

# Process Mining Analysis Templates

## Automated Process Discovery



Leading-edge, open-source process mining

[info@apromore.com](mailto:info@apromore.com) | [apromore.com](http://apromore.com)

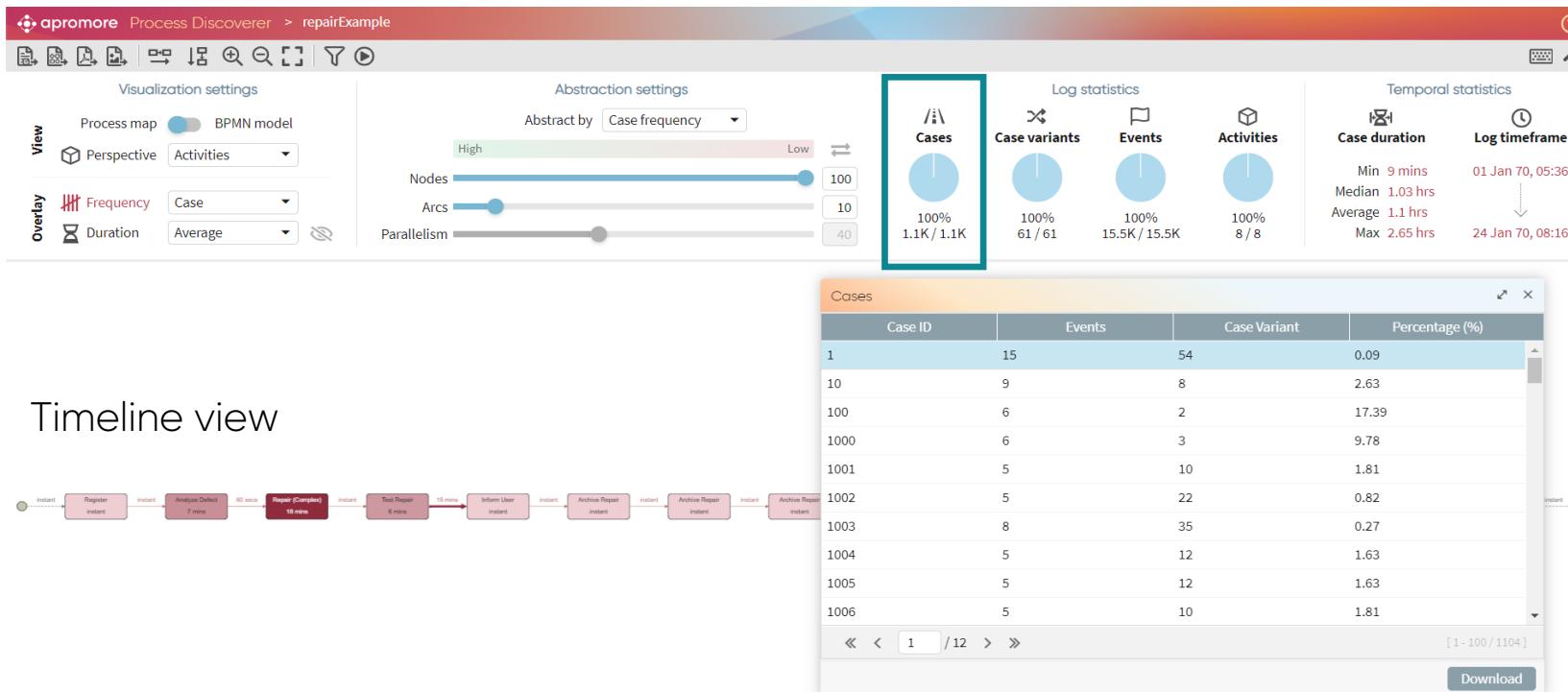
# Automated Process Discovery: Analysis Template

What?	How?
<b>Flow Analysis</b> <ul style="list-style-type: none"><li>Analyze the process structure &amp; main case variants</li><li>Identify parallelism, branching points &amp; rework loops</li><li>Analyze case entry and exit points, and check for incomplete cases</li></ul>	<ul style="list-style-type: none"><li>Visualize the most frequent case variants using Case variant filtering and the Case inspector</li><li>Use abstraction sliders to focus on the most frequent activities and dependencies.</li><li>Switch to BPMN view; inspect the behavior around the gateways; use frequency = max and filtering to disambiguate short loops from parallelism</li><li>Check the dashed lines in the process map (arcs emanating from the start event or leading to the end event). Unexpected dashed lines indicate some cases are incomplete</li></ul>
<b>Filtered Flow Analysis</b> Analyze different slices of the event log (groups of cases) or fragments (groups of events) separately	<ul style="list-style-type: none"><li>Use Case filtering to retain/remove specific cases, e.g. cases that have a specific path of activities</li><li>Use Event filtering to retain/remove subsets of activities, e.g. separate automated or non-core activities, or separate activities belonging to different systems.</li></ul>
<b>Frequency analysis</b> Analyze the most frequent activities and transitions	<ul style="list-style-type: none"><li>Use the color-coding in the process map to find most frequent activities and transitions; use the Activity inspector</li><li>Consider switching between multiple frequency metrics (max, avg, ...)</li><li>Use the Activities tab in the Dashboard plugin</li></ul>
<b>Handoff analysis</b> Analyze handoffs between workers, teams, groups, org units	<ul style="list-style-type: none"><li>Switch between perspectives in the Process Discovery plugin and assess handoffs between resources</li><li>Use the Resources and Other Attributes tabs in the Dashboard plugin</li></ul>

# Flow Analysis

In the Process Discoverer, you can visualize cases by their most frequent case variants using the Case Inspector (look at the Percentage column). You can also view the activities performed within each individual case via the Timeline view, by clicking on each case from the Case Inspector.

## 1 Analyze the process structure and main case variants



The screenshot shows the apromore Process Discoverer interface with the following sections:

- Visualization settings:** Includes a switch between "Process map" and "BPMN model" (selected), and dropdowns for "Perspective" (Activities) and "Overlay" (Frequency, Case, Duration).
- Abstraction settings:** A slider to "Abstract by Case frequency" with a scale from High to Low, and sliders for "Nodes" (100, 10, 40) and "Arcs" (1.1K / 1.1K).
- Log statistics:** Three pie charts showing 100% distribution across Cases, Case variants, Events, and Activities.
- Temporal statistics:** Statistics for Case duration (Min: 9 mins, Median: 1.03 hrs, Average: 1.1 hrs, Max: 2.65 hrs) and Log timeframe (01 Jan 70, 05:36 to 24 Jan 70, 08:16).
- Cases table:** A detailed table showing case statistics for 1104 cases, including Case ID, Events, Case Variant, and Percentage (%).
- Timeline view:** A process flow diagram showing a sequence of events: Register, Analyse Defect, Report (Complex), Test Repair, Inform User, Archive Repair, and another Archive Repair. Each event is labeled with its duration (e.g., 7 mins, 16 mins, 15 mins, 8 mins, etc.).

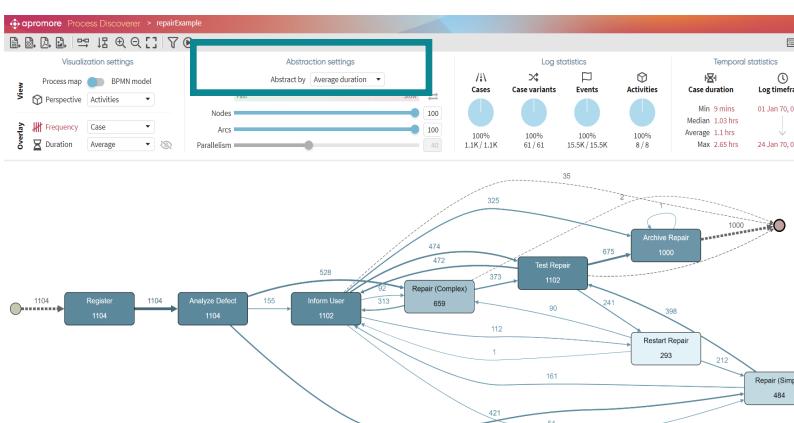
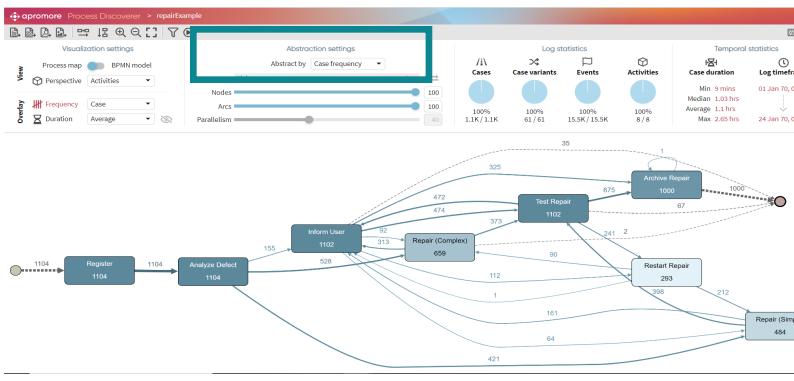


You can also download the case details in a CSV file.

# Flow Analysis

Apromore allows us to abstract a process map by "Case frequency" or "Average Duration".

## 1 Analyze the process structure and main case variants



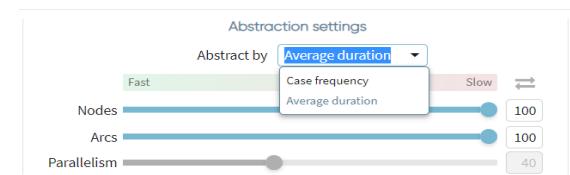
We can adjust the complexity of the discovered map by increasing or decreasing the case frequency or average duration of nodes and arcs that are visualized in our process map.

If we abstract by Case Frequency and shift the arcs or nodes slider towards the left, more edges/nodes with lower case frequency will be removed from the process map.

 Given a node slider setting, moving the arcs slider from High to Low will show only the minimum number of arcs (the most frequent), respectively, the maximum number of arcs, to keep all those nodes connected.

If we abstract by Average Duration and shift the arcs or nodes slider towards the left, more edges/nodes with low average duration will be removed from the process map.

 Given a node slider setting, moving the arcs slider from Fast to Slow will show only the minimum number of arcs (the fastest), respectively, the maximum number of arcs, to keep all those nodes connected.

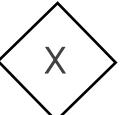
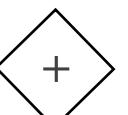
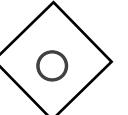


# Flow Analysis

Process activities are not necessarily executed in a sequential order. Sometimes activities are mutually exclusive or are executive in parallel.

## 2 Identify parallelism, branching points and rework loops

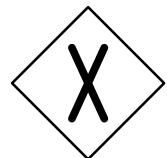
The precise order between activities can be discovered by switching from process map to BPMN process model. In BPMN, order relations are captured via gateways:

-  The XOR Gateway ("Exclusive OR") is used to model branching into exclusive paths [XOR-split] and simple merge between alternative paths [XOR-join].
-  The AND Gateway ("Parallel") is used to model parallel branching [AND-split] and their synchronization [AND-join].
-  The OR Gateway ("Inclusive OR") is used to model branching into one or more parallel paths [AND-split] and synchronizing merge points between one or more parallel paths [OR-join].

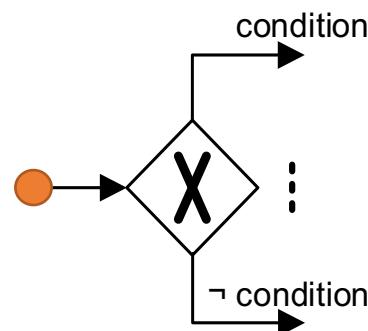
"Parallel branches" means that their relative order doesn't matter. For example if activities A and B are in parallel, this means there are cases where A precedes B and others where B precedes A.



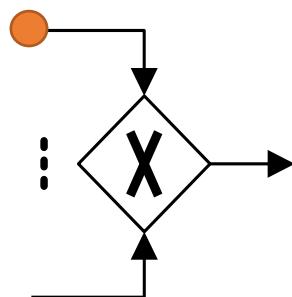
# XOR Gateway



An XOR Gateway captures decision points (XOR-split) and points where alternative flows are merged (XOR-join)

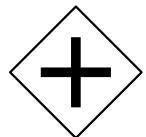


XOR-split → takes one outgoing branch

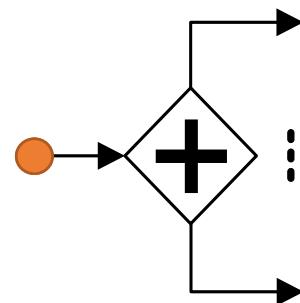


XOR-join → proceeds when one incoming branch has completed

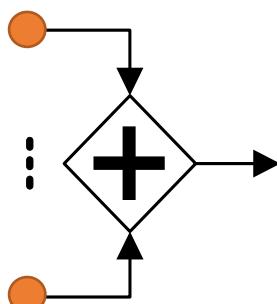
# AND Gateway



An AND Gateway provides a mechanism to create and synchronize "parallel" flows.

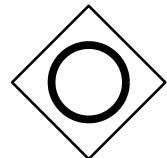


AND-split → takes all outgoing branches

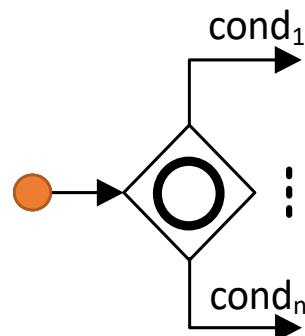


AND-join → proceeds when all incoming branches have completed

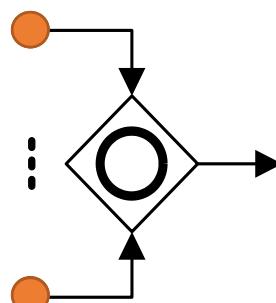
# OR Gateway



An OR Gateway provides a mechanism to create and synchronize n out of m parallel flows.



OR-split → takes one or more branches depending on conditions

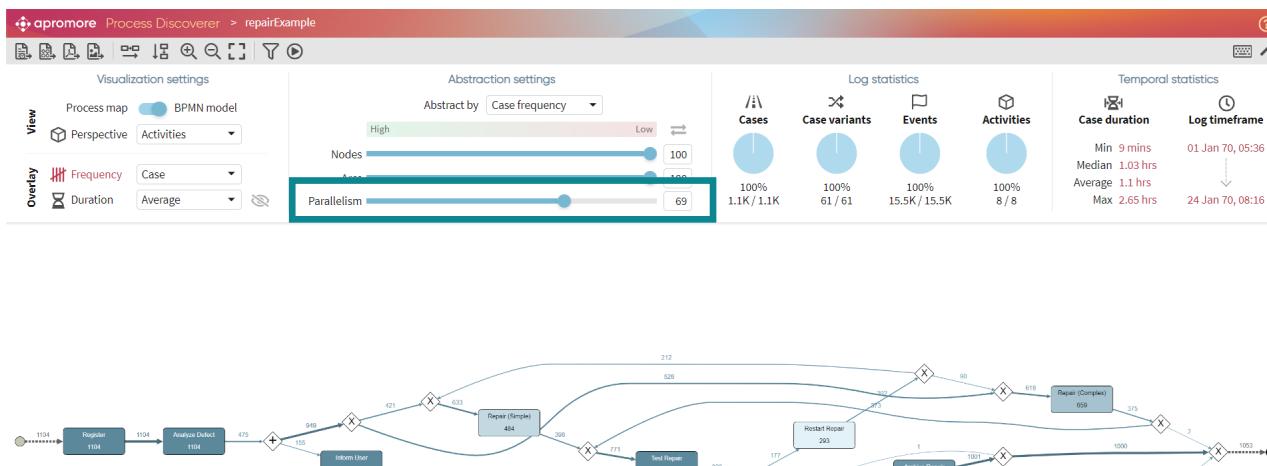


OR-join → proceeds when all active incoming branches have completed

# Flow Analysis

Apmore relies on the Split Miner algorithm to convert process maps into BPMN 2.0 process models. This can be used to display gateways and detect potential parallelism.

## 2 Identify parallelism, branching points and rework loops



The “sensitivity” to parallelism can be adjusted via the dedicated Parallelism slider.

- The lower the slider (left-most), the less tolerant is the algorithm to differences in the frequencies of candidate parallel activities.  
Result: you will get more XOR gateways.
- The higher the slider (right-most), the more tolerant is the algorithm to differences in the frequencies of candidate parallel activities.  
Result: you will get more AND gateways.

# Flow Analysis

Activities may be repeated within the same case, indicating rework.

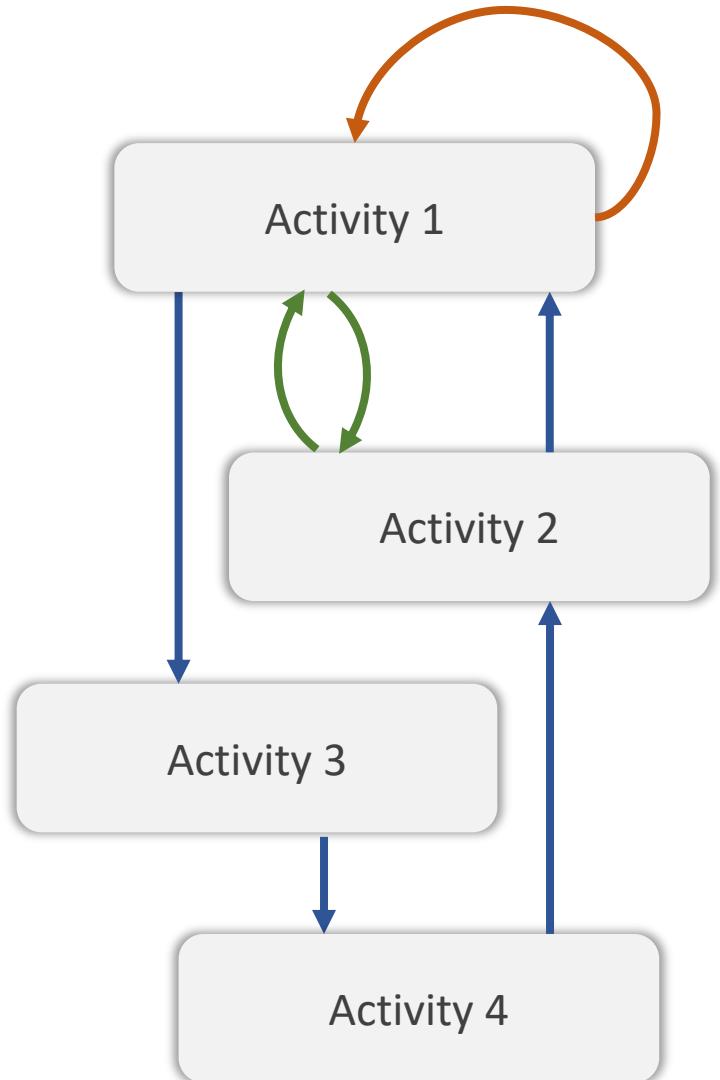
## 2 Identify parallelism, branching points and rework loops

There are 3 forms of rework:

- **Self loop:** an activity takes place n times in a row
- **Short loop ("ping-pong effect"):** two activities are repeated one after the other. Note: this may be confused with parallelism
- **Indirect repetition (loop):** several activities are executed several times in sequence.

To analyze repetitions, one can answer the following questions:

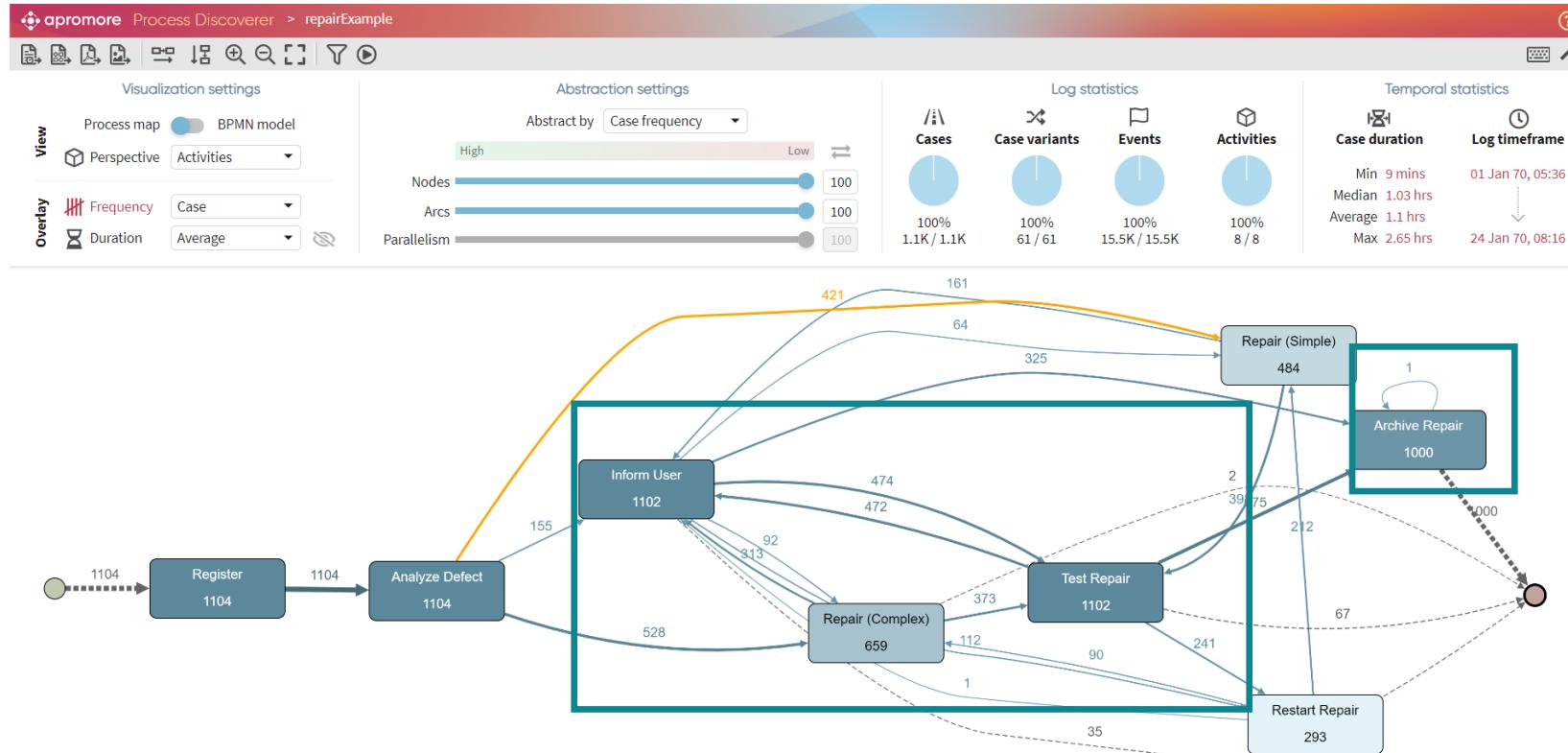
- (1) What type of repetition is present?
- (2) How often is an activity / series of activities repeated?
- (3) How many cases showcase this repetition? How often does it occur?
- (4) Which cases are affected exactly? And how do they differ from cases where there is no repetition?



# Flow Analysis

In some processes, certain activities are repeated – this can be done via self loops, short loops (i.e. "back and forth" between two activities) or indirect repetitions, where a series of activities are repeated.

## 2 Identify parallelism, branching points and rework loops

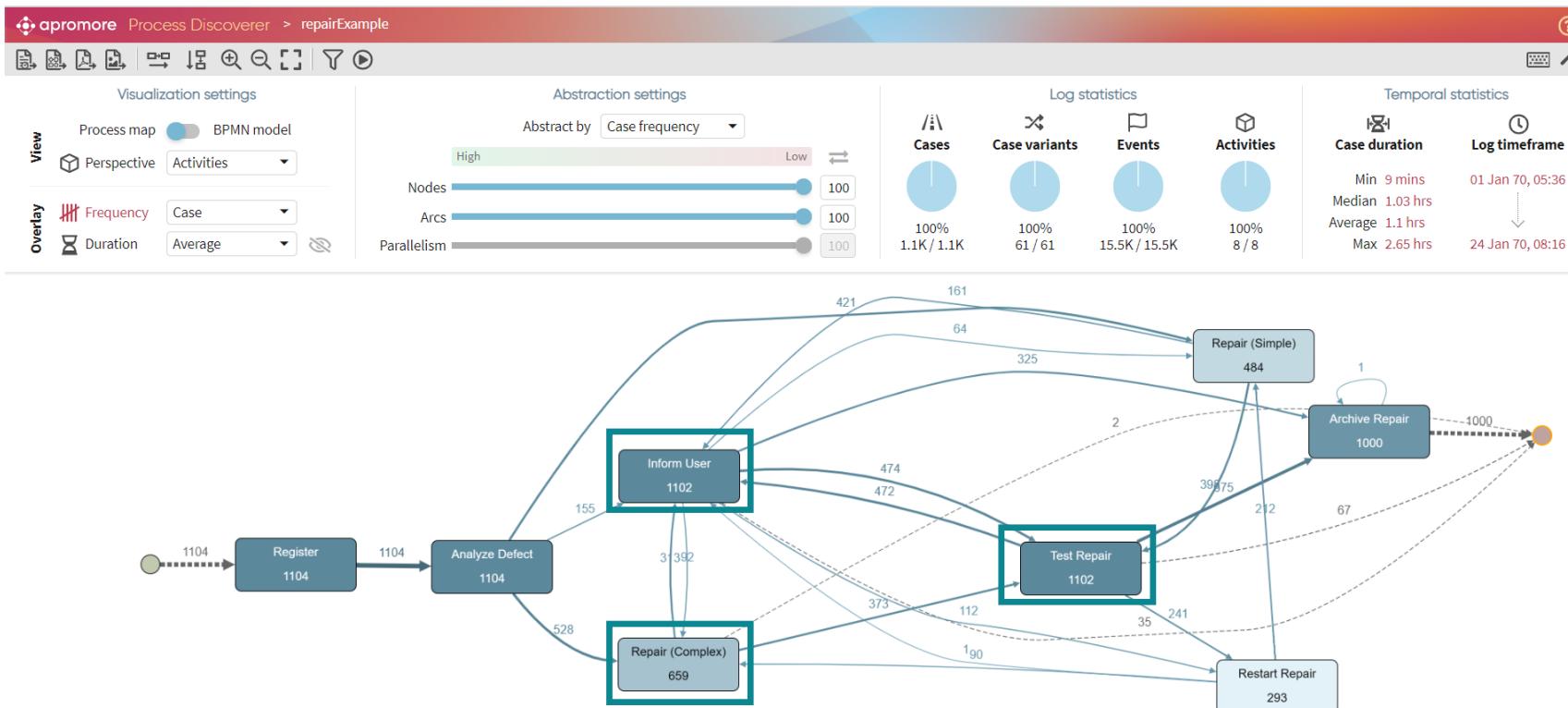


- "Archive Repair" repeats itself once.
- There exists a short-loop between "Inform User" and "Test Repair" but this needs to be verified via frequency overlay/filtering, to make sure this is not parallelism.
- There exist indirect repetitions between "Repair (Complex)/(Simple)", "Test Repair" and "Restart Repair".

# Flow Analysis

Apromore allows us to identify incomplete cases by visualizing dotted arcs leading to the end point (truncated case – we see only the head) or emanating from the start point (truncated case – we see only the tail) in a process map.

## 3 Analyze case entry and exit points and check for incomplete cases

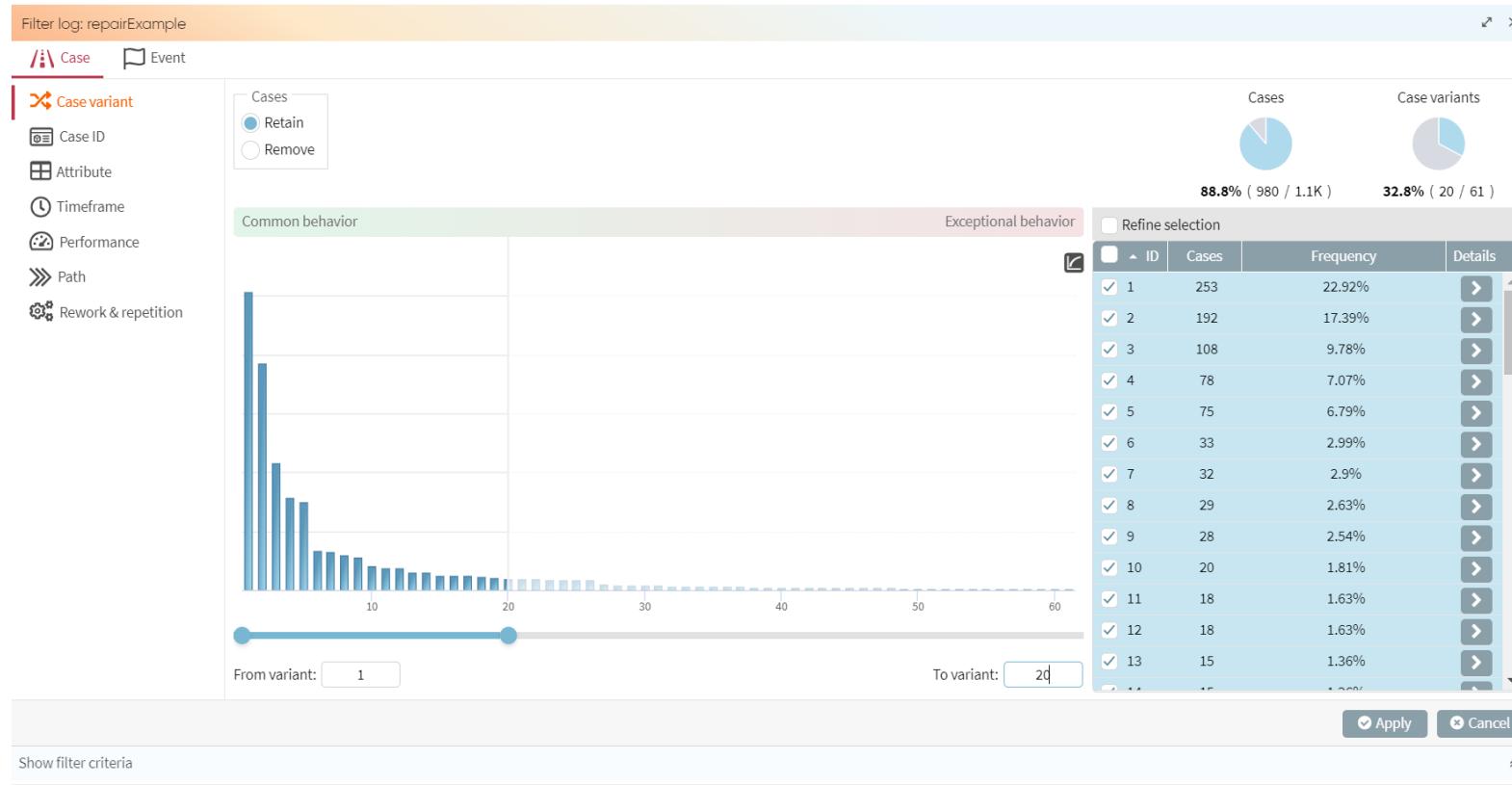


"Inform Users", "Repair (Complex)" and "Test Repair" are spurious end activities, as this process finishes only in "Archive Repair".

# Filtered Flow Analysis

In the Process Discoverer plugin you can also apply filters to analyze the results more precisely and to limit the number of cases. There are a variety of filtering options available to you.

- 1 ➤ Analyze different slices of the event log (sets of cases) or fragments (groups of events) separately



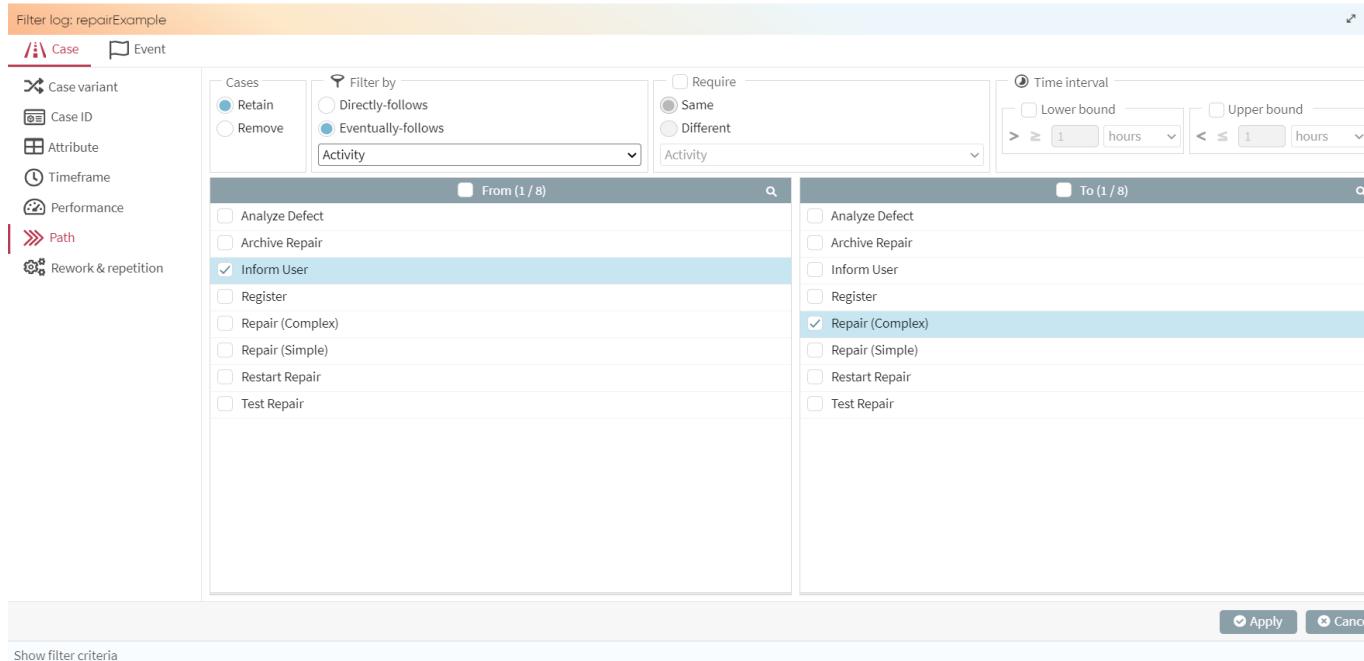
We can make use of the Case > Case Variant filter to retain/remove specific case variants based on their activity sequence or frequency.

For example, we may want to retain up to 20 case variants to analyze the most common process behavior.

# Filtered Flow Analysis

In the Process Discoverer plugin you can also apply filters to analyze the results more precisely and to limit the number of cases. There are a variety of filtering options available to you.

- 1 ➤ Analyze different slices of the event log (sets of cases) or fragments (groups of events) separately



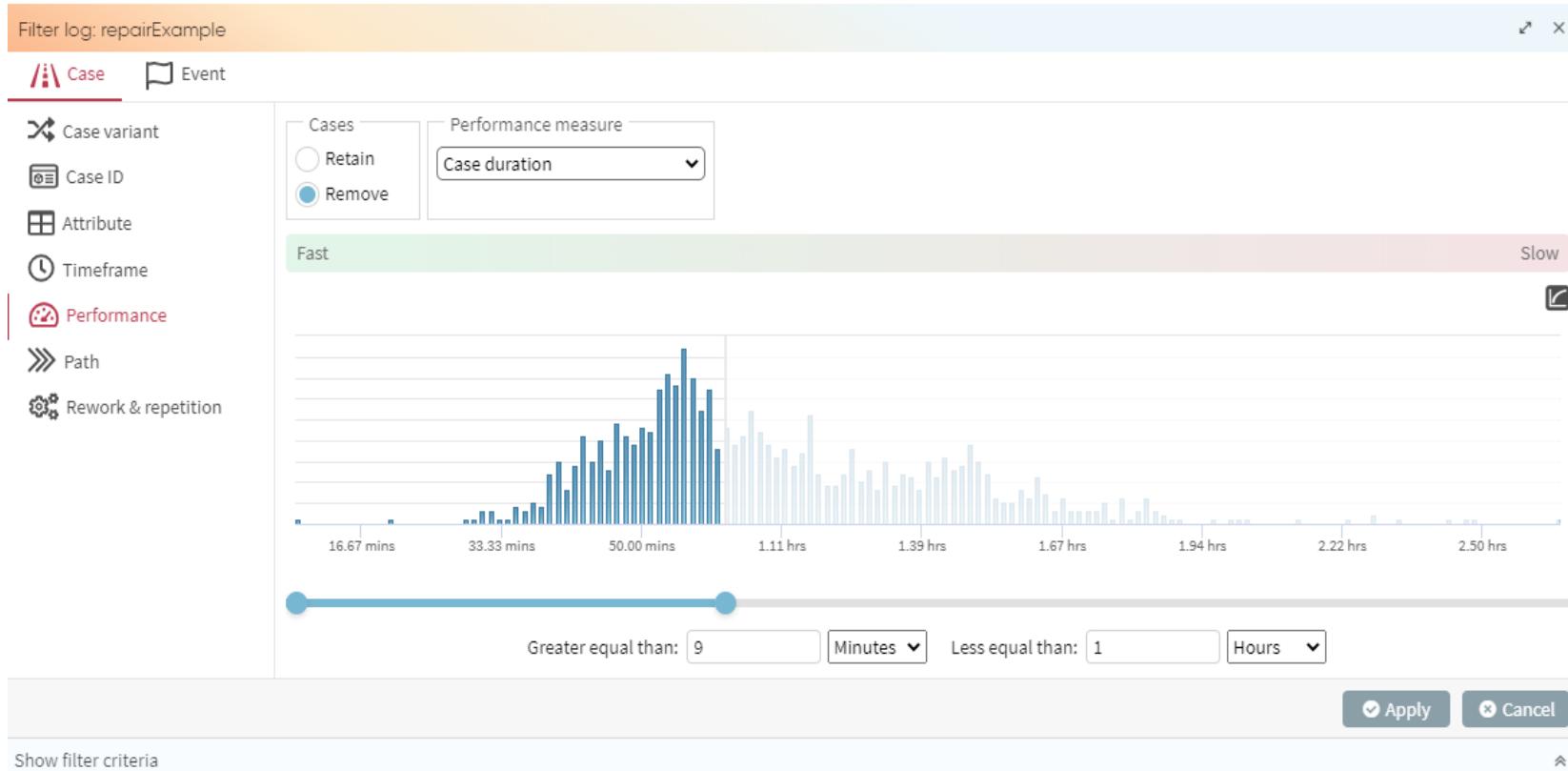
The Case > Path filter can be used to:

- retain/remove cases where there is a pair of events that fulfils a given condition (e.g. "Create invoice" directly or eventually followed by "Approve invoice"). The condition can also involve attributes and time intervals
- retain/remove cases that start with or finish with an event that fulfils a given condition (e.g. all traces that start with "Purchase order received" and finish with "Order cancelled").

# Filtered Flow Analysis

In the Process Discoverer plugin you can also apply filters to analyze the results more precisely and to limit the number of cases. There are a variety of filtering options available to you.

- 1 Analyze different slices of the event log (sets of cases) or fragments (groups of events) separately



In order to understand a process, we might want to identify outliers. We can use the Case > Performance filter for this purpose.

This filter allows us to retain/remove cases that have a duration (or other performance measure) above or below a given value (e.g. all cases that take more than 5 days).

# Filtered Flow Analysis

In the Process Discoverer plugin you can also apply filters to analyze the results more precisely and to limit the number of cases. There are a variety of filtering options available to you.

- 1 ➤ Analyze different slices of the event log (sets of cases) or fragments (groups of events) separately

The screenshot shows the 'Filter log' dialog for a log named 'repairExample'. The 'Case' tab is selected. Under 'Containing', 'Event attribute' is chosen, and 'Repair (Simple)' is checked. A preview table shows the following data:

Value (5 / 8)	Cases	Frequency
Repair (Simple)	484	7.068%
Inform User	1102	16.092%
Archive Repair	1000	14.603%
Analyze Defect	1104	16.121%
Restart Repair	293	4.279%
Test Repair	659	9.623%
Register	1104	16.121%
Repair (Complex)		

At the bottom, there are 'Apply' and 'Cancel' buttons, and a link 'Show filter criteria'.

We can use the Case > Attribute filter to retain/remove cases that fulfil a given condition (e.g. all cases that have an event "Amend purchase order").

# Filtered Flow Analysis

In the Process Discoverer plugin you can also apply filters to analyze the results more precisely and to limit the number of cases. There are a variety of filtering options available to you.

- Analyze different slices of the event log (sets of cases) or fragments (groups of events) separately

The screenshot shows the 'Event' tab of the filter dialog. The table lists events with their values and frequencies:

Value	Total	Frequency
SolverC1	534	3.448%
SolverC2	514	3.319%
SolverC3	402	2.596%
<b>SolverS1</b>	<b>592</b>	<b>3.823%</b>
<b>SolverS2</b>	<b>498</b>	<b>3.216%</b>
SolverS3	480	3.1%
System	7242	46.765%
Tester1	902	5.825%
Tester2	904	5.838%
Tester3	910	5.876%
Tester4	788	5.088%
Tester5	844	5.45%
Tester6	876	5.657%

At the bottom are 'Apply' and 'Cancel' buttons.

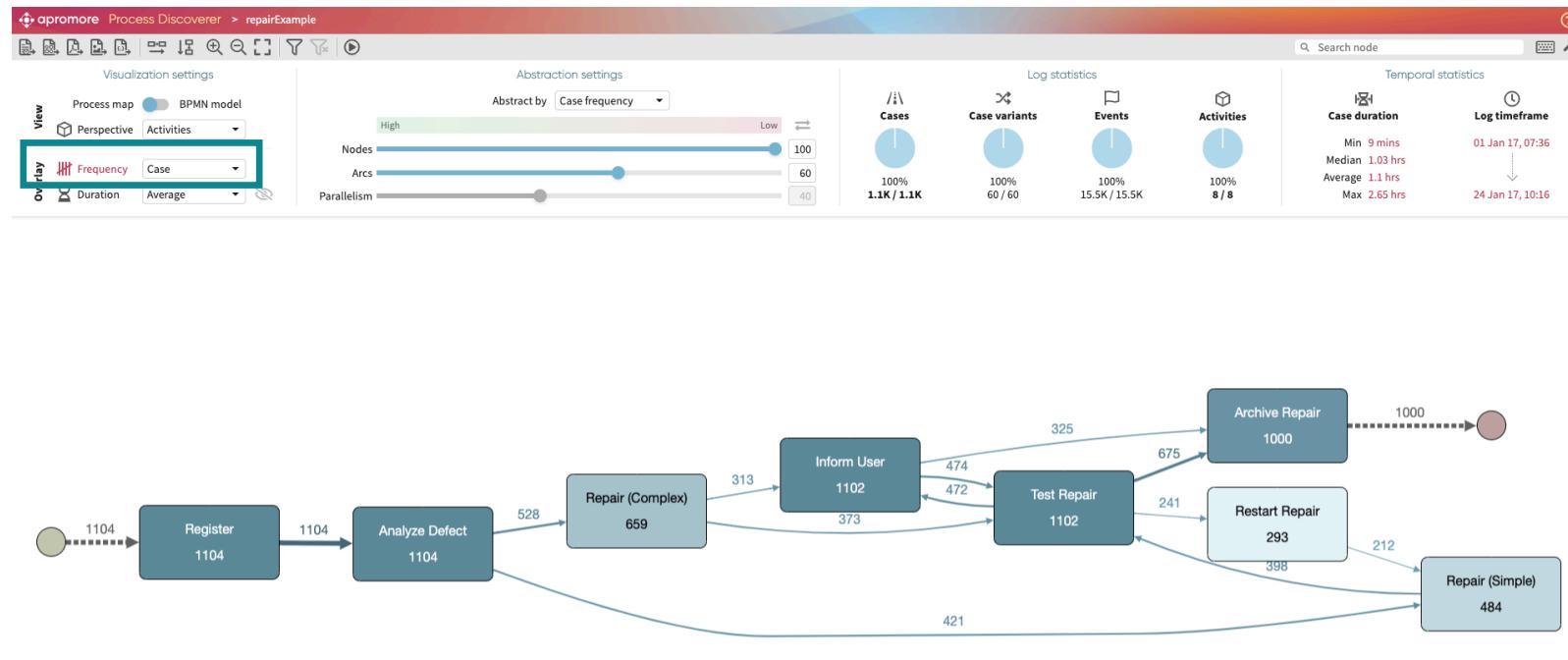
Instead of “slicing” the log into groups of cases, we can “dice” the log to focus on specific fragments. We can achieve so by using an Event > Attribute filter.

For example, we can use such a filter to remove those events where the value of “Resource” is “Solver S1” and “Solver S2”. The tool will automatically reconstruct the flow after removing those events, so that the process map/BPMN model is fully connected from the start to the end point.

# Frequency Analysis

Apromore can visualize the process map by using color-coding techniques to find the most frequent activities and transitions.

## 1 Analyze the most frequent activities and transitions



Frequency statistics are visualized via labels on activities and arcs, as well as via colours and line thickness (on an ocean blue scale) for nodes and arcs.

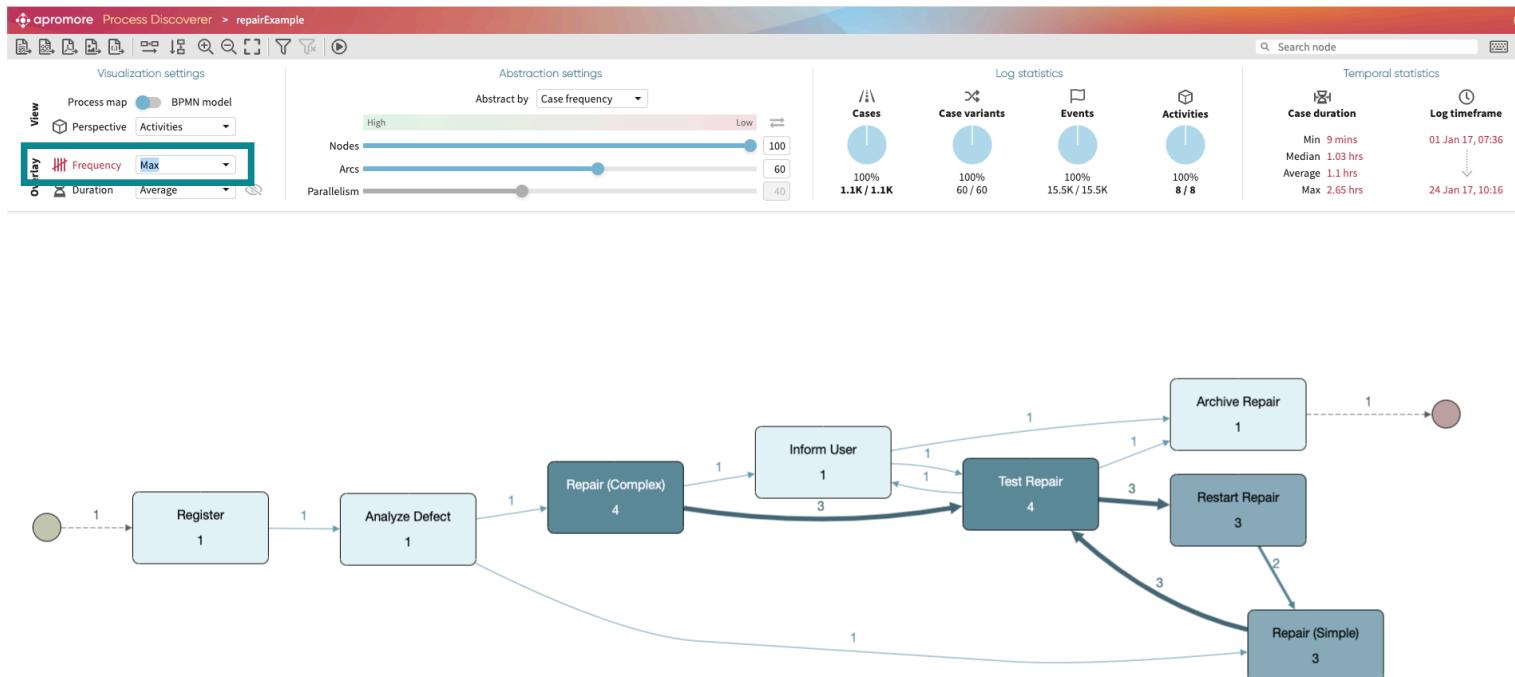
The transition between "Register" and "Analyze Defect" is the most frequent, as it is observed in all 1104 cases of the log.

# Frequency Analysis

Apromore can visualize the process map by using color – coding techniques to find the most frequent arcs.

1

Analyze the most frequent activities and relations



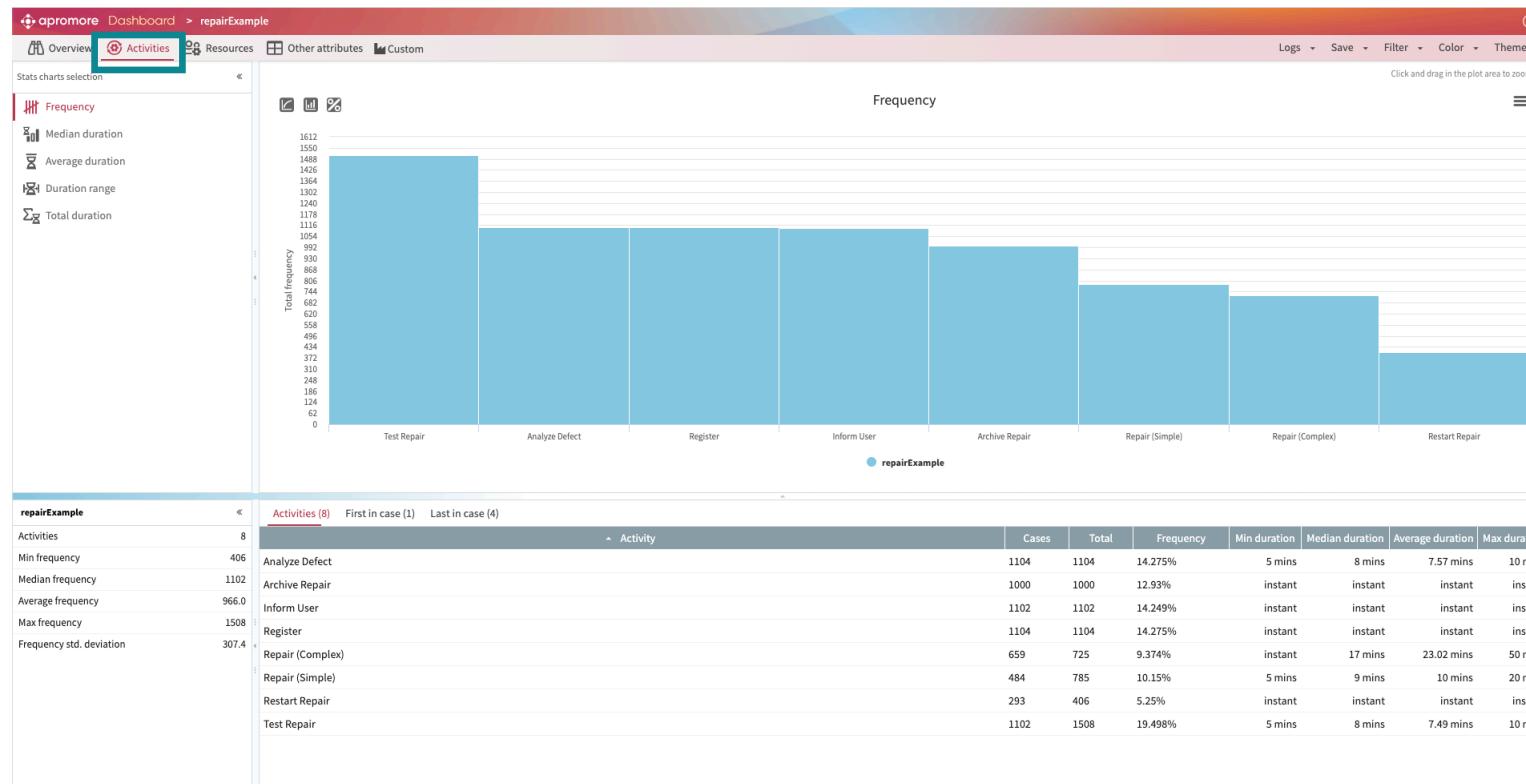
By selecting the "Max" ("Min") option from the "Frequency" drop-down menu, we can see the maximum (minimum) number of cases activities and transitions have been observed in.

For example, there is at least one case where activity "Repair (Complex)" was performed four times.

# Frequency Analysis

The Activities tab in the Dashboard plugin offers a complementary perspective by showing the frequency of each resource in terms of activities they worked on.

1 ➤ Analyze the most frequent activities and relations



# Handoff Analysis

Poor handovers can affect, among others, process efficiency. Delays due to the transfer of work between resources can be identified with the handoff analysis.



A slow handoff may be due to different reasons:

- Inadequate handoff / missing information
- No notification that an activity is completed
- Too busy resource (resource-capacity bottleneck)

**Handoff analysis in Apromore**

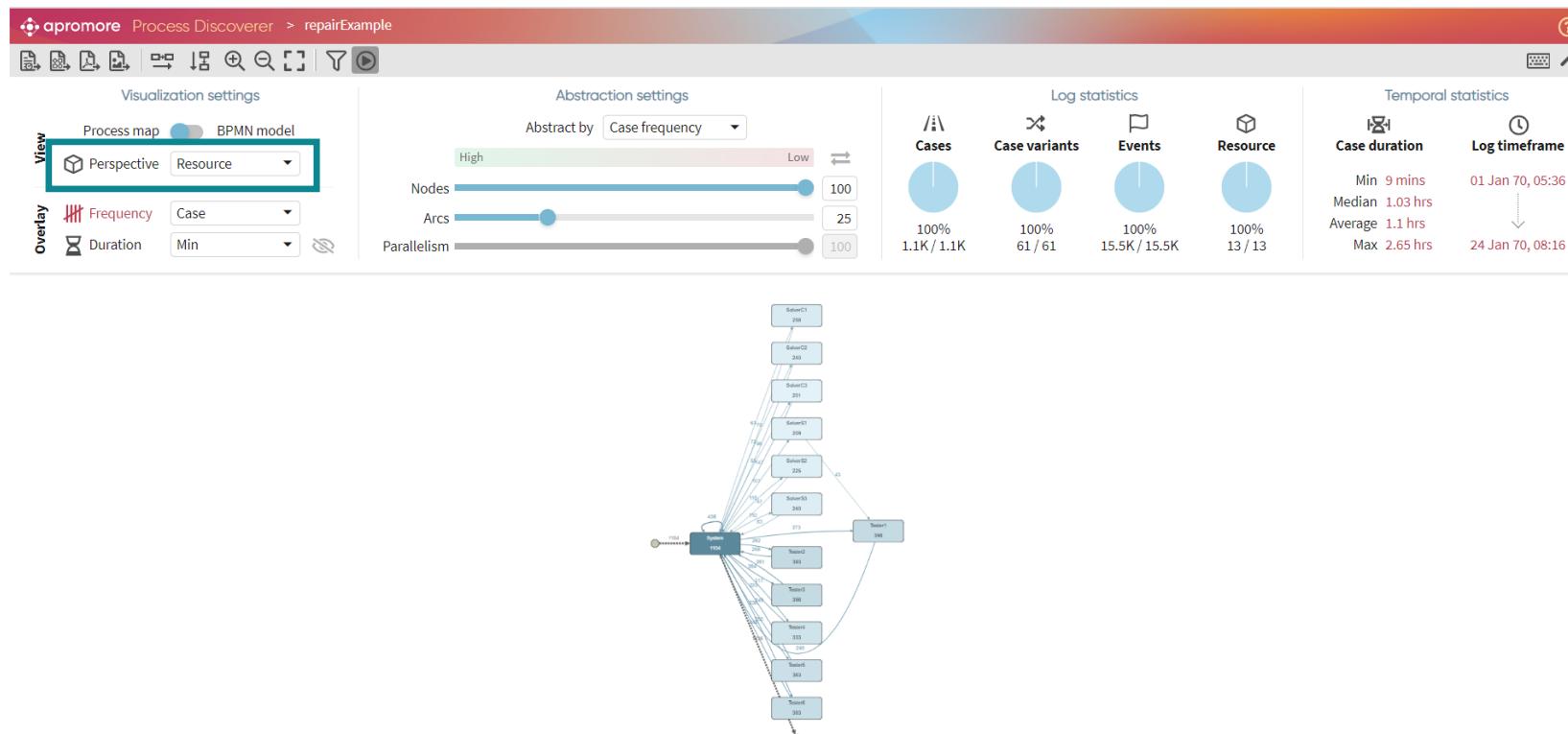
A Case > Path filter can be used to identify handoff inefficiencies. Select Require Different Resource to display only those cases where different resources are involved (handoffs occur). To analyze these cases in greater detail, further filters can be applied.

# Handoff Analysis

Apromore allows switching between different process perspectives in the Process Discoverer.

1

Analyze handoffs between workers, teams, groups and organizational units



By changing the perspective to "Resource", we can analyse the handoff between resources.

# Handoff Analysis

Path filter can be used to deepen the handoff analysis.

## 1 Analyze handoffs between workers, teams, groups and organizational units

Filter log: repairExample

Case  Event

Cases  Case ID  Attribute  Timeframe  Performance  Path  Rework & repetition

Filter by: Directly-follows  Eventually-follows

Require  Same  Different  "Resource"

Time interval  Lower bound  $> \geq 1$  hours  Upper bound  $< \leq 1$  hours

From (1 / 8) To (1 / 8)

<input type="checkbox"/> Analyze Defect	<input type="checkbox"/> Analyze Defect
<input type="checkbox"/> Archive Repair	<input type="checkbox"/> Archive Repair
<input type="checkbox"/> Inform User	<input type="checkbox"/> Inform User
<input type="checkbox"/> Register	<input type="checkbox"/> Register
<input checked="" type="checkbox"/> Repair (Complex)	<input type="checkbox"/> Repair (Complex)
<input type="checkbox"/> Repair (Simple)	<input type="checkbox"/> Repair (Simple)
<input type="checkbox"/> Restart Repair	<input type="checkbox"/> Restart Repair
<input type="checkbox"/> Test Repair	<input checked="" type="checkbox"/> Test Repair

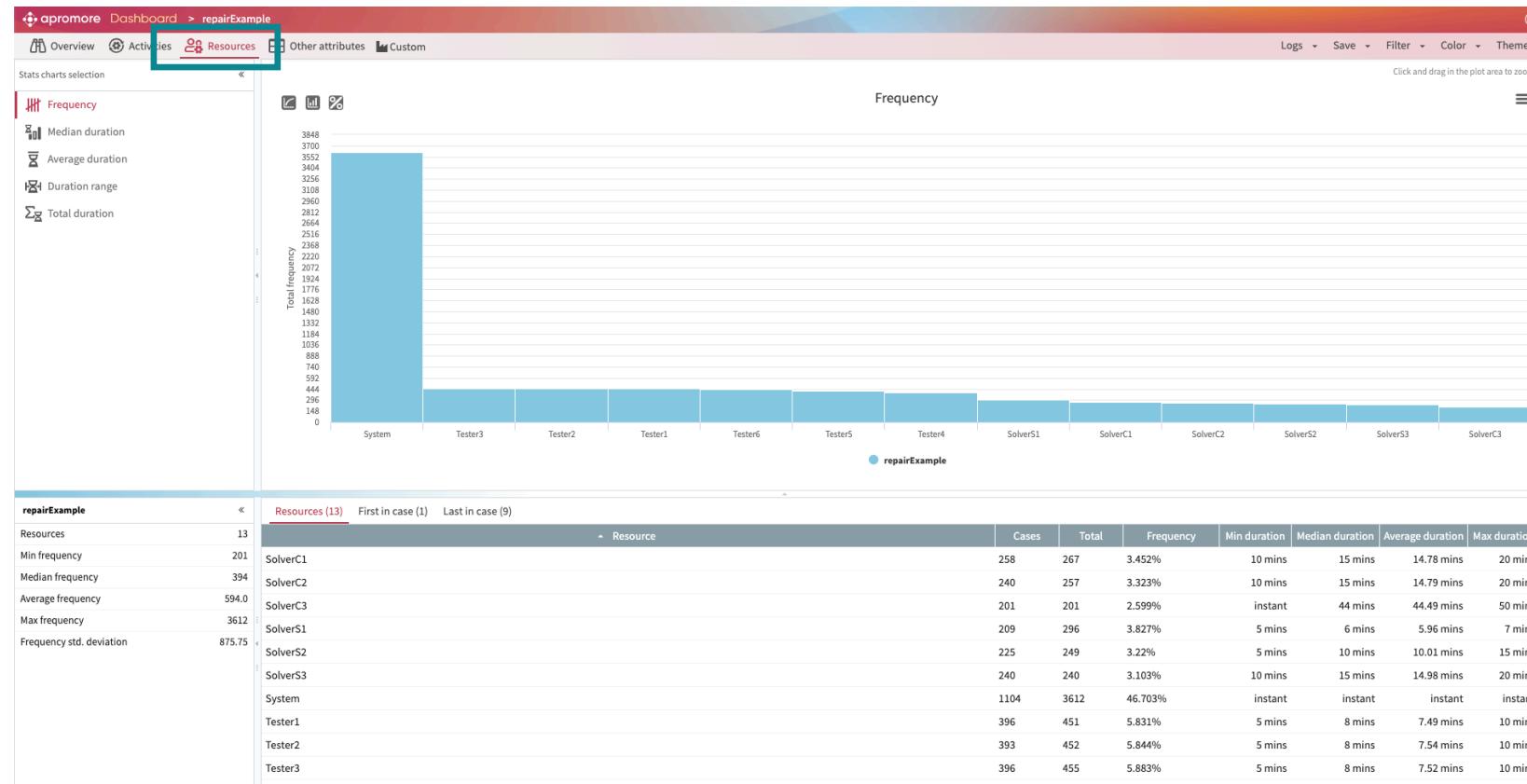
Show filter criteria

Handoff analysis between resources can also be identified by using a Case > Path filter where two activities (linked via a directly-follows relation) are performed by different resources, e.g. "Repair (Complex)" and "Test Repair".

# Handoff Analysis

The Resources tab in the Dashboard plugin offers a complementary perspective by showing the frequency of each resource in terms of activities they worked on.

1 ➤ Analyze handoffs between workers, teams, groups and organizational units





# apromore

Leading-edge, open-source process mining

[info@apromore.com](mailto:info@apromore.com) | [apromore.com](http://apromore.com)