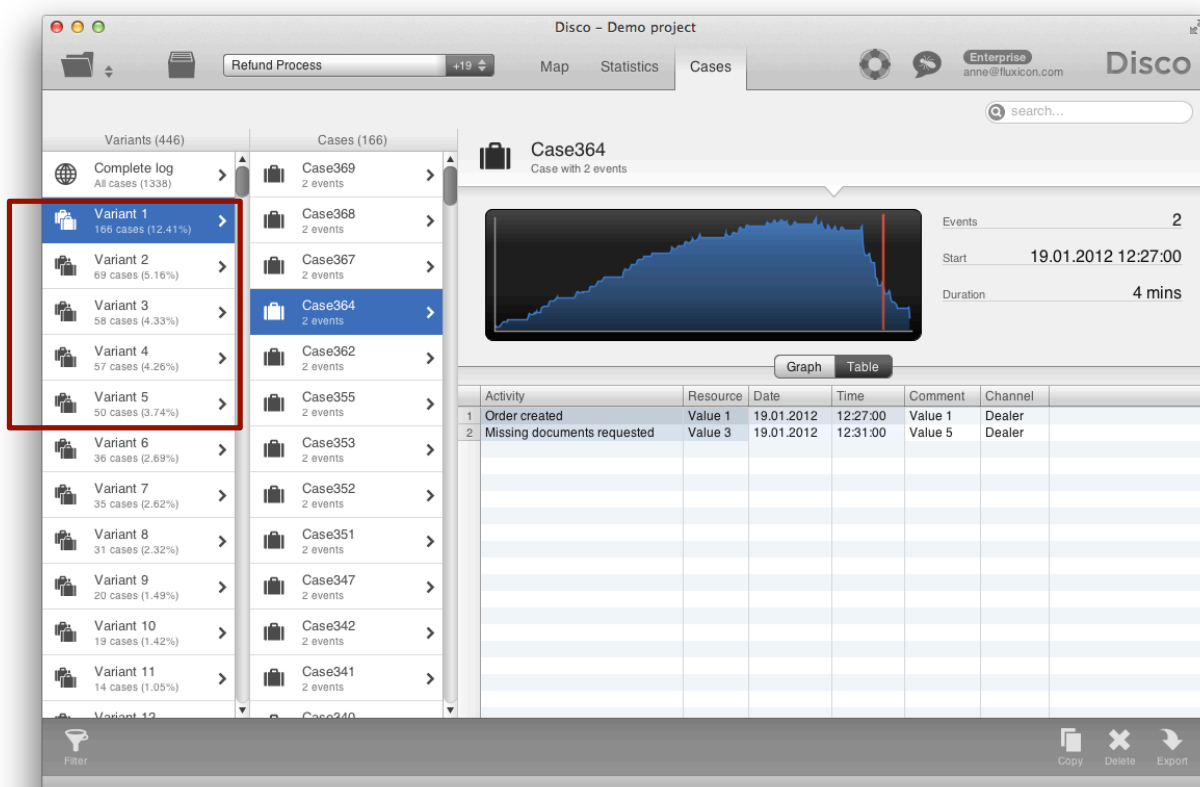
For example, the “spaghetti” process map from the beginning could be greatly simplified to just the main activities ‘Order created’ and ‘Missing documents requested’ by pulling down the Activities slider (see below).



2) Focusing on the Main Variants

An alternative method to quickly get a simplified process map is to focus on the main variants of the process. You find the variants in the Cases view in Disco.

For example, one case from the most frequent variant (Variant 1) is shown in the screenshot below: There are just two activities in the process, first 'Order created' and then 'Missing documents requested' (so, most cases are actually, strangely, waiting for feedback from the customer, but we are not focusing on this at the moment).

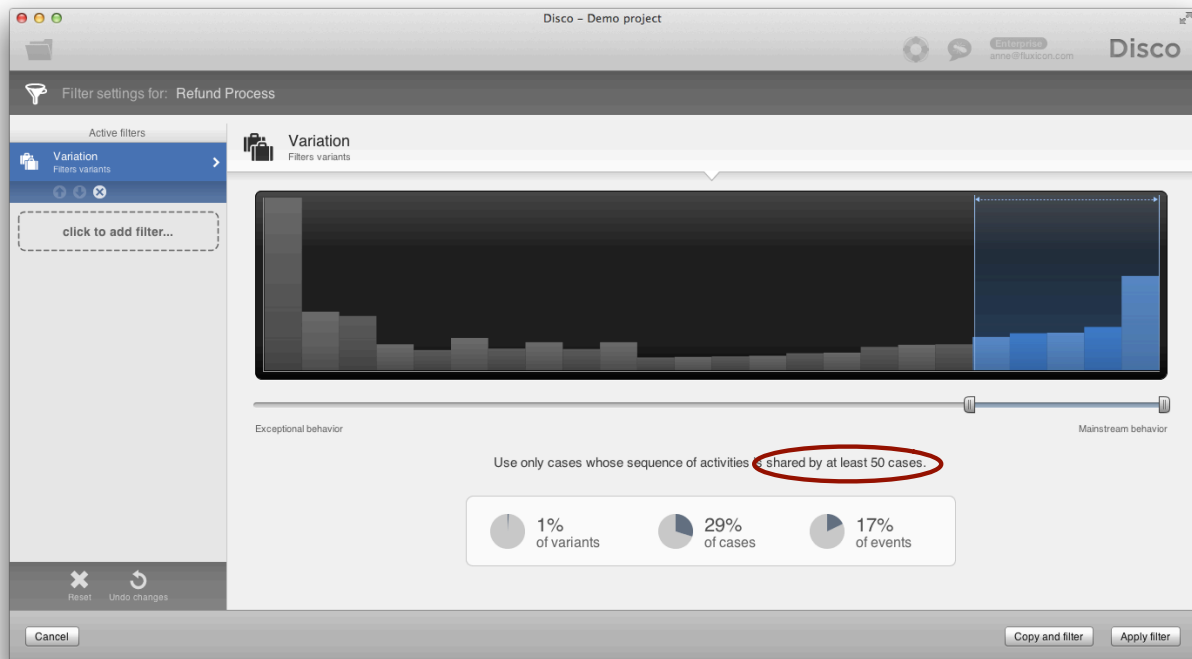


If you look at the case frequencies and the percentages for the variants, then you can see that the most frequent variant covers 12.41%, the second most frequent covers 5.16% of the process, etc. What you will find in more structured processes is that often the Top 5 or Top 10 variants may already be covering 70-80% of your process. So, the idea is to directly leverage the variants to simplify the process.

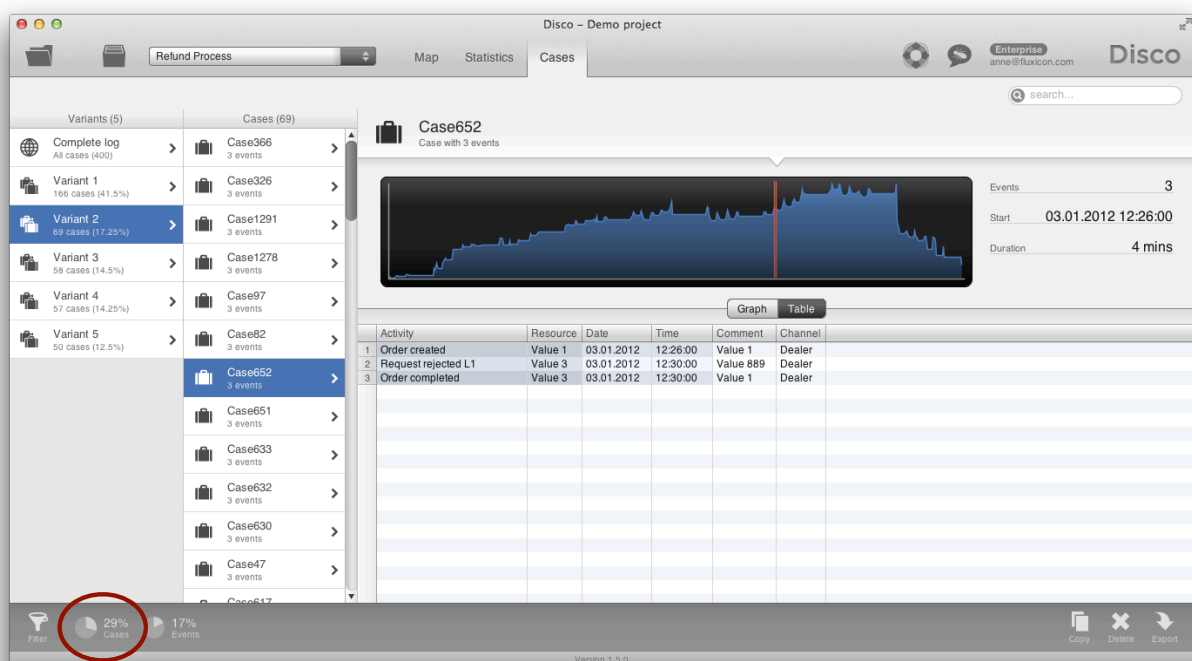
Note: This strategy only works for structured processes. In unstructured processes (for example, for patient diagnosis and treatment processes in a hospital, or for clicks-streams on a website) you often do not have any dominant variants at all. Every case is unique.

In such unstructured processes, variant-based simplification is completely useless, but the interactive simplification sliders from the previous section still work (they always work).

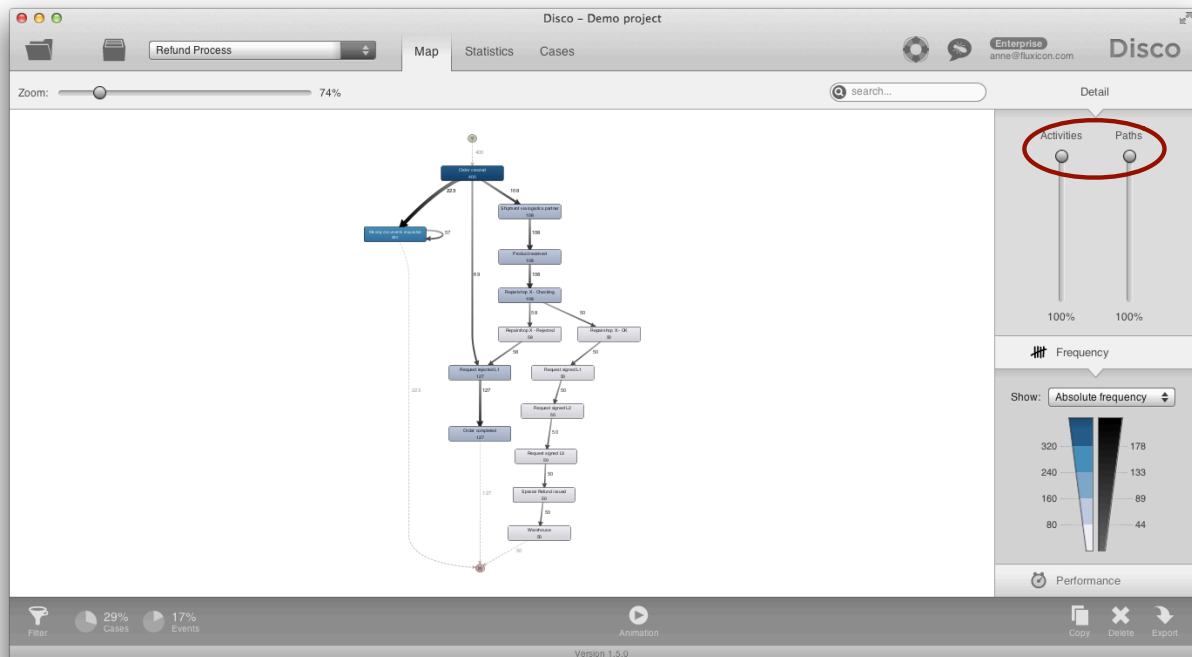
You can easily focus on the main variants in Disco by using the Variation filter (see below). For example, here we focus on the Top 5 variants by only keeping the variants that have a support of 50 cases or more.



Only the Top 5 variants are kept and we see that these few (out of 446) variants are covering 29% of the cases.



If you now switch back from the Cases view to the Map view, you can see the process map just for those 5 variants (see below).



The trick here is that, this way, you can easily create a process map with 100% detail (notice both the Activities and paths sliders are pulled up completely) - But of course only for the variants that are kept by the filter.

This method can be particularly useful if you need to quickly export a process map for people who are not familiar with process mining. If you export the process map with 100% detail then all the numbers add up (no paths are hidden) and you do not need to explain what “spaghetti” processes are and why the process map needs to be simplified. You can simply send them the exported PDF of the process map and say, for example, “This is how 80% of our process flows” (depending how many % your variant selection covers).

Note that less frequent activities are often hidden in the more exceptional variants, and you do not see them when you focus on the main variants. Use the interactive simplification sliders from the previous section to quickly get a simplified map with the complete overview of what happens in your process.

Part II: Remove Incomplete Cases

3) Remove Incomplete Cases

Especially, if you just got a new data set and simply want to make a first process map, you typically do not want to get into a detailed analysis right away. For example, you often want to validate that the extracted data is right, or you might need to quickly show the process owner a first picture of how the discovered process looks like.

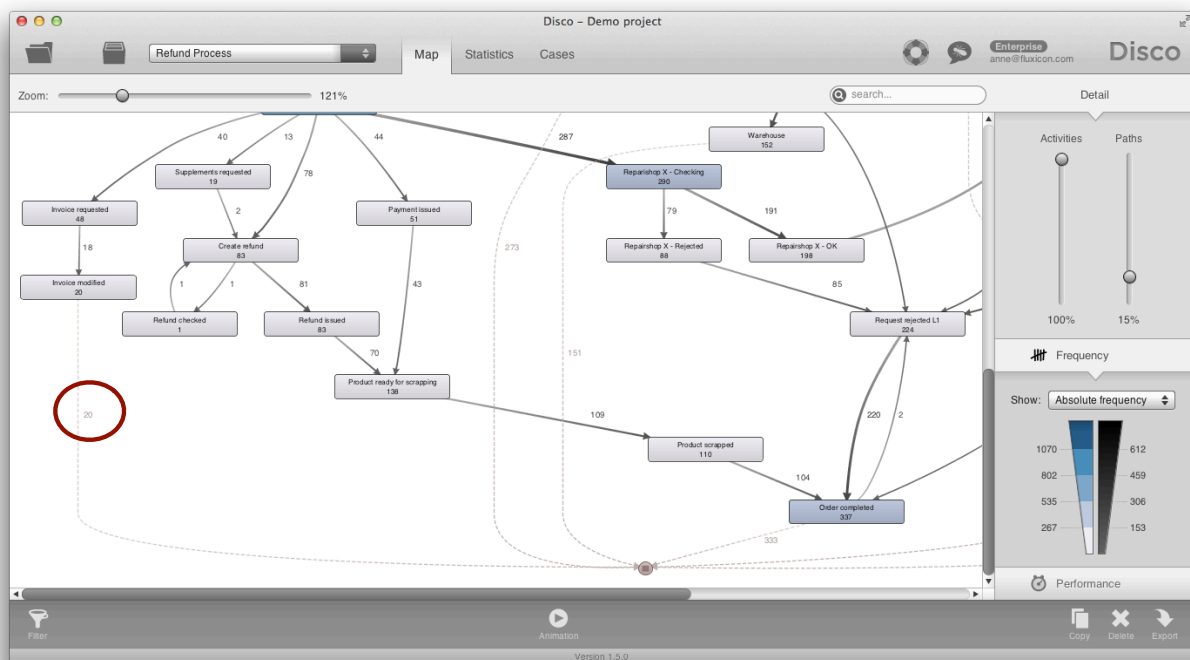
Obviously, a complex process map is getting in your way to do that.

Now, while filtering incomplete cases is a typical preparation step for your actual analysis, you might want to check whether you have incomplete cases also to get a simpler process map. Here is why.

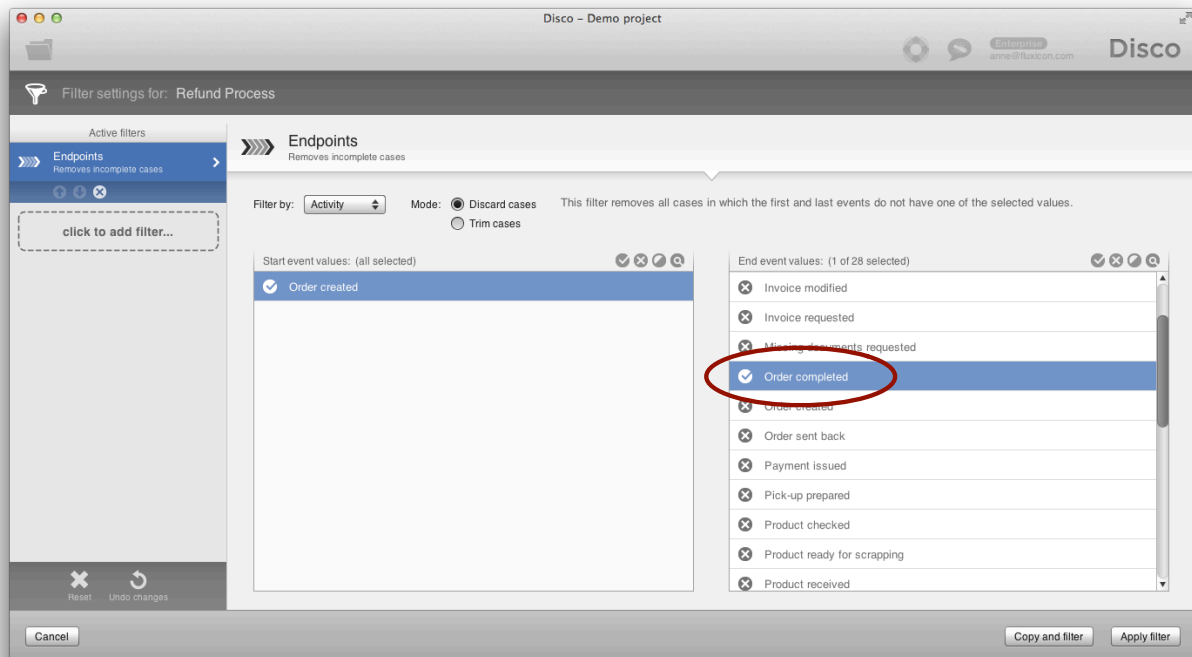
In many cases, the data that is freshly extracted from the IT system contains cases that are not yet finished. They are in a certain state now and if we would wait longer then new process steps would appear. The same can happen with incomplete start points of the process (things may have happened before the data extraction window).

For the analysis of, for example, process durations it is very important to remove incomplete cases, because otherwise you will be judging half-finished cases as “particularly fast”, reducing the average process duration in a wrong way. But incomplete cases can also inflate your process map layout by adding many additional paths to the process end point.

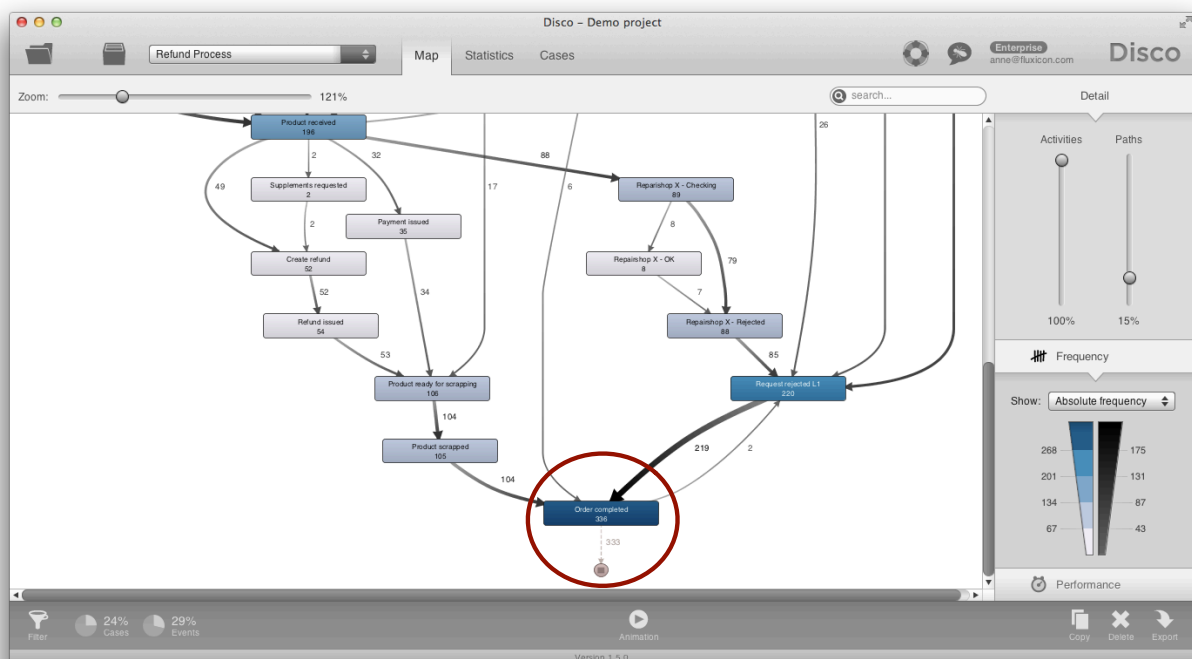
To understand why, take a look at the process map below. It shows that next to the regular end activity ‘Order completed’ there are several other activities that were performed as the last step in the process — showing up as dashed lines leading to the end point at the bottom of the map. For example, ‘Invoice modified’ was the last step in the process for 20 cases (see below). This does not sound like a real activity for the process, does it?



By removing incomplete cases, you can just add an Endpoints filter in Disco and select the start and end activities that are valid start and endpoints in your process (see below).



The resulting process map will be simpler, because the graph layout becomes simpler (see below).



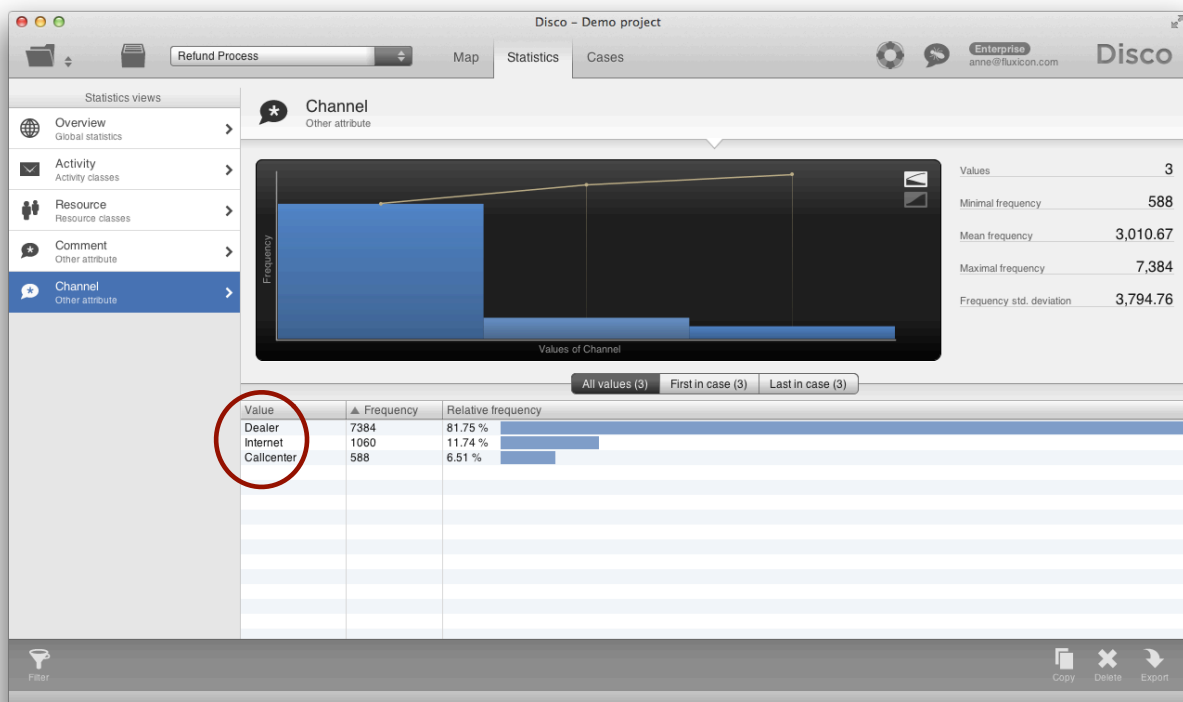
Part III: Divide and Conquer

4) Multiple Process Types

The third category of strategies is called ‘Divide and conquer’ because it is about breaking up your data in various ways to make it more manageable.

A first way to split up your data is to realize that very often your process actually consists of multiple process types. You may get the whole data set in one file, because this is how it is extracted, but this does not necessarily mean that you have to analyze all that data at once.

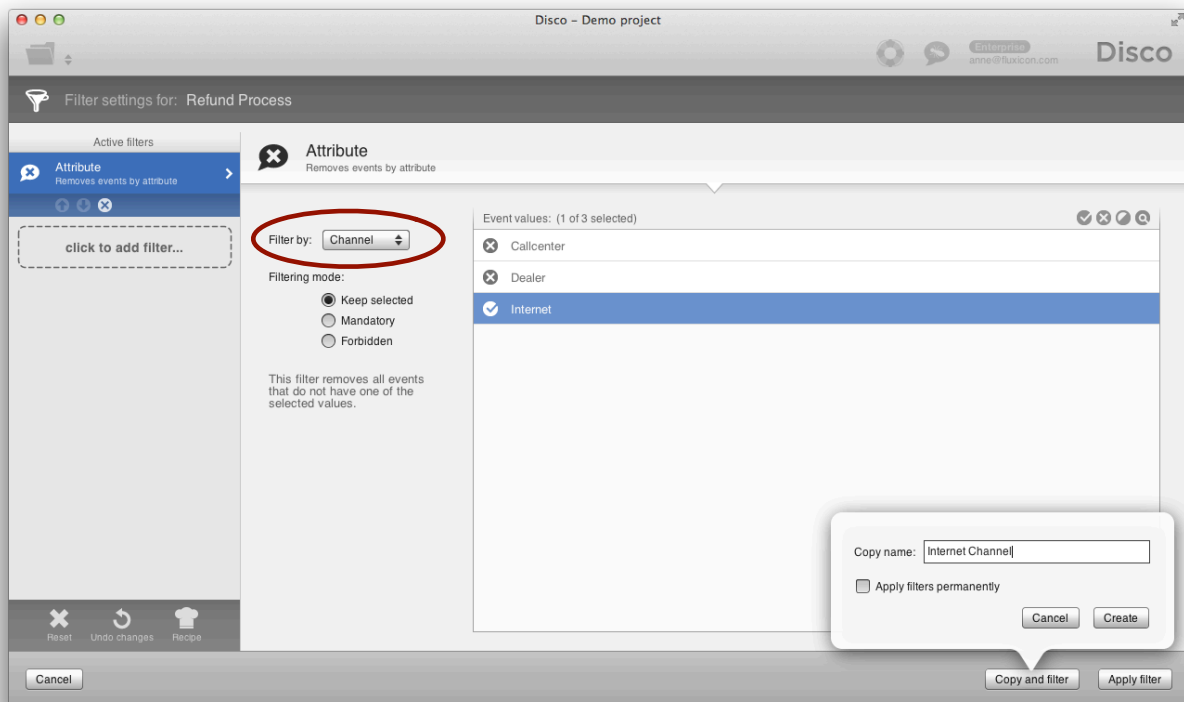
For example, the customer service refund process used as example in the previous sections has an attribute that indicates the channel by which the process was started: Customers can (a) initiate the refund themselves through the internet by filling out a form, (b) they can call the help desk, or (c) they can go back to the dealership chain, where they bought the product in the first place (see below).



The processes for these different channels are not the same. For example, the refund process for the dealer channel involves completely different process steps than for the other two channels. However, if we do not separate them from each other then we get all of the different processes in one picture, making the process map unnecessarily complicated.

A similar situation can be found in IT Service Desk processes. For example, in a change management process the actual process steps can be quite different depending on the change category: Implementing a change to the SAP system is not the same as creating a new user account. The change category attribute can be used to separate the data for these different process types.

In Disco, you can easily filter data sets on any process attribute that you have imported. Simply add an Attribute filter and select the attribute indicating your process type in the 'Filter by' drop-down list (see below).



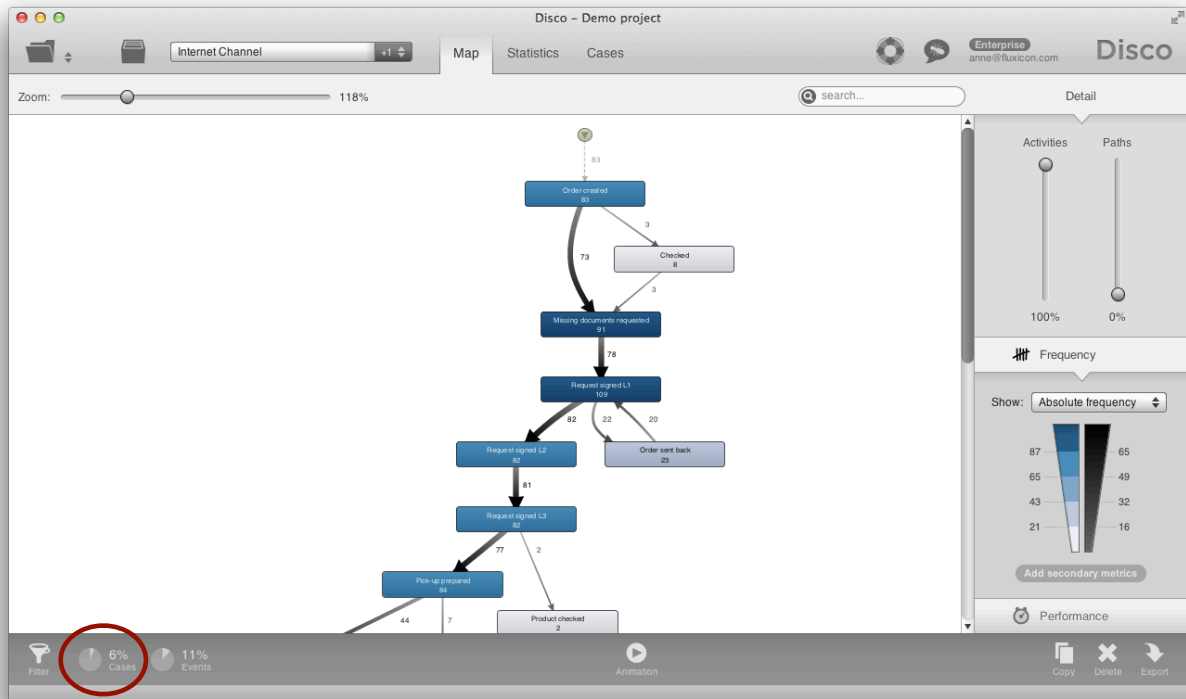
What we recommend when you split up data sets is that you use the 'Copy and filter' button instead of the 'Apply filter' button to apply the filter to a copy (see above). For example, for three different process types, you can simply create three copies, one for each process type, to further analyze these processes in isolation.

In fact, creating copies is a very good idea for many situations: Every copy is preserved in your Disco project view, and you can easily switch back and forth between them, record notes about your observations, and so on.

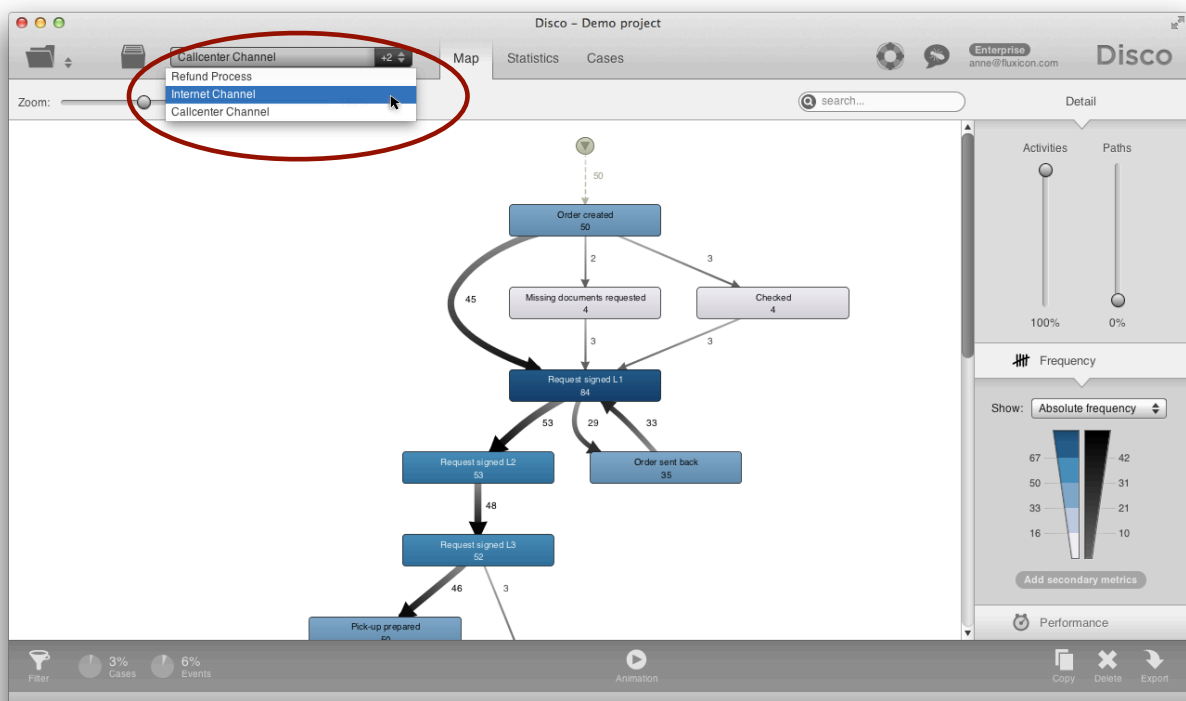
When you create the copy, make sure to give it a name that is meaningful, for example, indicating the process type that is analyzed. This way, you can find them back quickly.

Note: Note that copies are managed efficiently in Disco (pointing to the same underlying data set where possible), so you do not need to be afraid to use them also for very large data sets.

For example, here you see the refund process, filtered for the Internet channel (covering 6% of the cases).



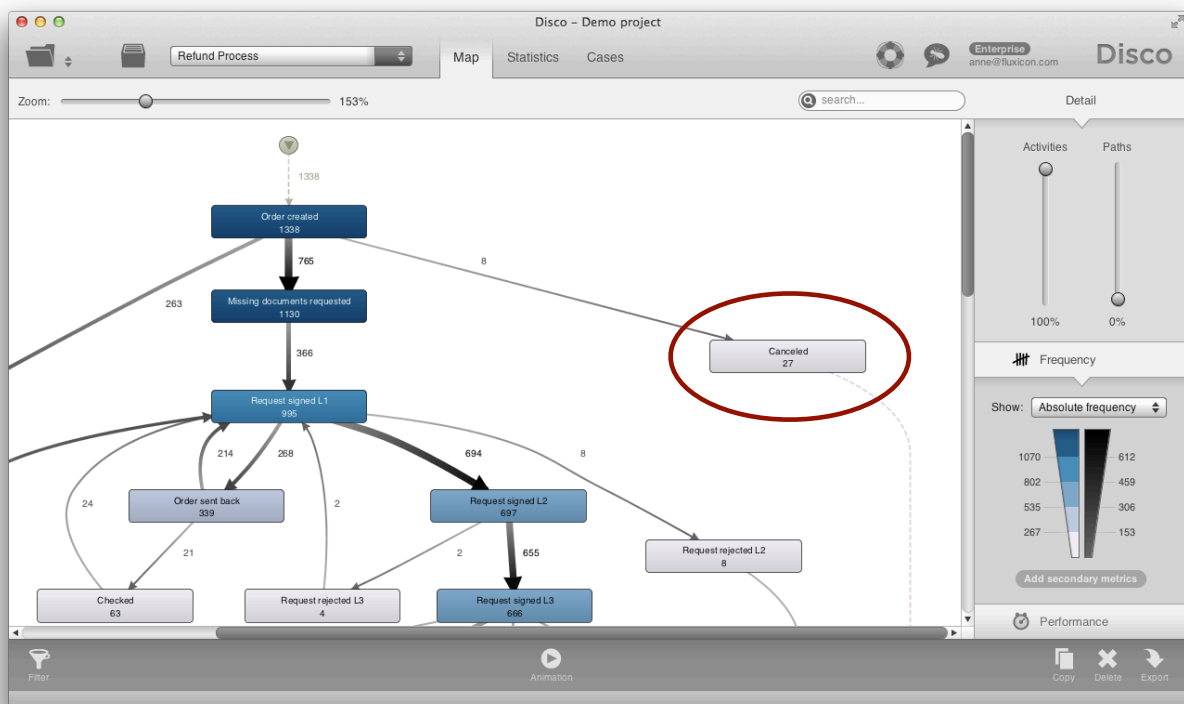
And this is the process for the Callcenter channel. Through the drop-down list you can quickly switch back and forth between them.



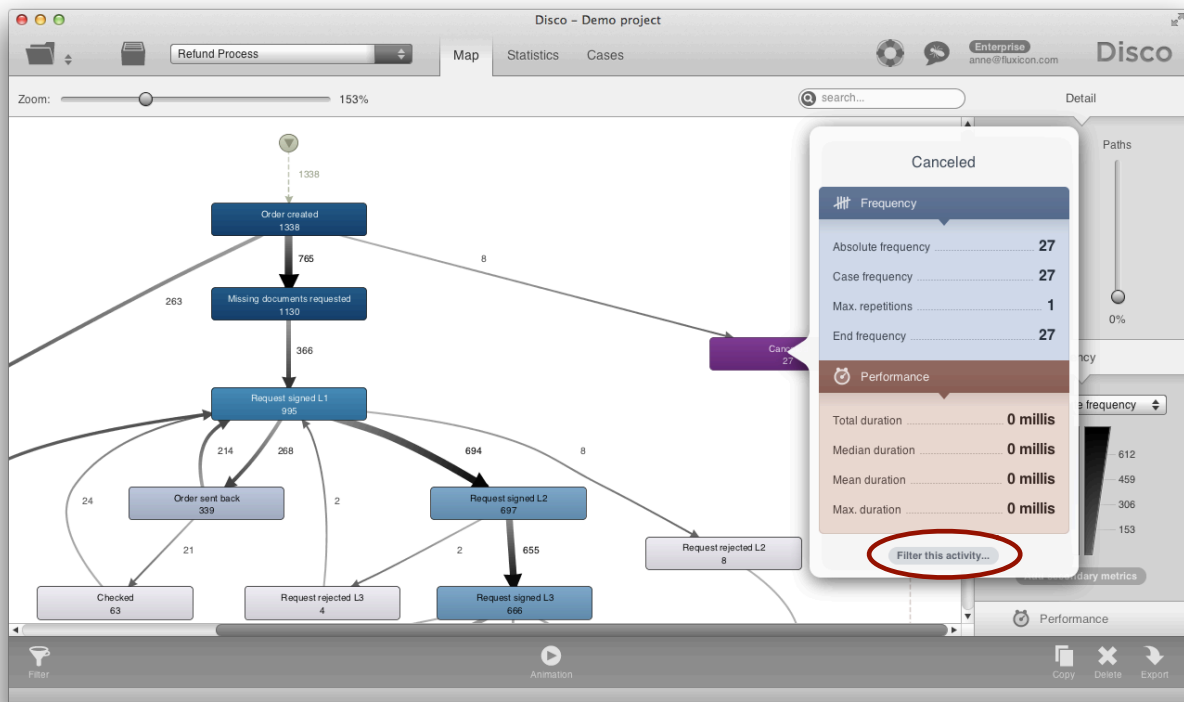
5) Semantic Process Variants

A second way to split up the data set is by so-called “semantic process variants”. The idea here is that, again, there are multiple process types that should be separated, but in this case there is no attribute available that can be simply used to filter for this category.

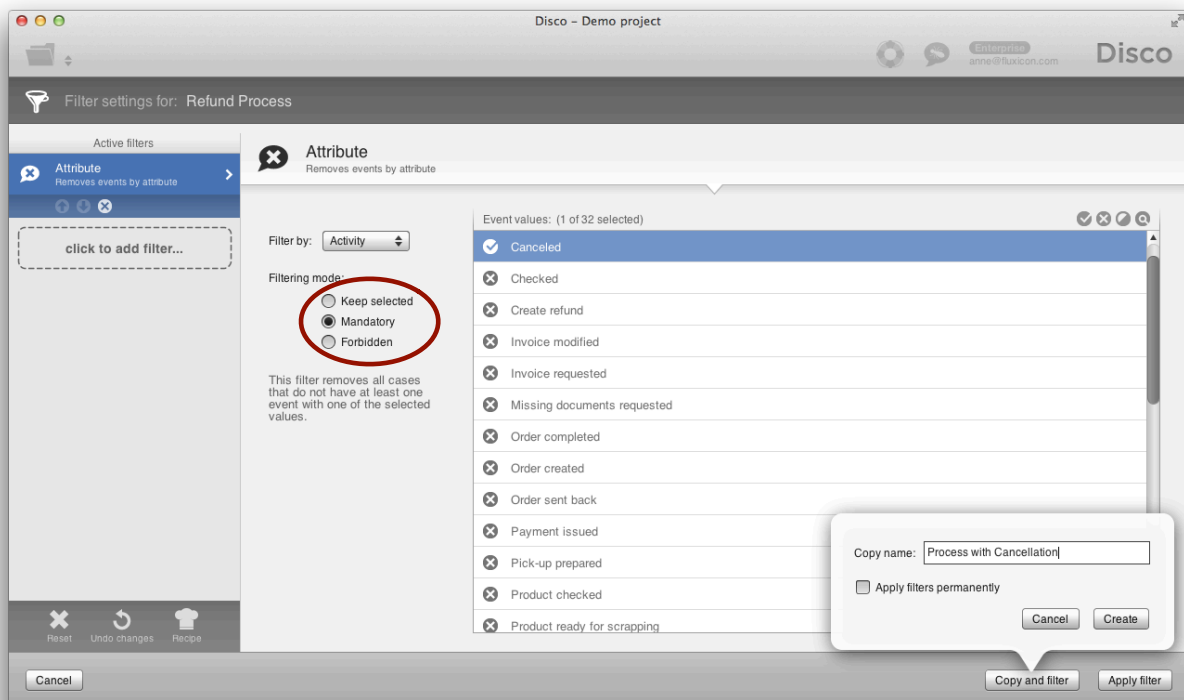
Instead, the process variant exists implicitly, defined by the business perspective, based on the behavior in the process. For example, for the refund service process discussed above, the process owner made a clear distinction between cancelled and non-cancelled orders. They had made a separate process documentation for when cancellations are possible, so for them cancelled and non-cancelled processes were different process types and needed to be separated.



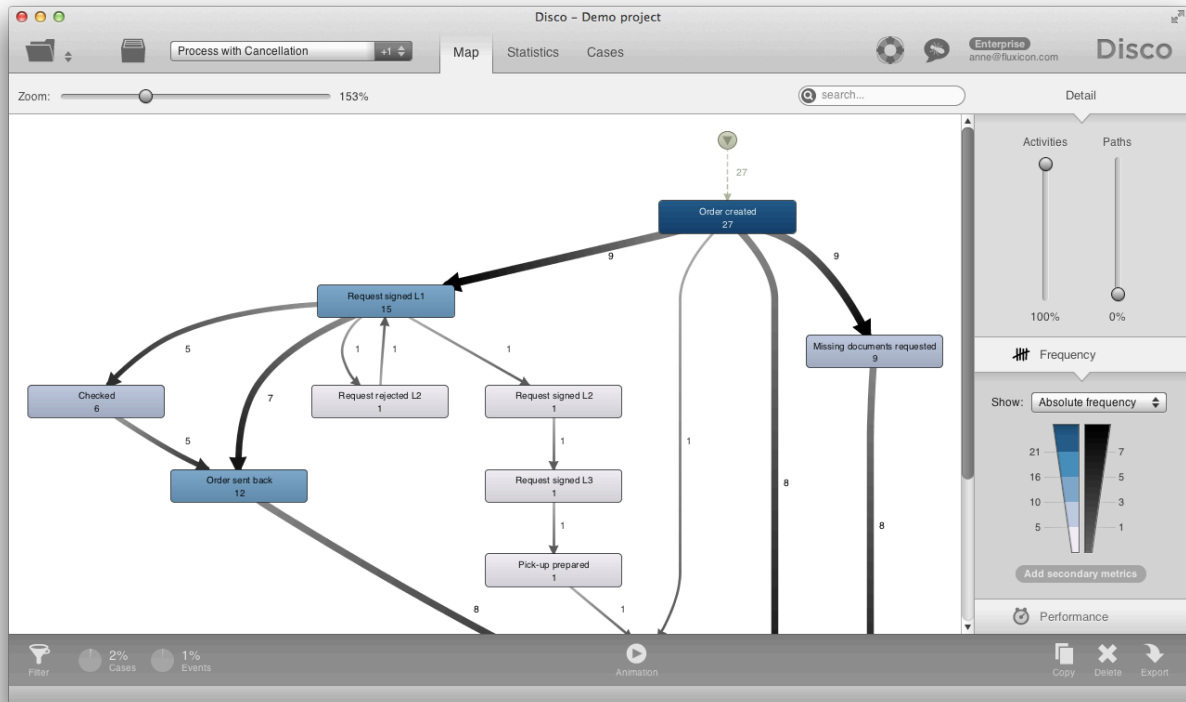
In Disco, you can simply click on an activity to filter cases that perform or do not perform a certain activity. A pop-over dialog with a button ‘Filter this activity...’ appears (see next page).



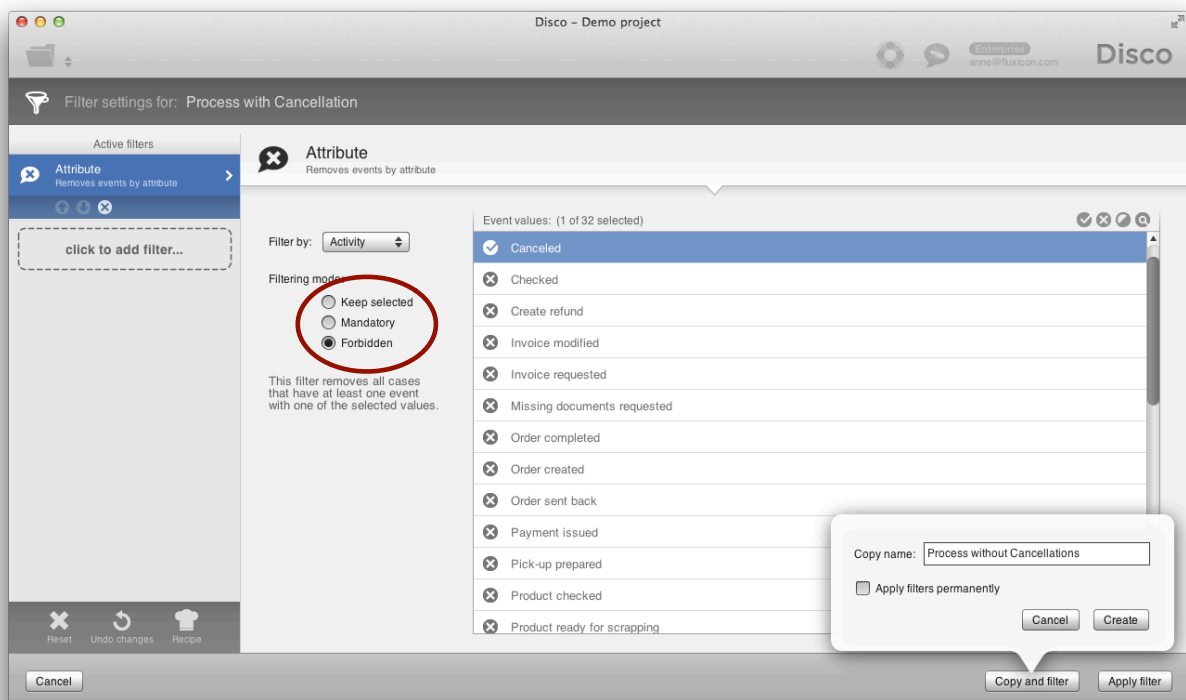
If you press this button, a pre-configured Attribute filter in Mandatory mode will be created (see below).



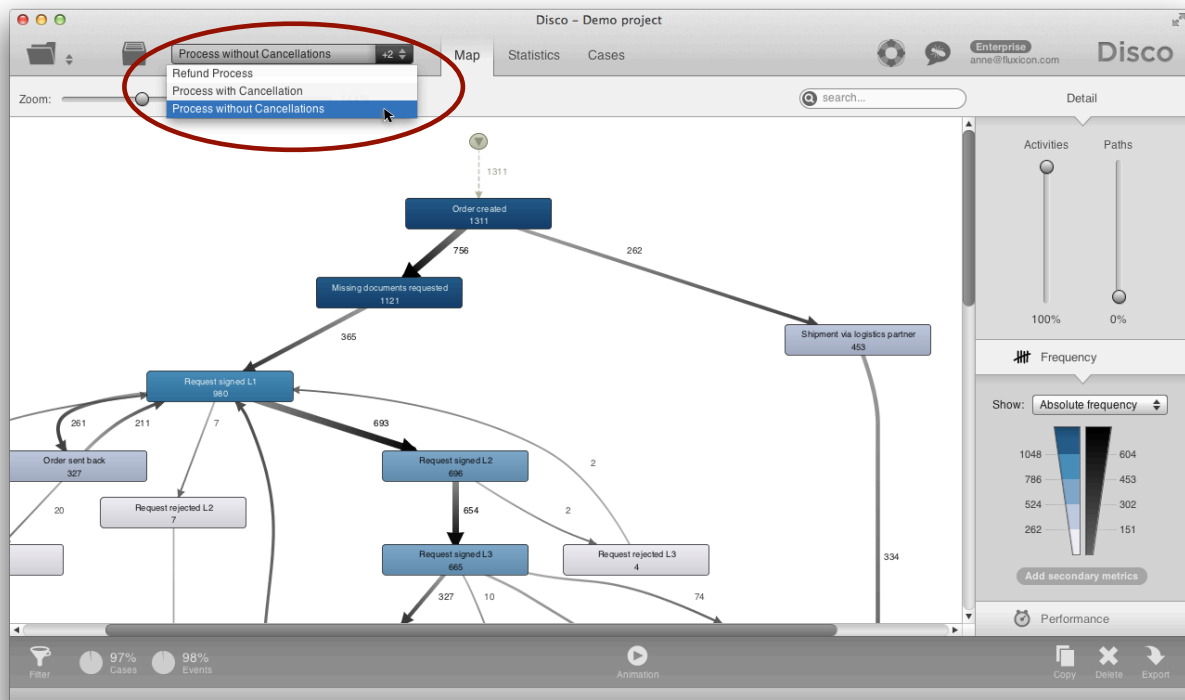
Applying this filter keeps only cases that *at any point in the process* performed the Canceled activity (see below).



Conversely, you can use the Forbidden mode to remove all orders with the Canceled activity from the data set.



In this case, only those cases that *never at any time in the process* performed the Canceled activity remain. Again, you can make copies to keep your divided data sets separated and analyze what happened in canceled orders and in your normal process in isolation.



Compared to the process types filtered by attribute, the semantic process variants are a bit more tricky. You need to talk to the process owner to understand how they look at the process. If they have documented their process, have they created different version based on some variation of behavior in the process? Do they look at claims that need to be approved by the supervisor differently from the standard claims that can be handled directly by the clerk?

Once you have found out how the process is viewed from the stakeholders who work with it every day, process mining gives you a very powerful tool to quickly split up the process in the same way.

Next to the simple presence and absence of activities, you can use many more behavior-based patterns for filtering. For example, the Follower filter can define rules about combinations of activities over time (does something happen before or after something else - directly in the next step or any time later, how much time has passed in between, was it done by the same person, etc.), and you can combine all of the above.

This is one of the greatest powers of process mining: That you can easily define behavior-based process patterns for filtering, without programming, in an interactive and explorative way!

6) Breaking up Process Parts

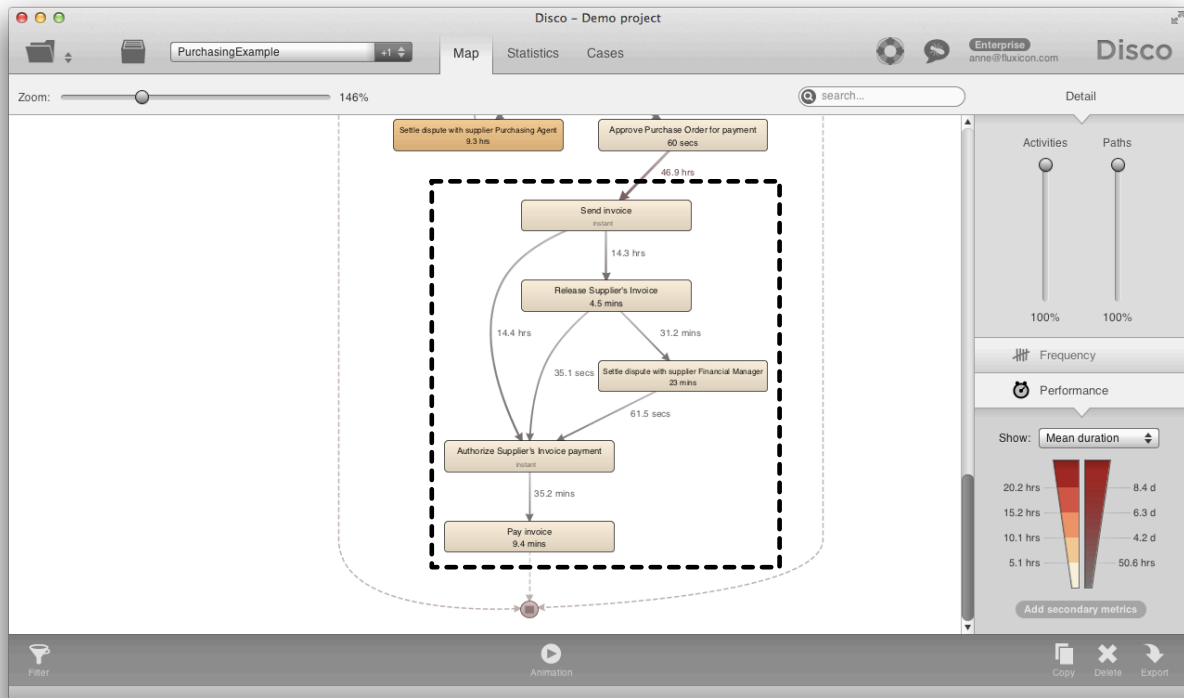
A third way to break up your data set is to focus on a certain part of the process only. You can compare it to taking a pair of scissors and cutting out a part of the process.

Especially for very long processes with many different phases it can be useful to split up the different process parts and analyze them in isolation before putting everything together.

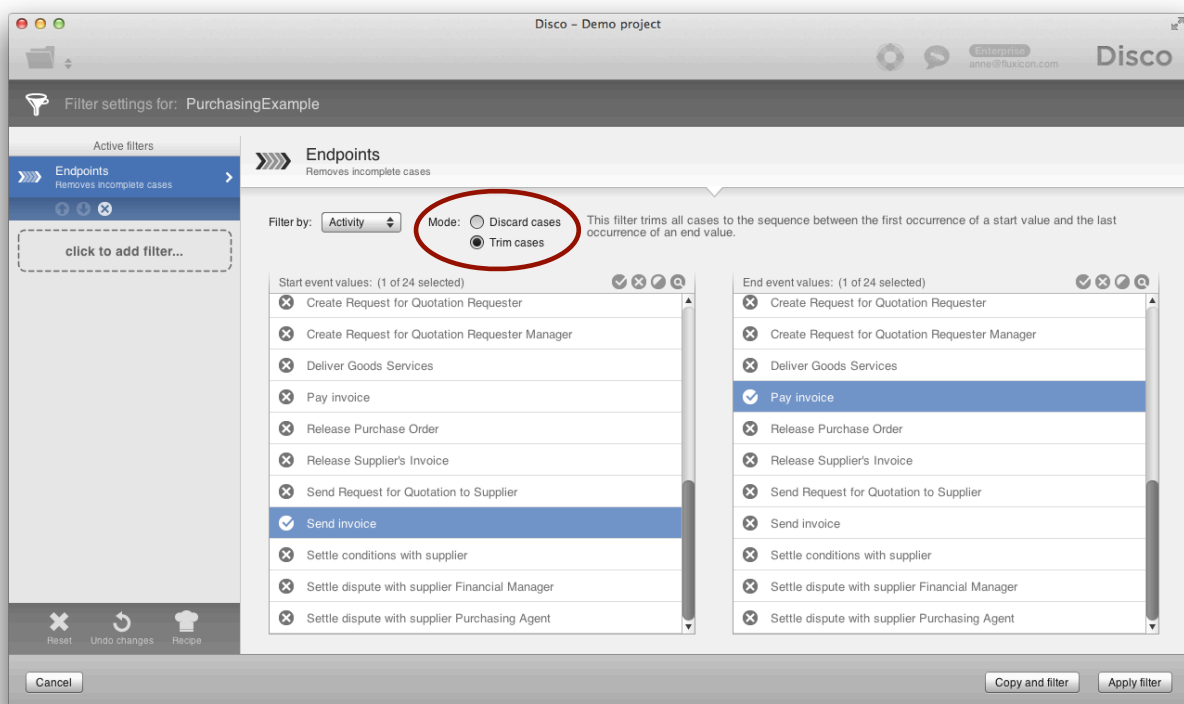


For example, let's take the purchasing example that comes with the sandbox of Disco (see next page). Now assume that you want to focus on the invoicing part of the process only, from the time that the invoice was sent until it was paid (and anything that happened in between).

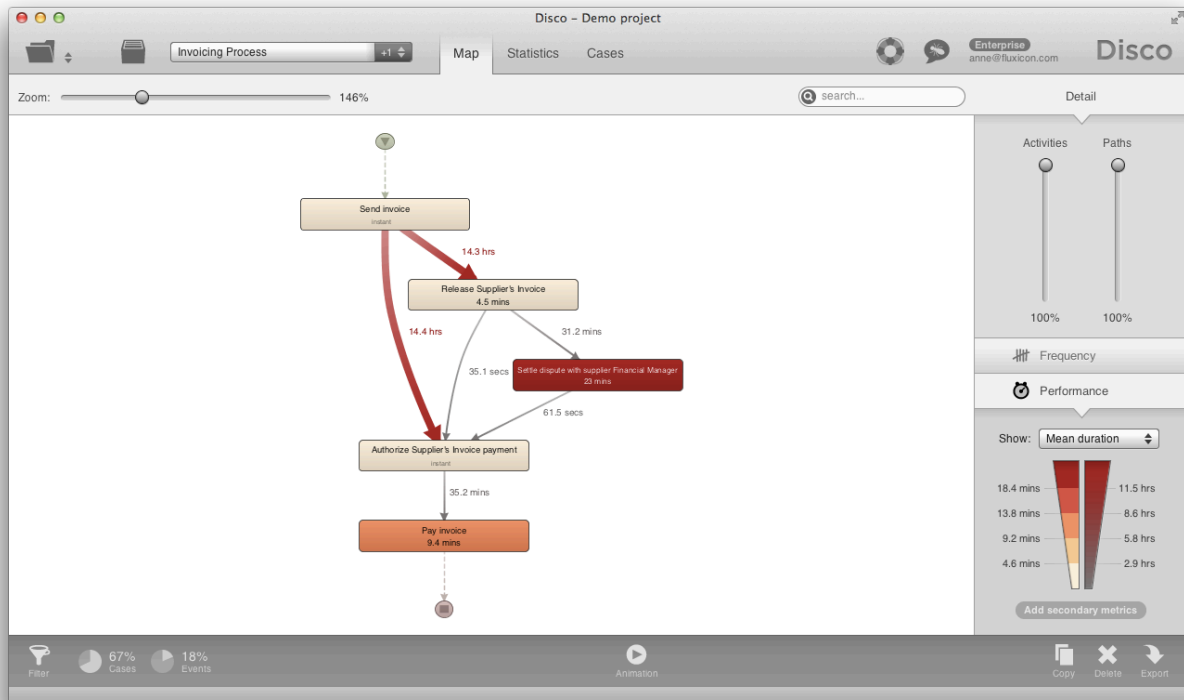
We would like to “cut out” this part of the process (see dashed area mark-up for the part we want to focus on).



The Endpoints filter in Trim mode can be used for this (simply cuts all events before start and after end value):



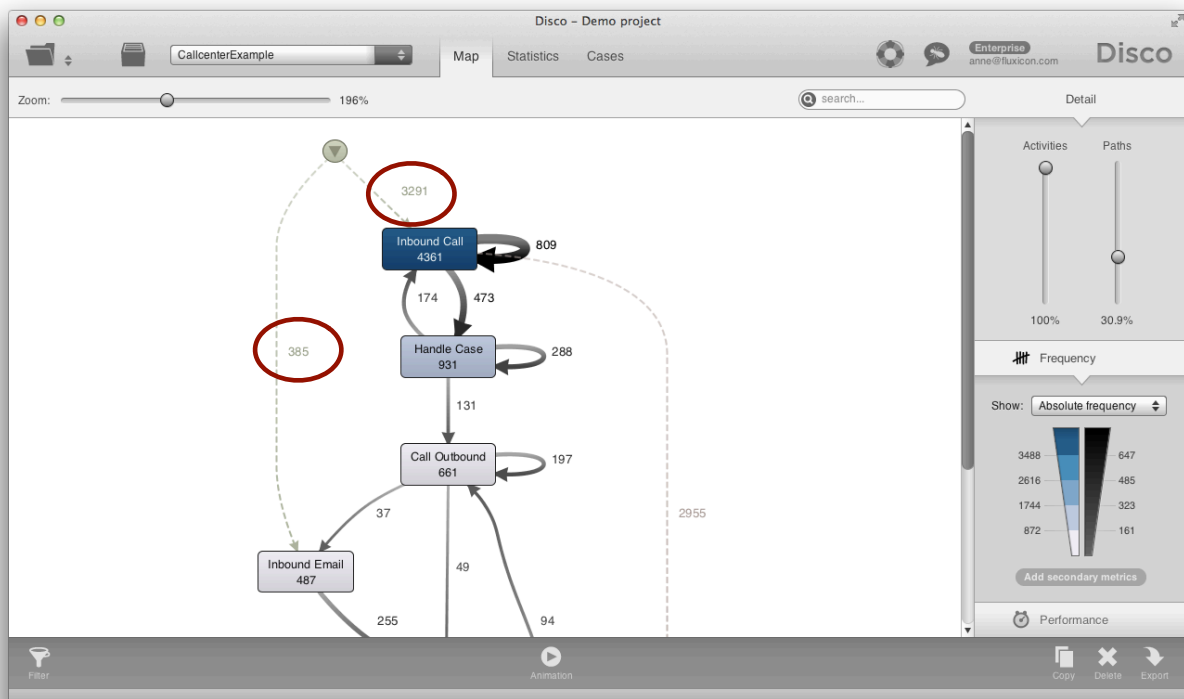
As a result, we have now split up the invoicing process part from the rest of the process and can analyze it in isolation (see below).



7) Different Start and End Points

A fourth divide and conquer strategy is to look at the start and end points of the process.

For example, in the following call center process the customer can start a service request either through a call or through an email, by filling out a form on the website. These different start points are highlighted in the process map by the two dashed lines from the start point (see below).

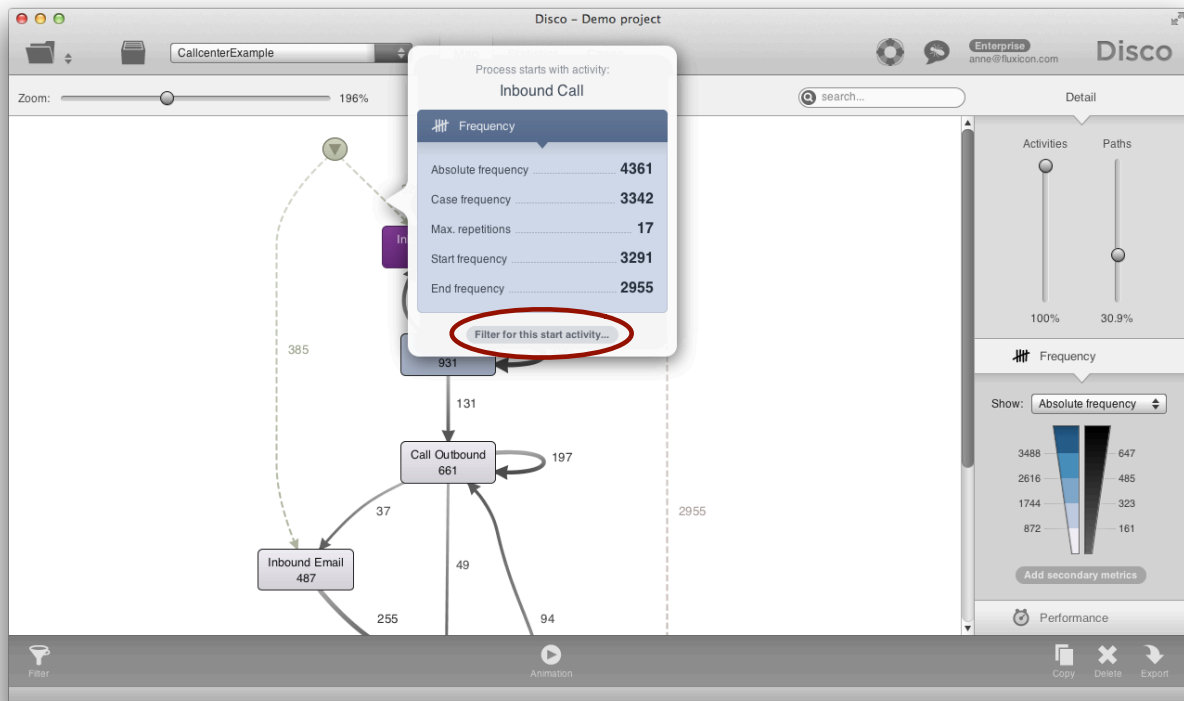


In some situations, the precise process and rules and expectations around the process change depending on how the process was initiated. For example, while it is often the goal to solve a customer problem in the first call (First call resolution rate) this is less realistic in an email thread, which typically needs more interactions to solve a request. This needs to be taken into account in the analysis.

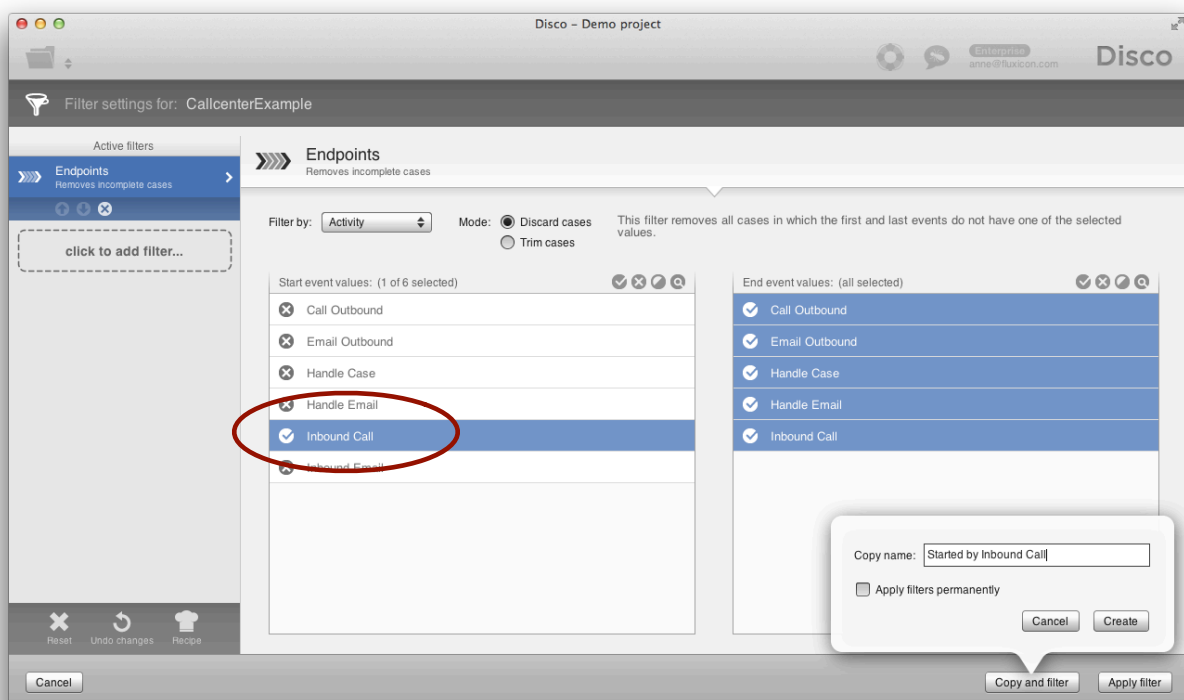
While previously we have already looked at the Endpoints filter in Disco to remove incomplete cases, this time we can use the Endpoints filter to separate data sets based on their start or end points from a business perspective.

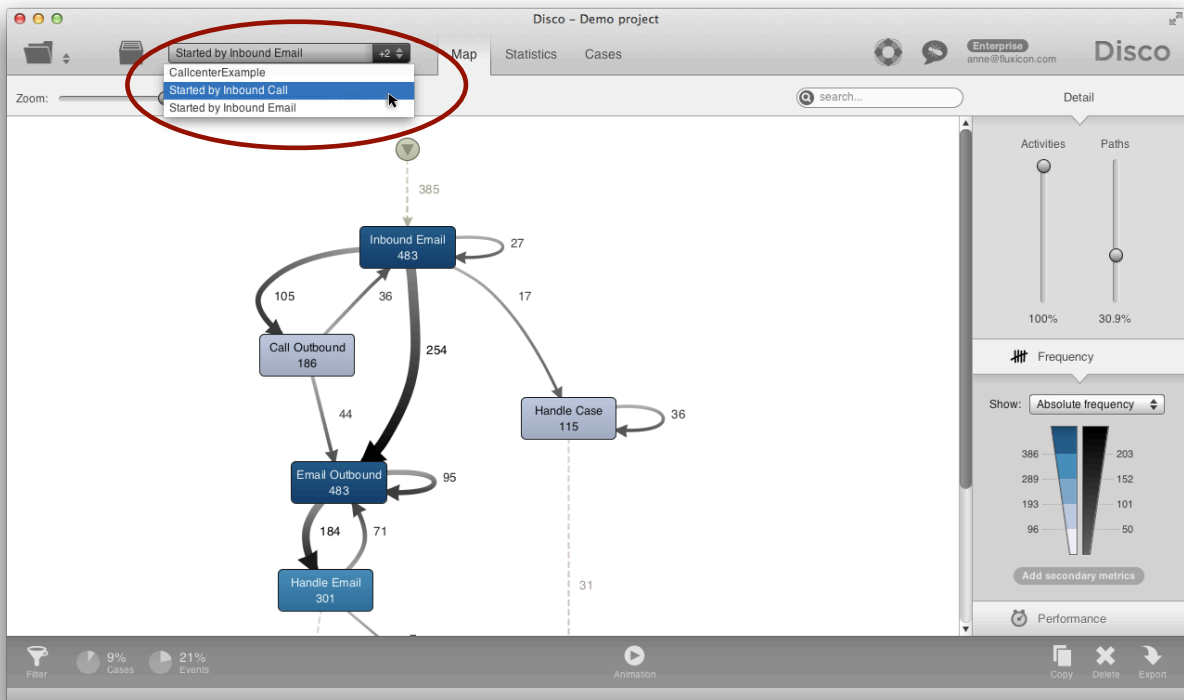
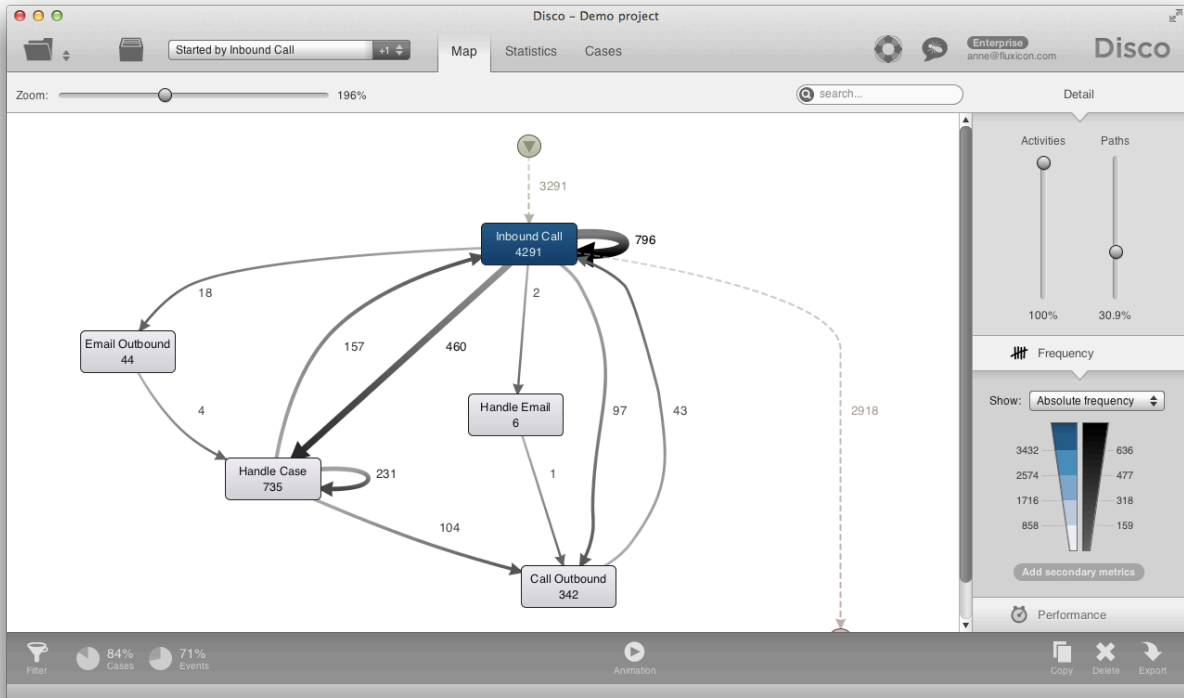
Note: You will see that in many situations you can use the same filter either for cleanup or for analysis purposes, depending on the situation.

In Disco, an Endpoints filter can simply be added by clicking the dashed line in the process map (see below).



You can directly apply the pre-configured filter or, again, make copies to keep them separate (see below).





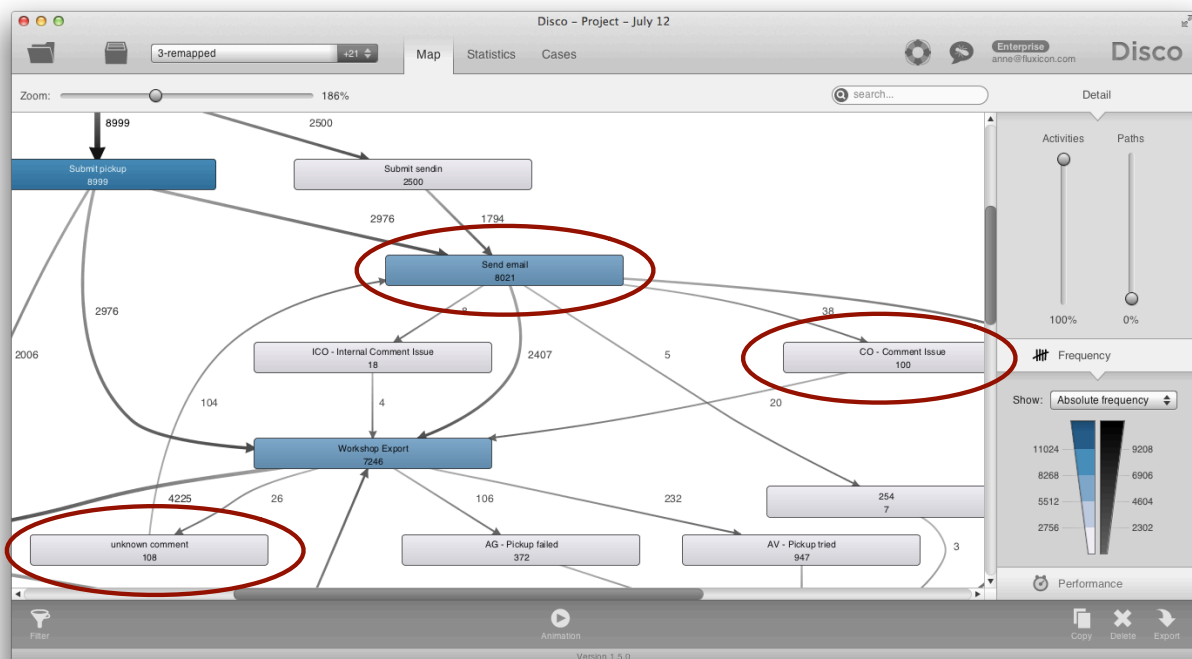
Part IV: Leaving Out Details

8) Removing “Spider” Activities

The last category of simplification strategies is about leaving out details to make the process map simpler.

One way to do that is to look out for what we call “Spider” activities. A spider activity is a step in the process that can be performed at any point in time in the process.

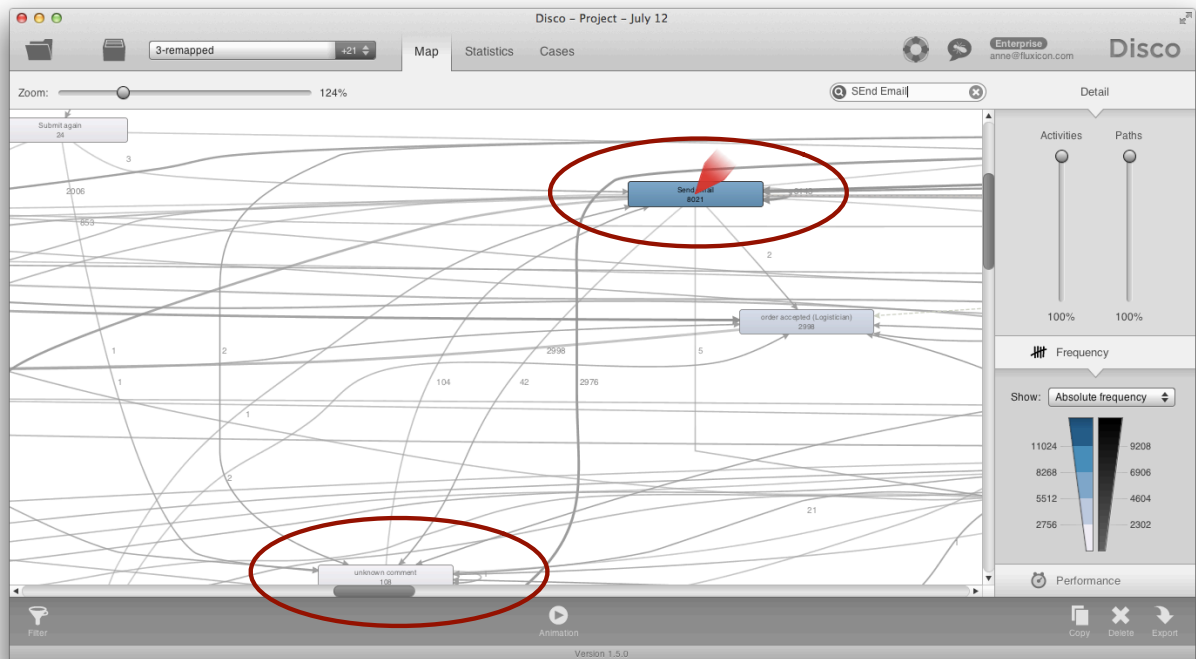
If you take a look at the original service refund process data, you will notice activities such as ‘Send email’ and a few comment activities that are showing up in central places of the process map, because they are connected to many other activities in the process (see below).



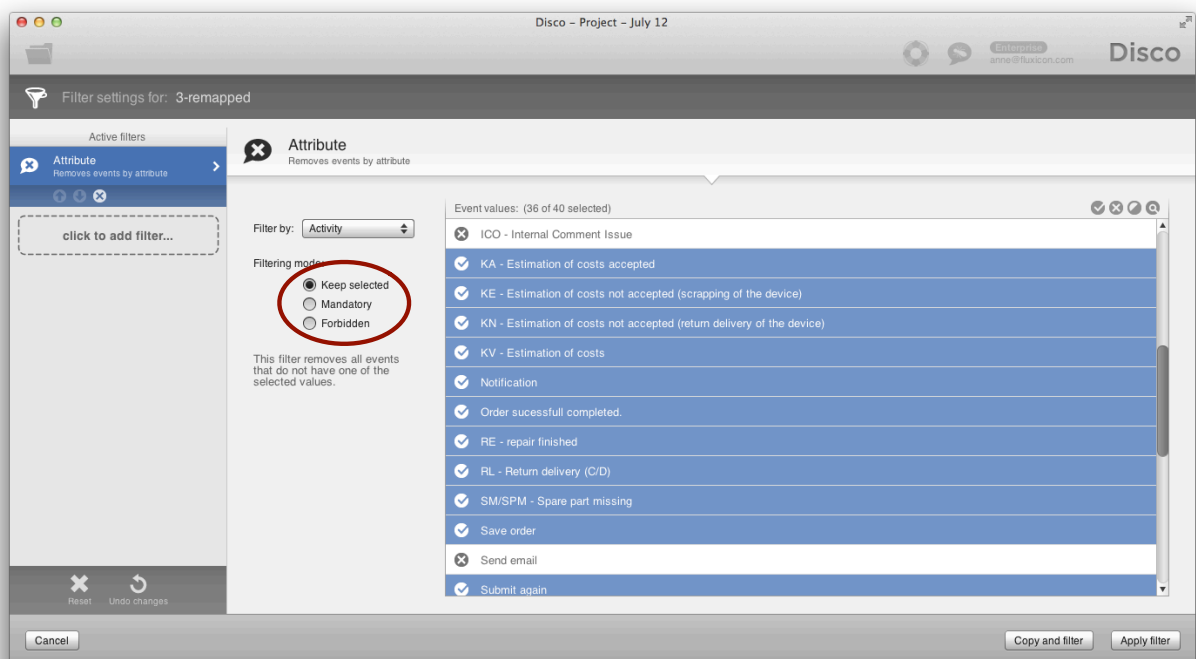
The thing is that — although these activities are showing up in such a central (“spider”) position — they are actually often among the least important activities in the whole process. Their position in the process flow is not important, because emails can be sent and comments can be added by the service employee *at any point* in the process.

Because these activities sometimes happen at the beginning, sometimes at the end, and sometimes in the middle of the process, they have many arrows pointing to them and from them, which unnecessarily complicates the process map.

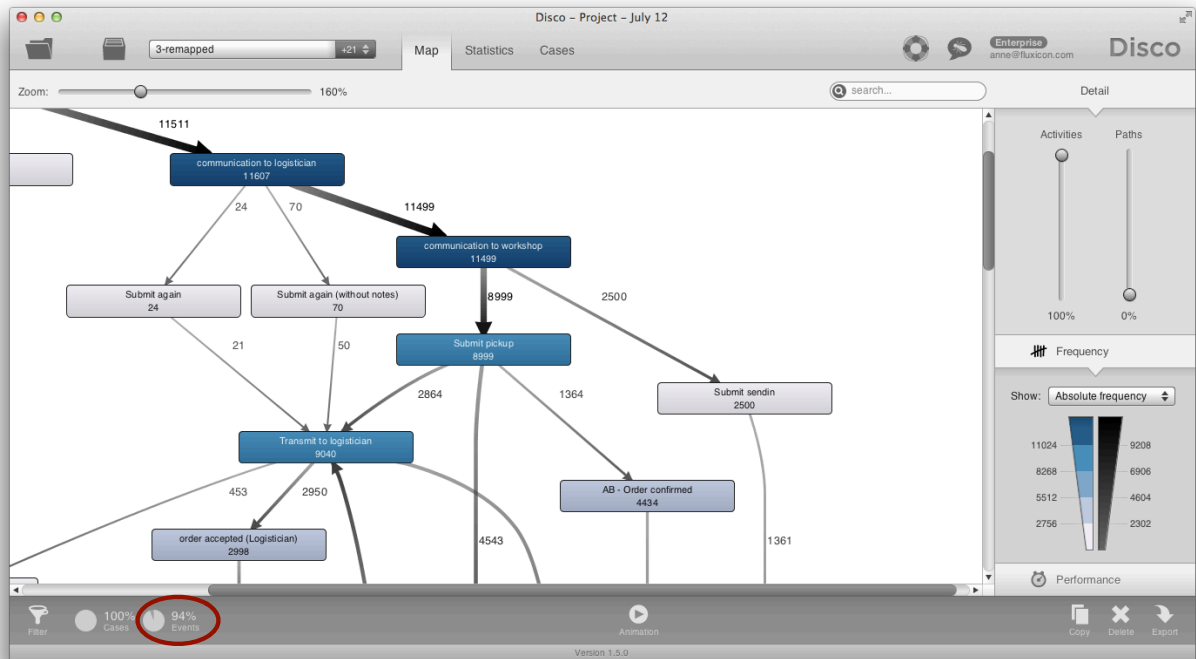
In fact, if we increase the level of detail by pulling up the Paths slider, the picture gets even worse (see next page).



You can easily remove such spider events by adding an Attribute filter and deselecting them (see below). In the standard Keep selected mode this filter will only remove the deselected events but keep all cases.



The result is a much simpler process map, without these distracting “spider” activities (see below). So, the next time you are facing a spaghetti process yourself, watch out for such unimportant activities that merely complicate your process map without adding anything to your process analysis.

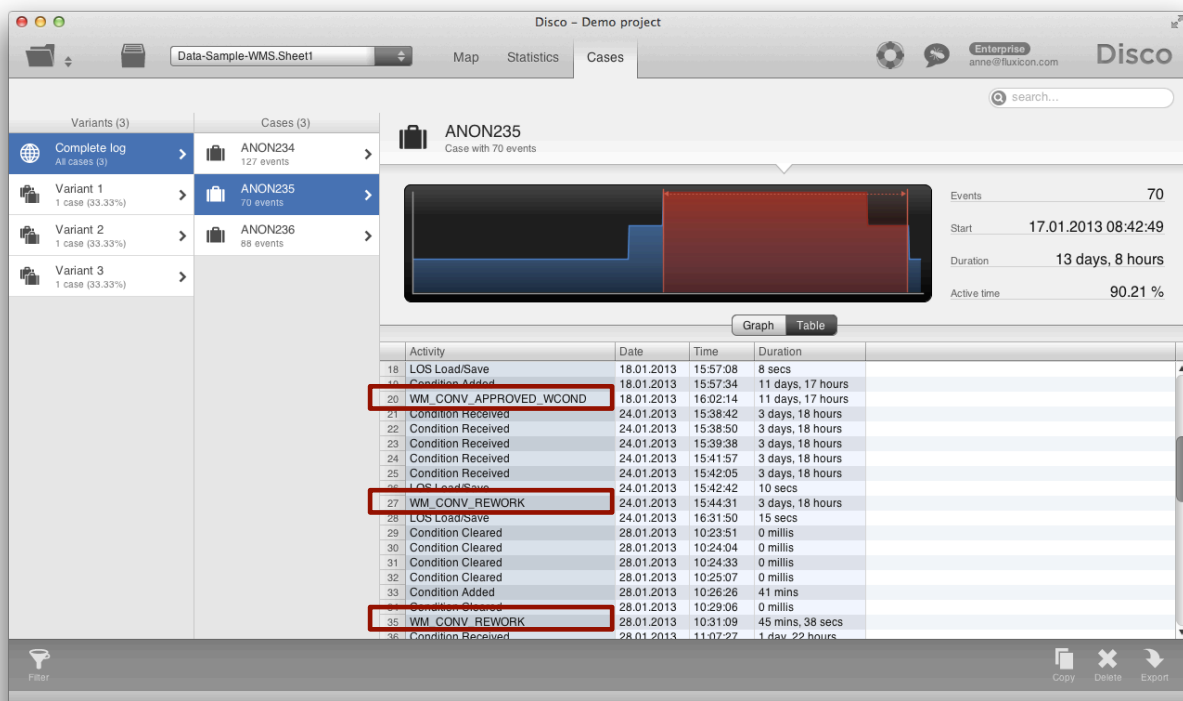


9) Focusing on Milestone Activities

Finally, the last strategy to leave out details is the reverse of the “spider” activity strategy before: Instead of starting from the complete set of events in your data set and looking at where you might leave some out, take a critical look at the different types of events in your data and ask yourself which activities you want to focus on.

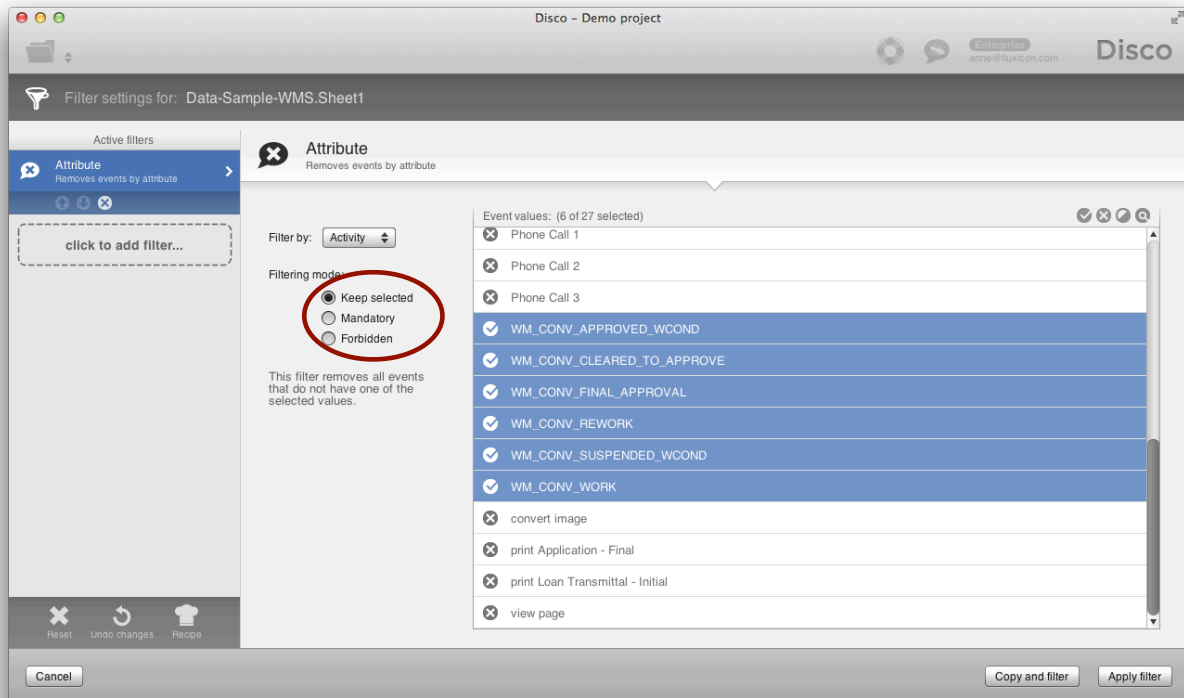
Just because all these different events are contained in your data set does not mean that they are all equally important. Often the activities that you get in your log are on different levels of abstraction. Especially when you have a large number of different activities, it can make sense to start by focusing on just a handful of these activities — the most important milestone activities — initially.

For example, in the anonymized data sample below you see a case with many events and detailed activities such as ‘Load/Save’ and ‘Condition received’. But there are also some other activities that look different (for example, ‘WM_CONV_REWORK’), which are workflow status changes in the process.

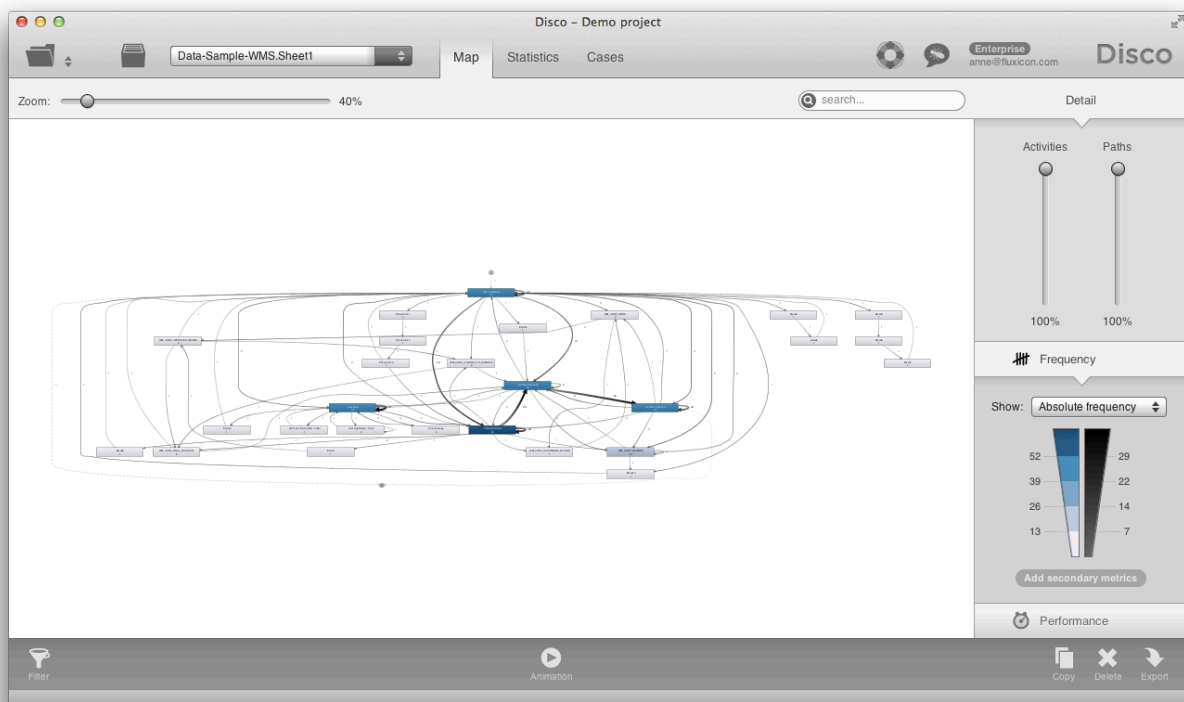


It makes a lot of sense to start by filtering only these ‘WM_’ activities to get started with the analysis and then to bring back more of the detailed steps in between where needed.

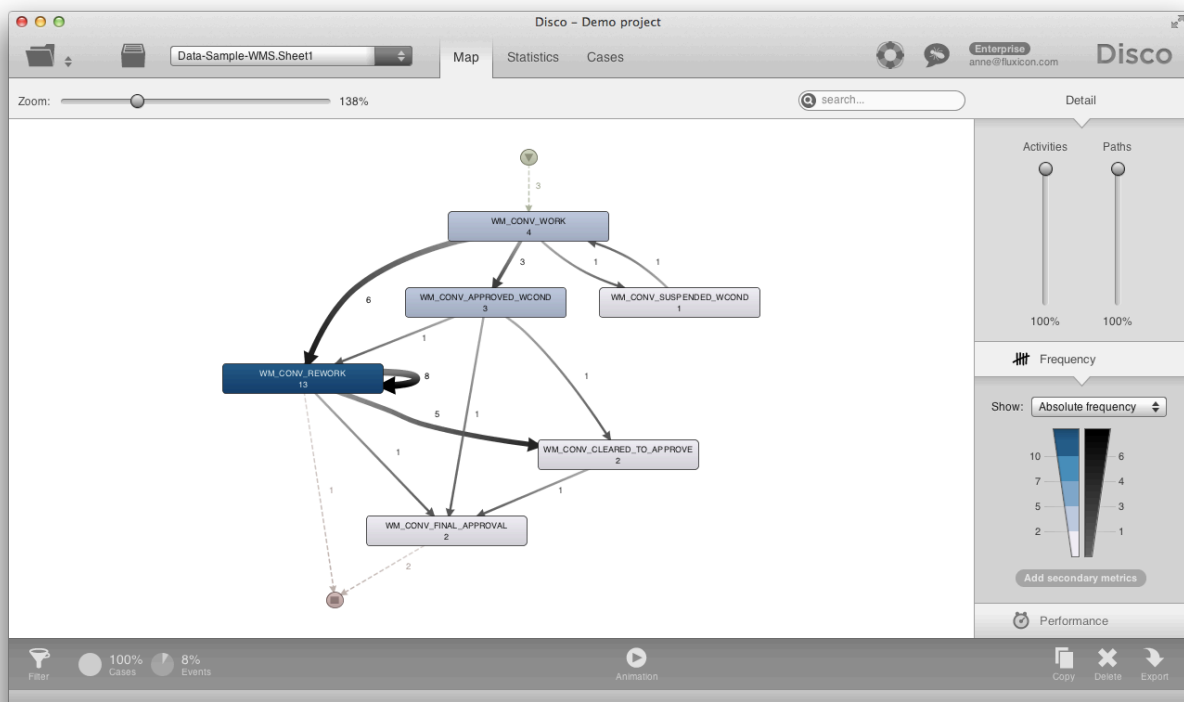
In Disco, you can use the Attribute filter in Keep selected mode as before, but you would deselect all values first and then select just the ones you want to keep (see next page).



As a result, a complex process map with many different activities ...



... can be quickly simplified to showing the process flow for the selected milestone activities for all cases.



If you have no idea what the best milestone activities in your process are, you should sit together with a process or data expert and walk through some example cases with them. They might not know the meaning of every single status change, but with their domain knowledge they are typically able to quickly pick out the milestone events that you need to get started.

It can also be a good idea to start the other way around: Ask your domain expert to draw up the process with only the most important 5 or 7 steps. This can be just on a piece of paper or a white board and will show you what *they* see as the milestone activities in their process from a business perspective. Then go back to your data and see to which extent you can find events that get close to these milestones.

Focusing on milestone activities is a great way to bridge the gap between business and IT and can help you to get started quickly also for very complex processes and extensive data sets.

We hope this was useful and you could pick up a trick or two. Let us know which other methods you have used to simplify your “spaghetti” maps!

References

- [1] Christian W. Günther. [Process Mining in Flexible Environments](#), PhD Thesis, Eindhoven, 2009.