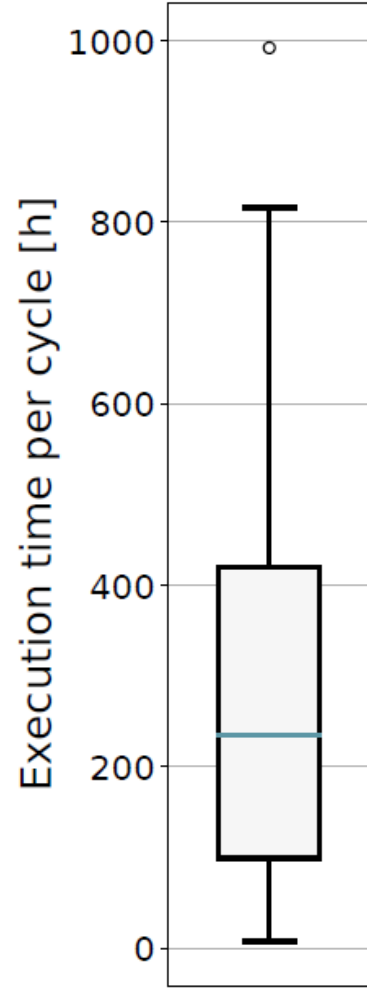
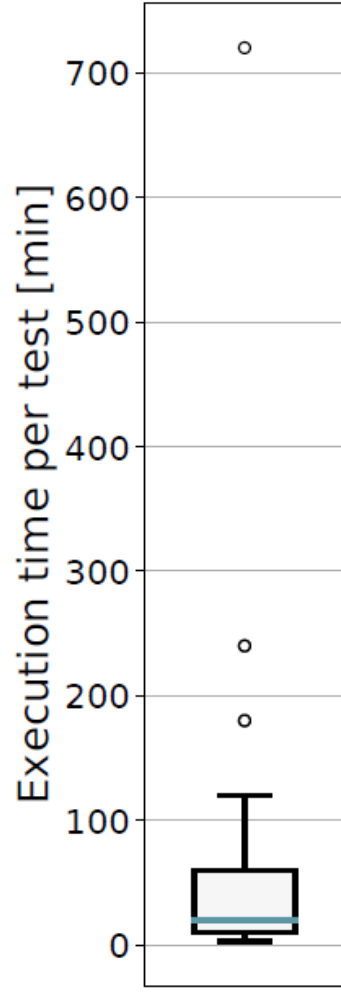
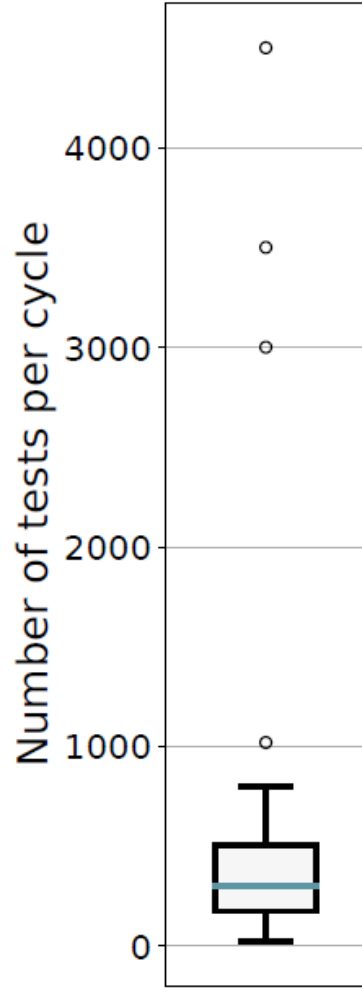
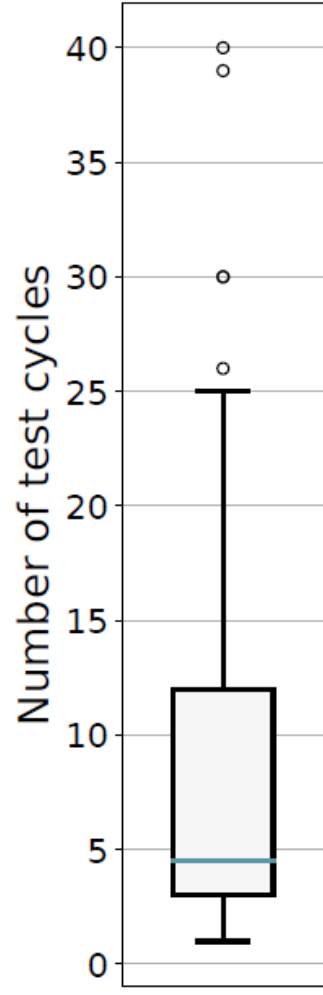
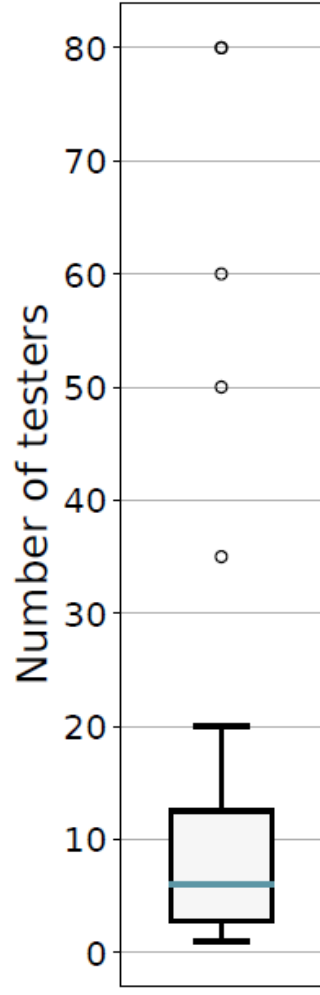
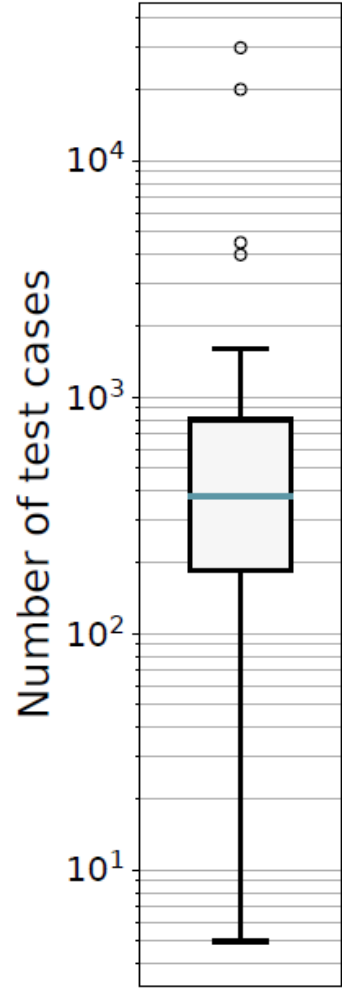
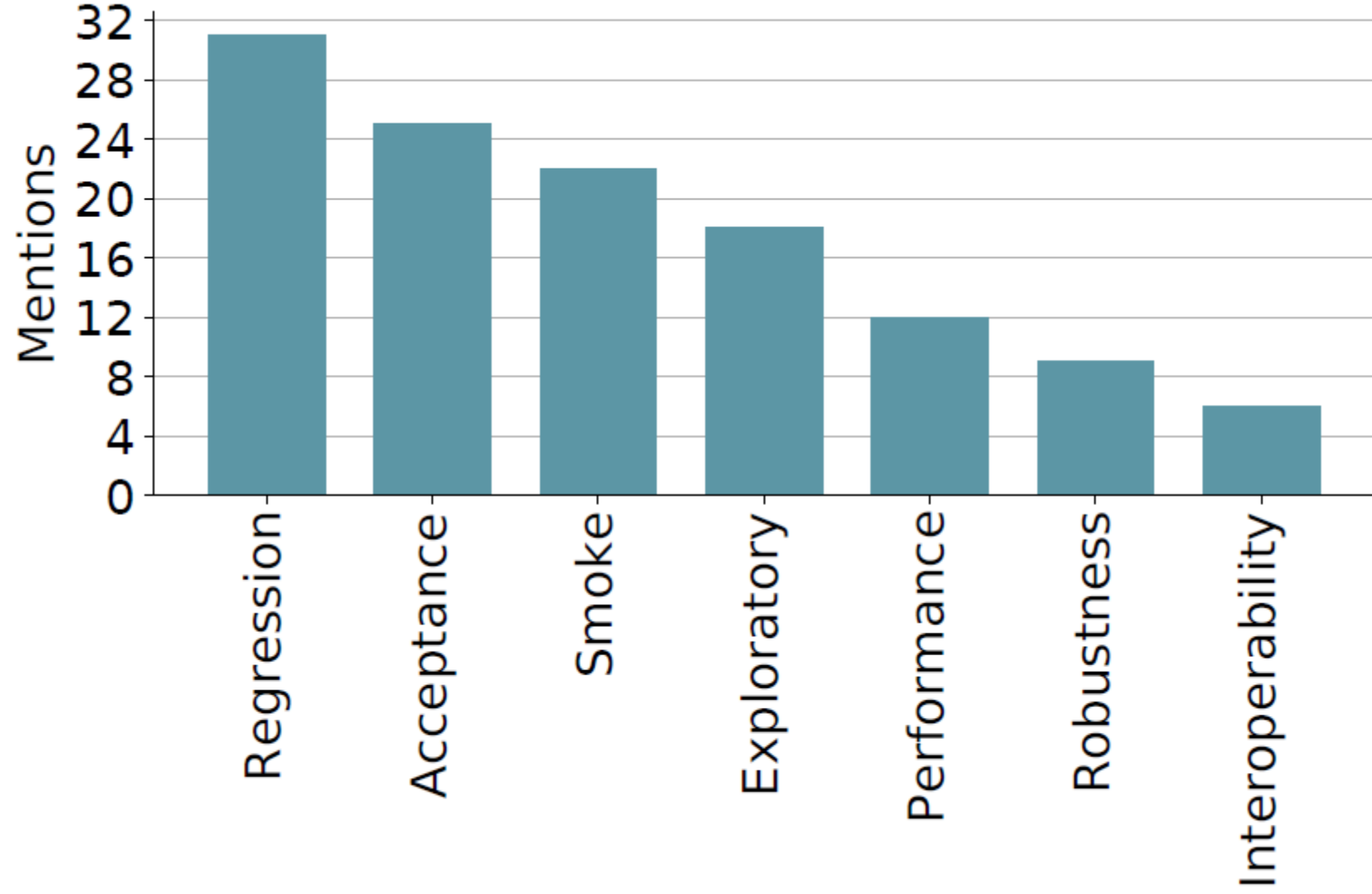
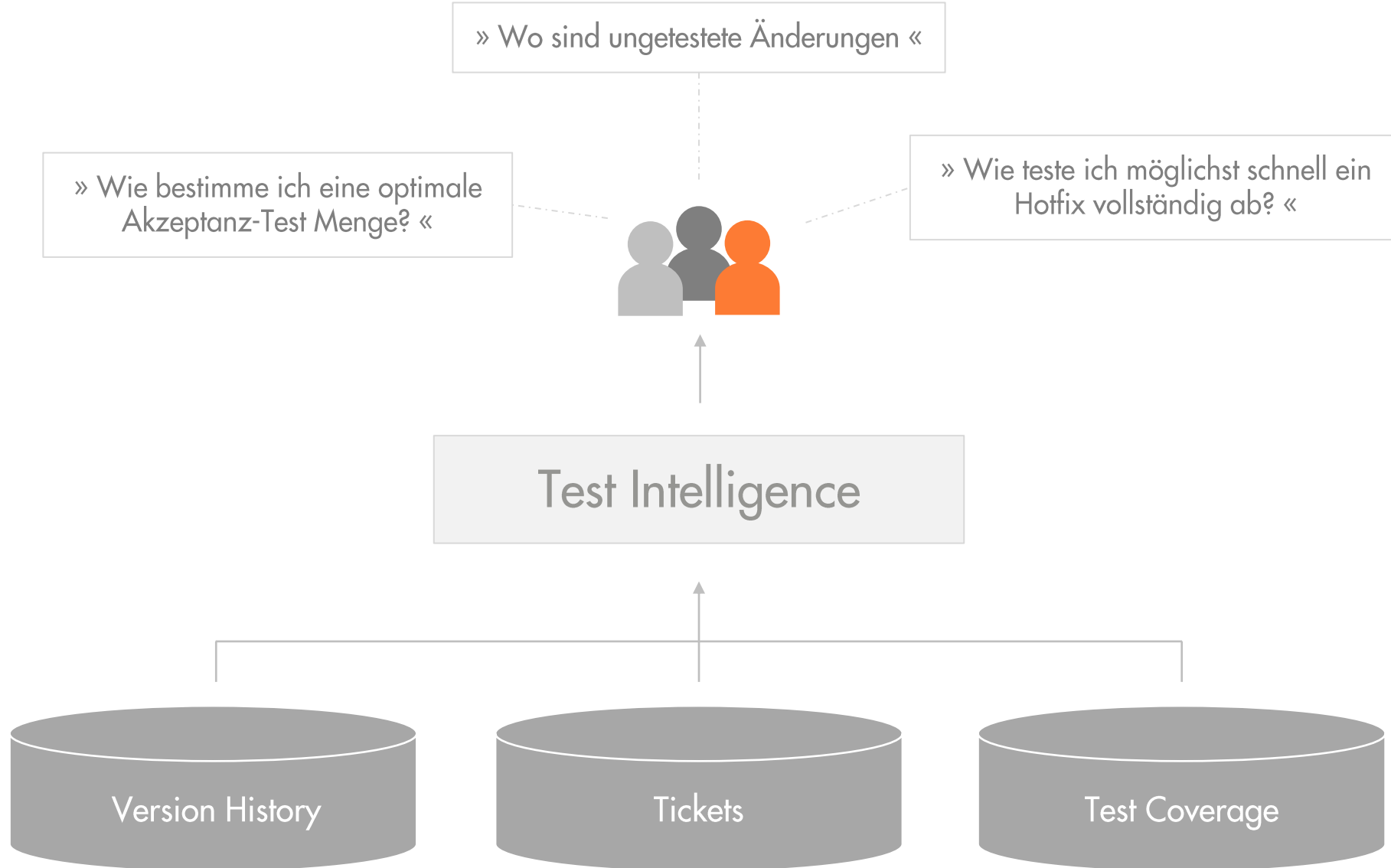


# Test Intelligence für manuelle Tests

Manuelle Tests?





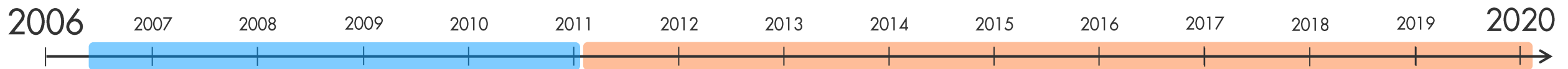


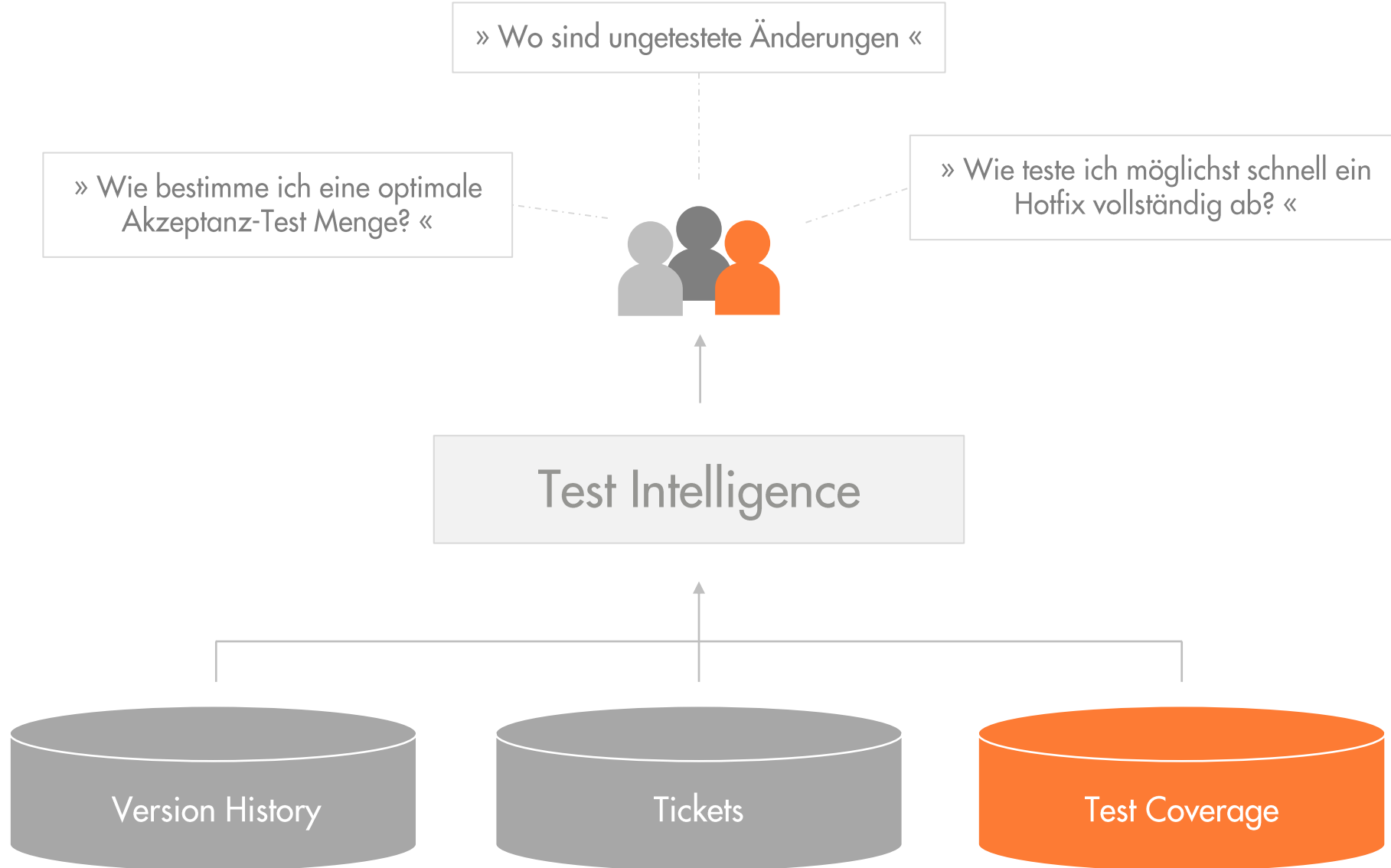


TUM



CQSE





```
114
115i     private static void createAndShowGUI(String[] args) {
116         assert calledOnEDT() : threadInfo();
117
118         Messages.setMsgHandler(new GUIMessageHandler());
119
120         // GlobalKeyboardWatch.showEventsSlowerThan(100, TimeUnit.MILLISECONDS);
121
122         Theme theme = Themes.DEFAULT;
123         // if a LaF was set from the command line, then don't override it
124         if (System.getProperty("swing.defaultlaf") == null) {
125             theme = AppPreferences.loadTheme();
126             Themes.install(theme, false, true);
127         }
128
129         int uiFontSize = AppPreferences.loadUIFontSize();
130         String uiFontType = AppPreferences.loadUIFontType();
131
132         Font defaultFont = UIManager.getFont("defaultFont");
133         if (defaultFont != null) { // if null, we don't know how to set the font
134             if (uiFontSize != 0 || !uiFontType.isEmpty()) {
135                 Font newFont;
136                 if (!uiFontType.isEmpty()) {
137                     newFont = new Font(uiFontType, Font.PLAIN, uiFontSize);
138                 } else {
139                     newFont = defaultFont.deriveFont((float) uiFontSize);
140                 }
141
142                 FontUIResource fontUIResource = new FontUIResource(newFont);
143                 UIManager.put("defaultFont", fontUIResource);
144
145                 if (theme.isNimbus()) {
146                     UIManager.getLookAndFeel().getDefaults().put("defaultFont", fontUIResource);
147                 }
148             }
149         }
150
151         var pw = PixelitorWindow.get();
152         Dialogs.setMainWindowInitialized(true);
153
154         // Just to make 100% sure that at the end of GUI
155         // initialization the focus is not grabbed by
156         // a textfield and the keyboard shortcuts work properly
157         FgBgColors.getGUI().requestFocus();
158
159         TipsOfTheDay.showTips(pw, false);
160
161         MouseZoomMethod.load();
162         PanMethod.load();
163
164         // The IO-intensive preloading of fonts is scheduled
165         // to run after all the files have been opened,
166         // and on the same IO thread
167         openCLFilesAsync(args)
168             .exceptionally(throwable -> null) // recover
169             .thenAcceptAsync(v -> afterStartTestActions(), onEDT)
170             .thenRunAsync(Utils::preloadFontNames, onIOThread)
171             .exceptionally(Messages::showExceptionOnEDT);
172     }
173
```



Sample Only the Active Layer/Mask

Untitled1 x Picture1.png x

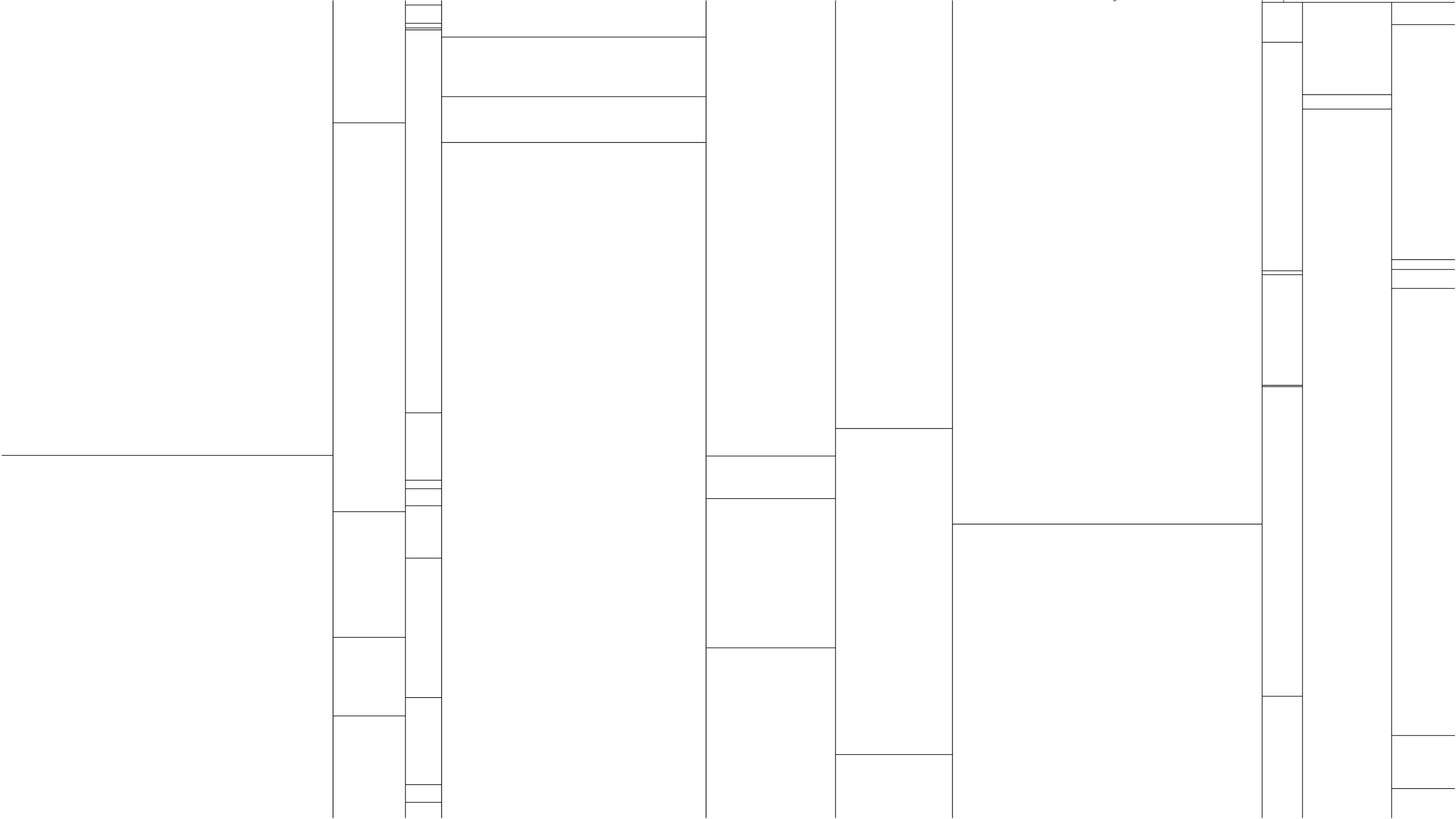


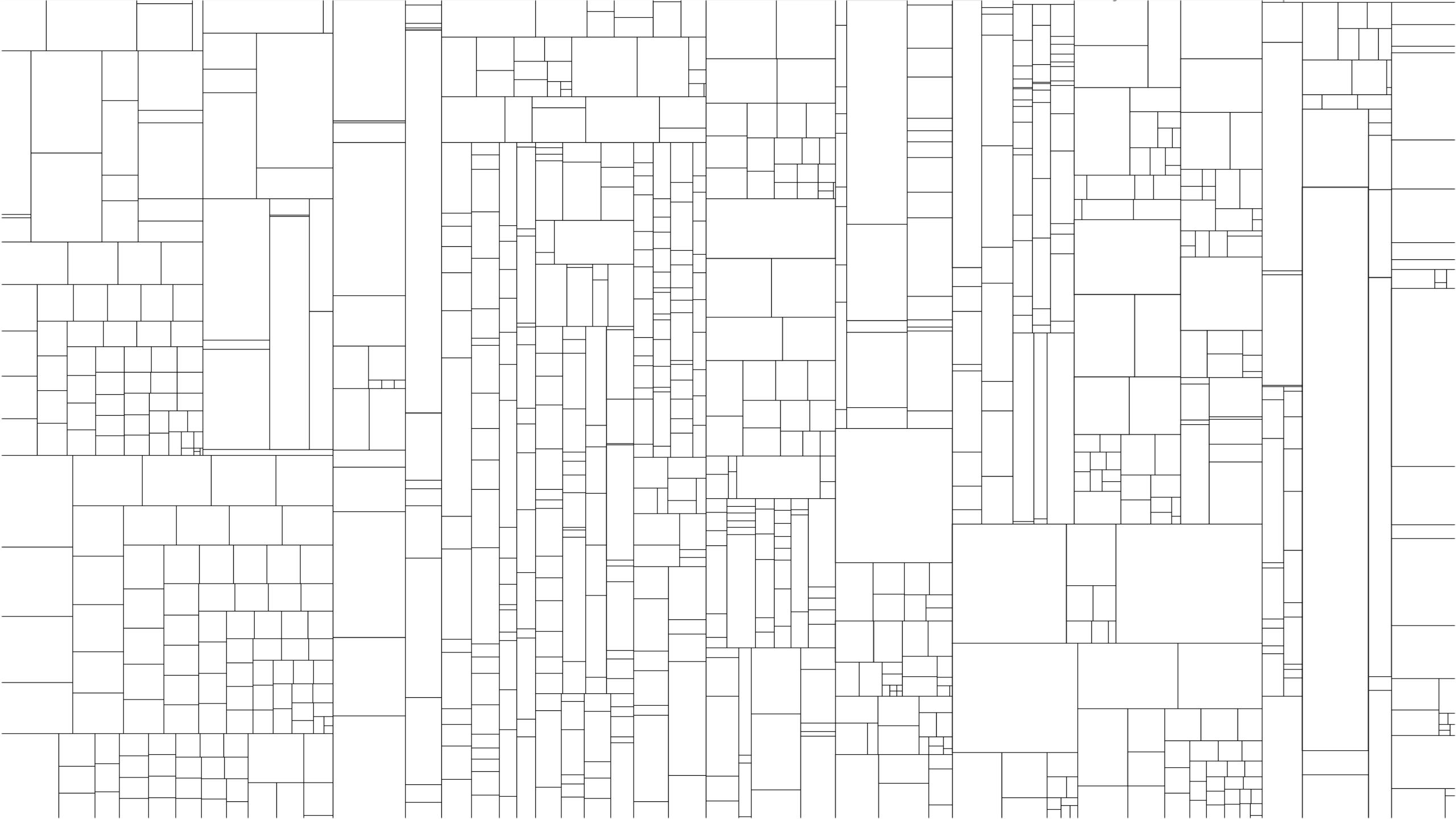
Layers

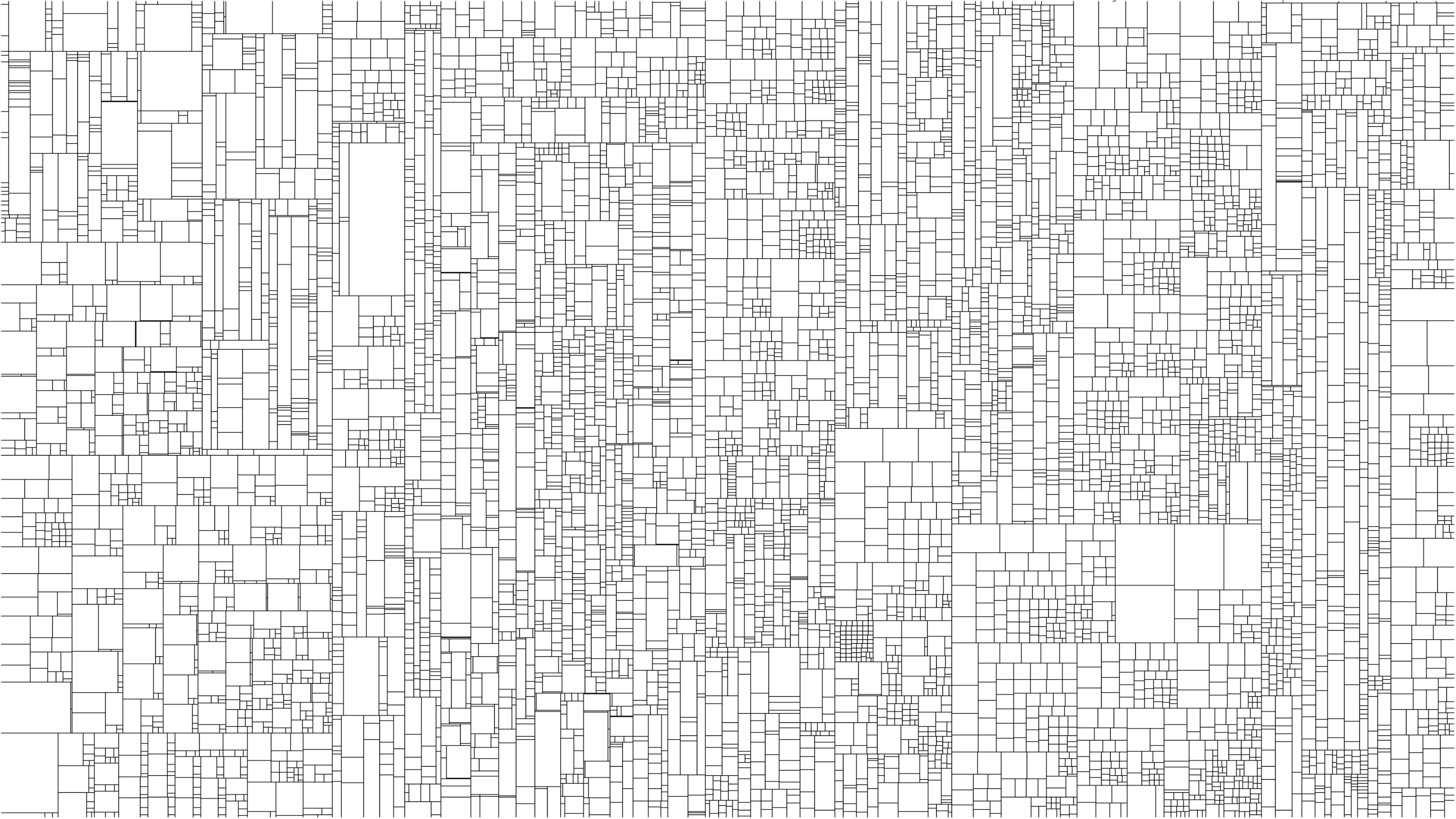
Opacity: 100 % Normal

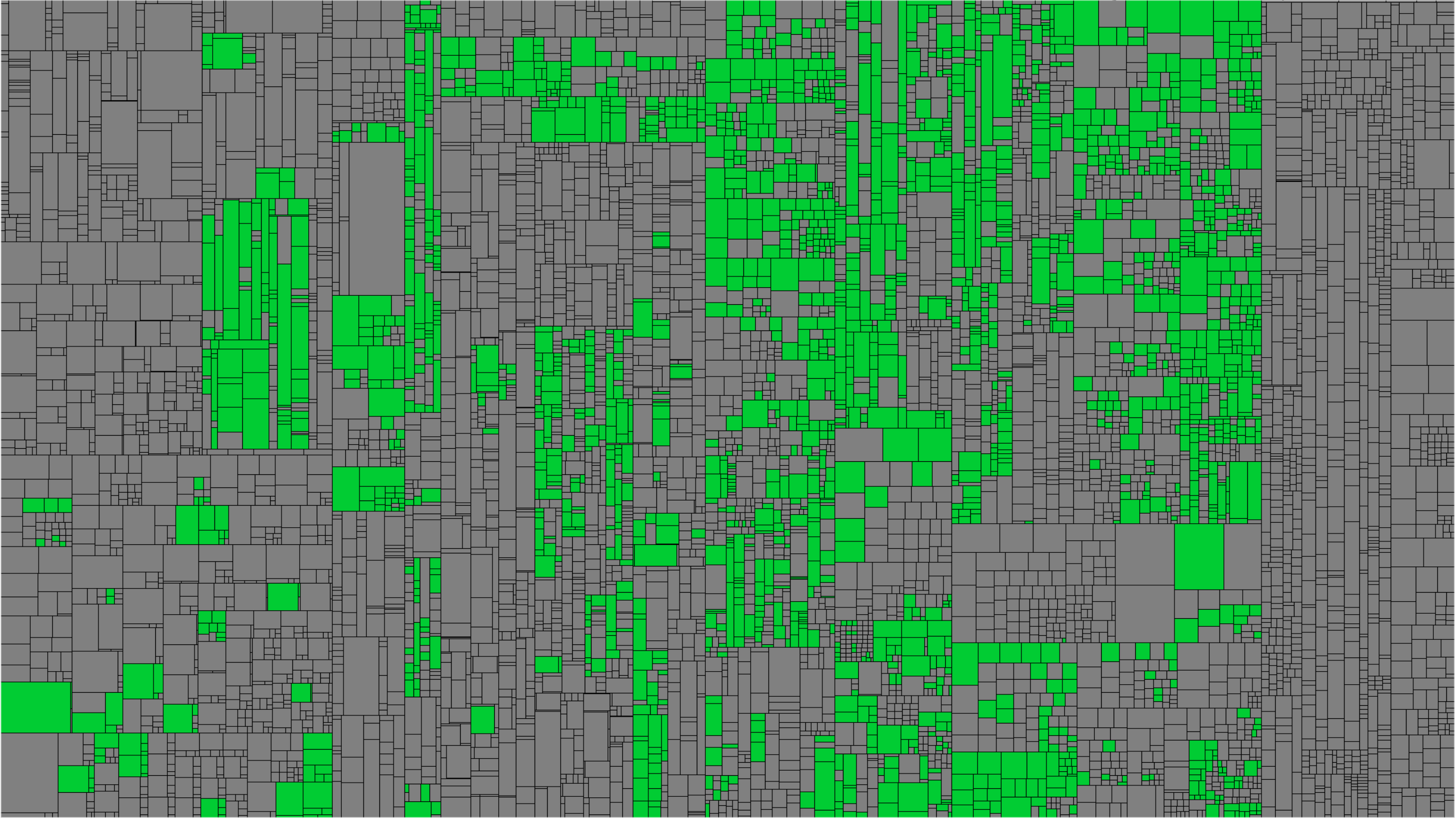
layer 1

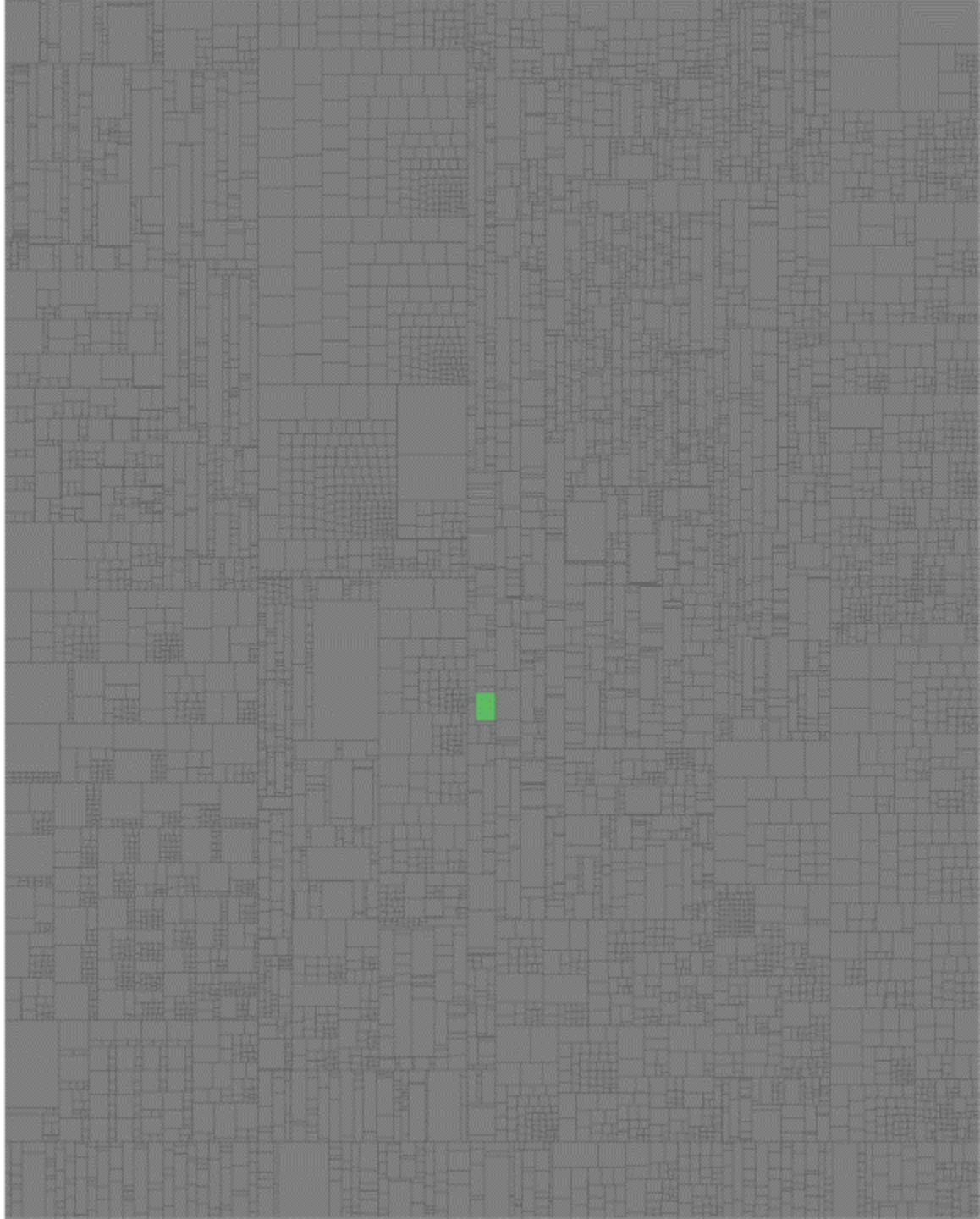


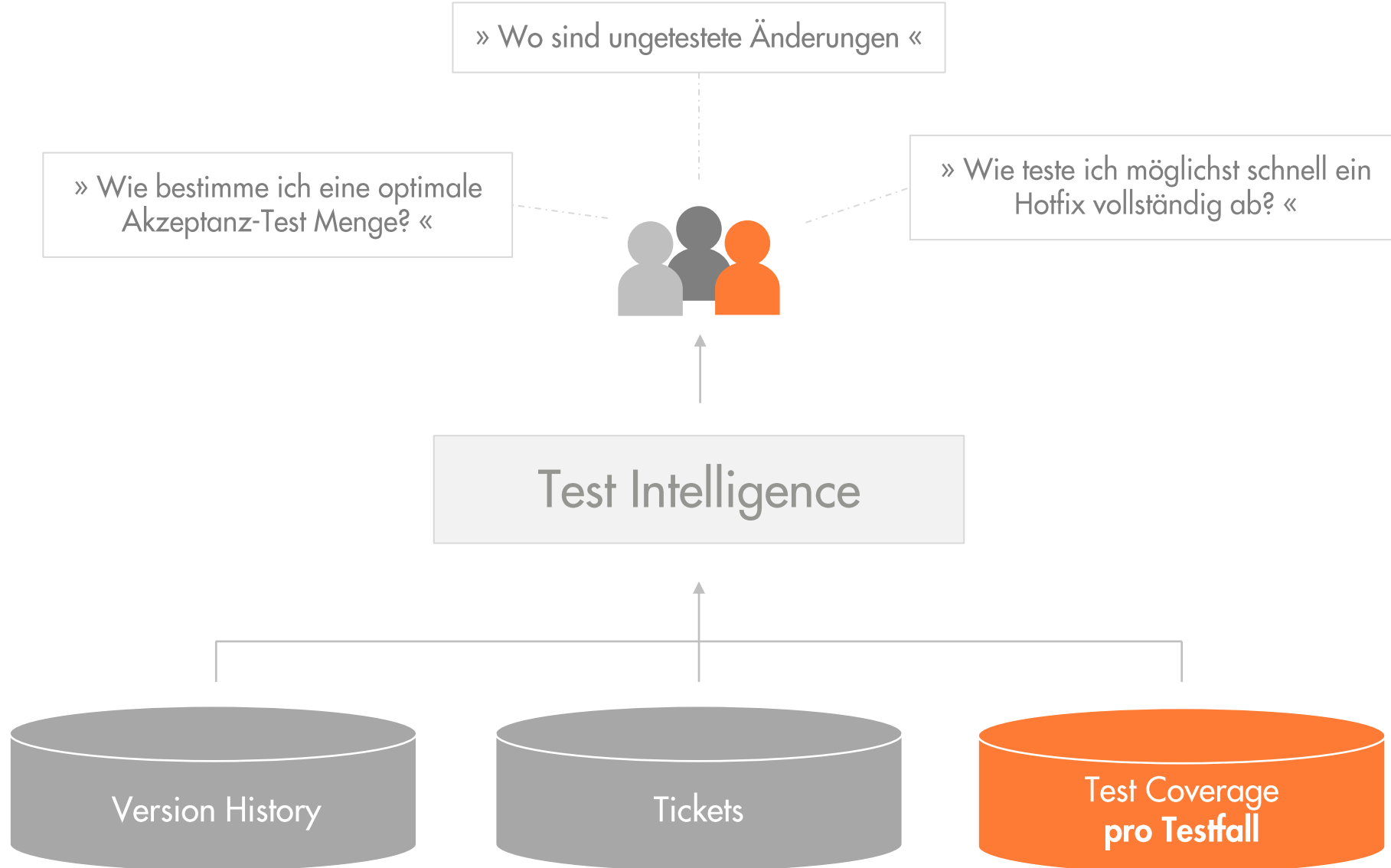


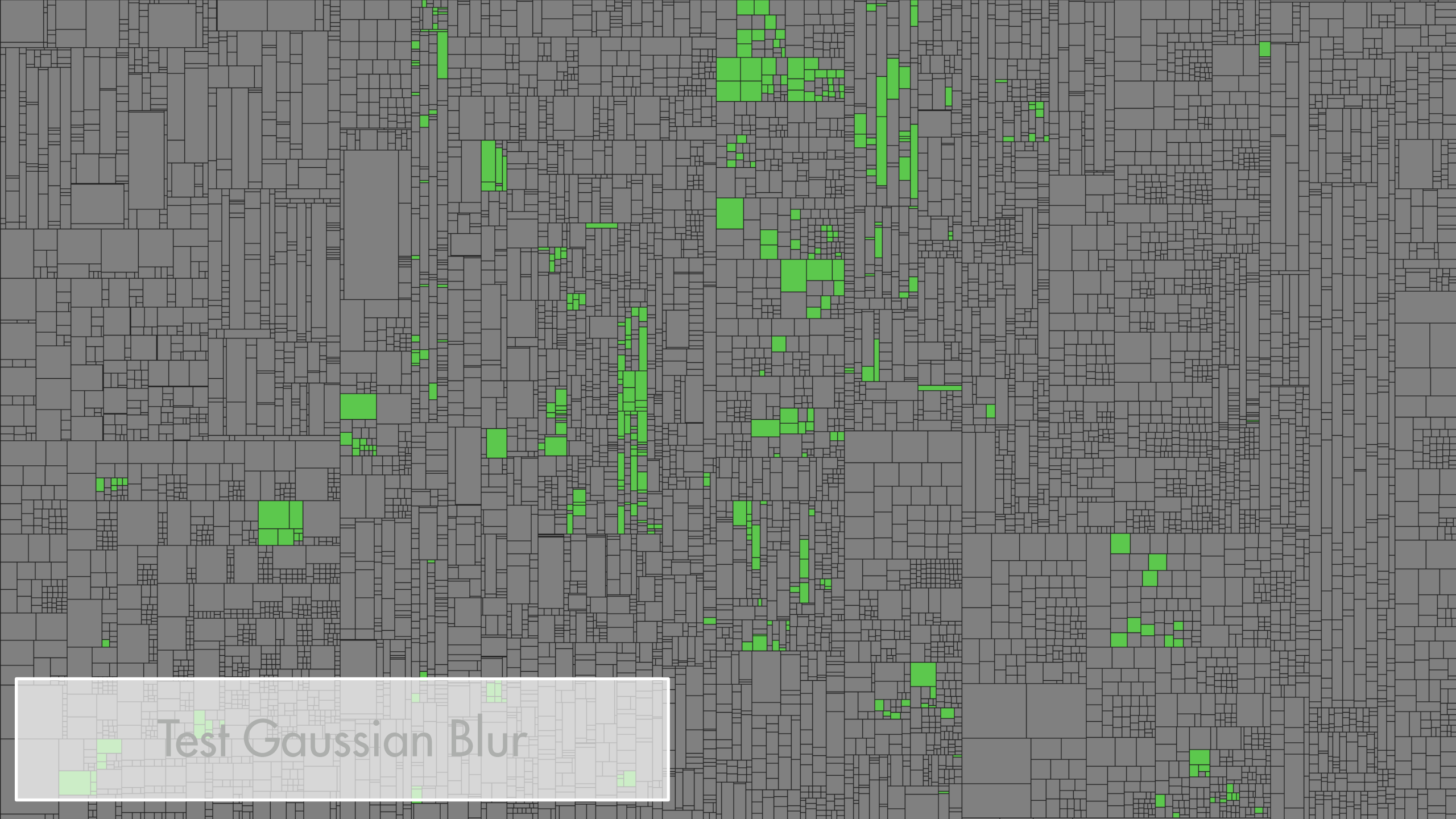










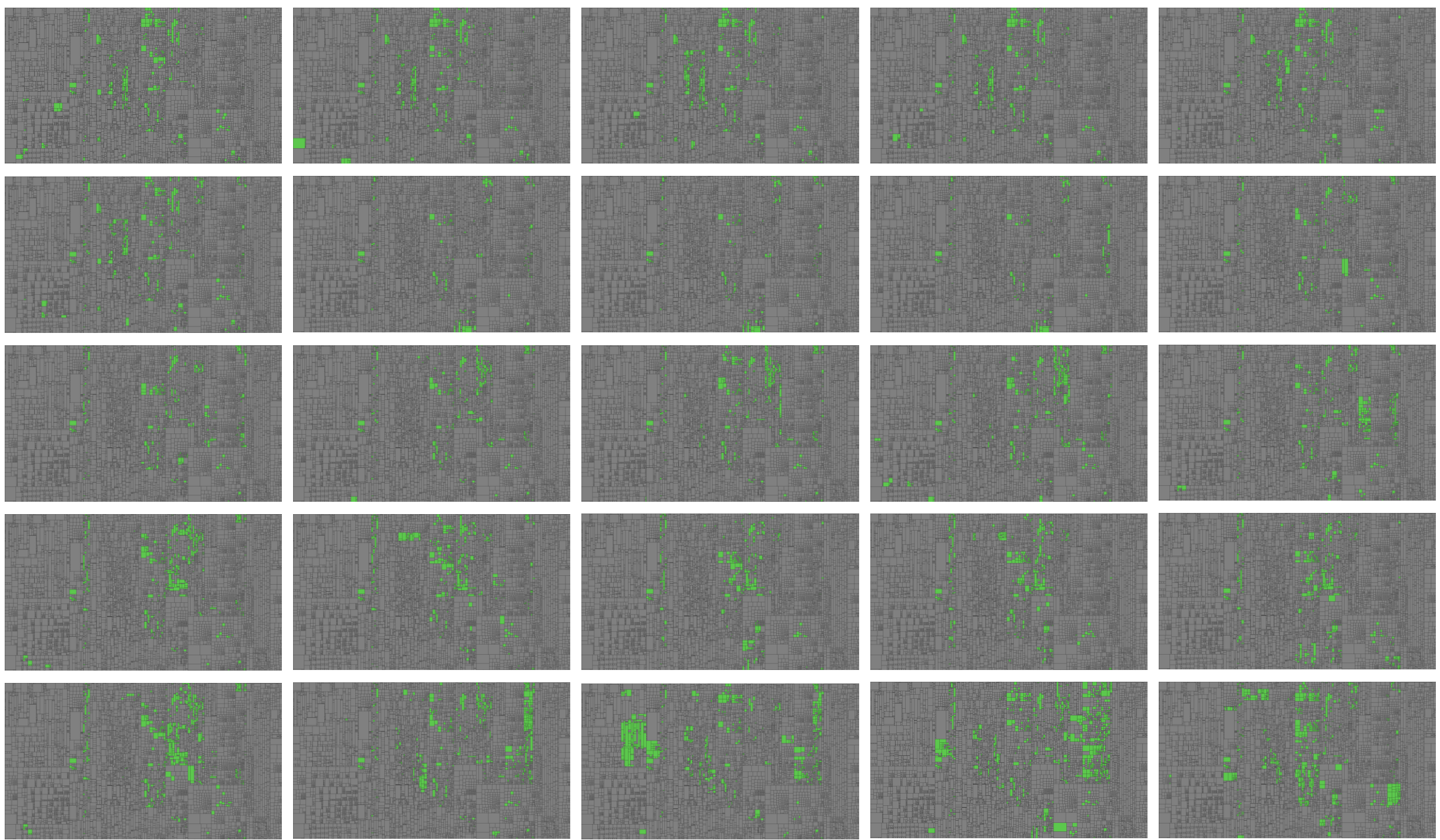


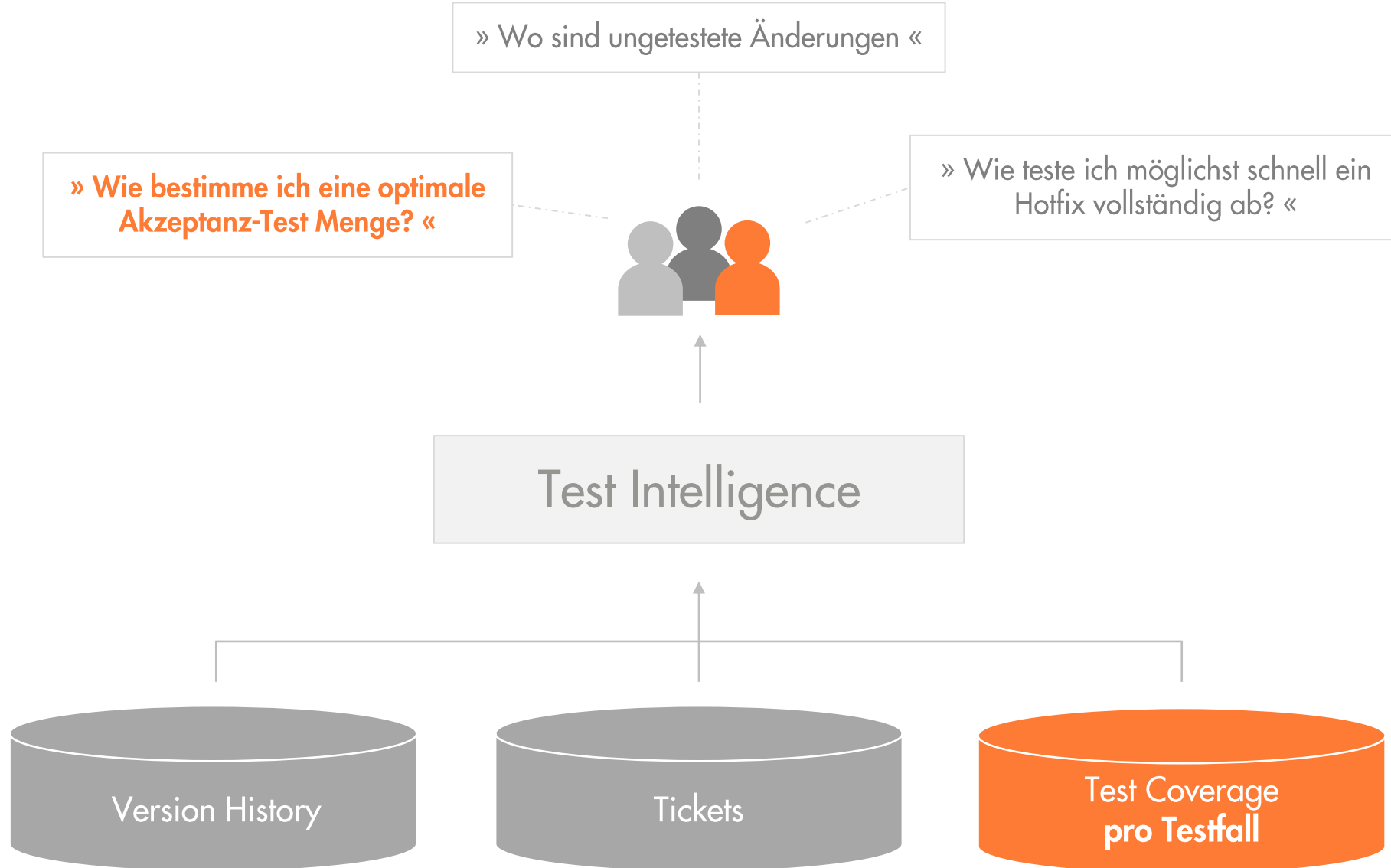
Test Gaussian Blur

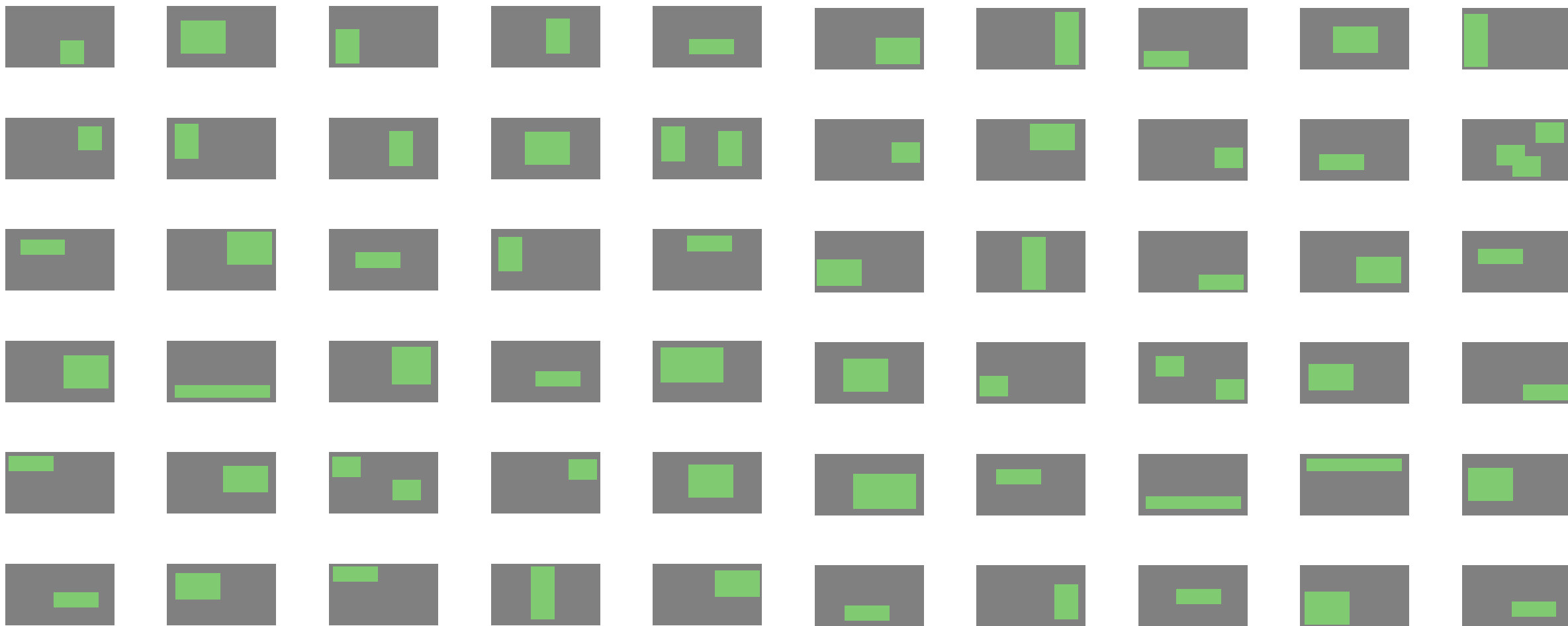


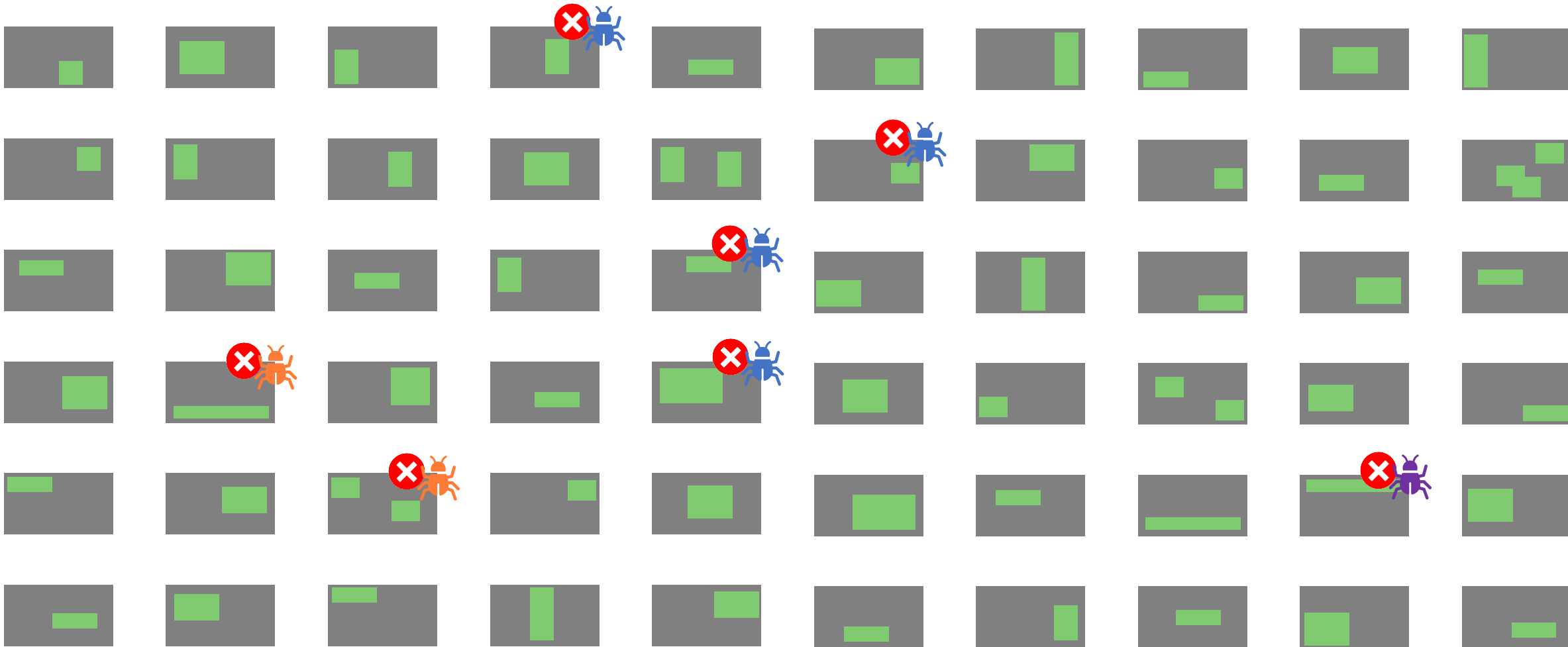


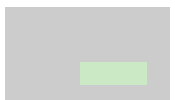
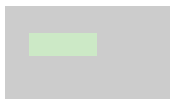
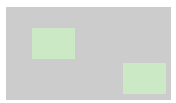
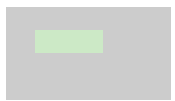
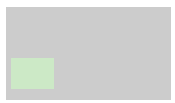
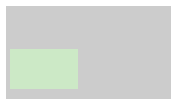
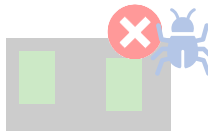
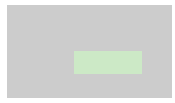
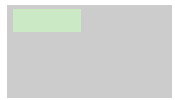
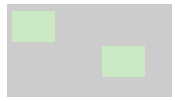
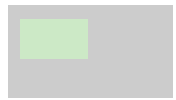
Test Create and Modify  
Selection

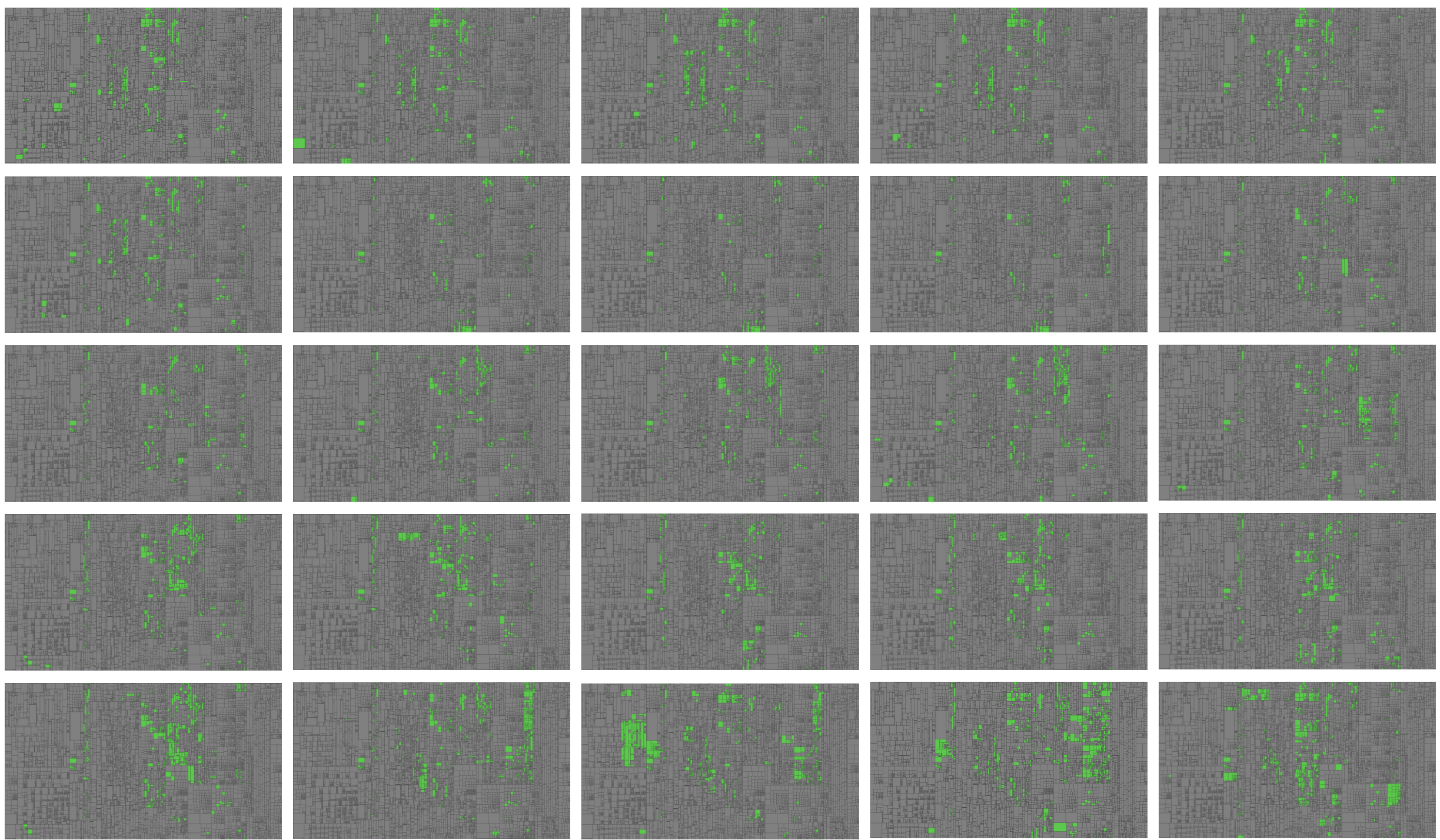


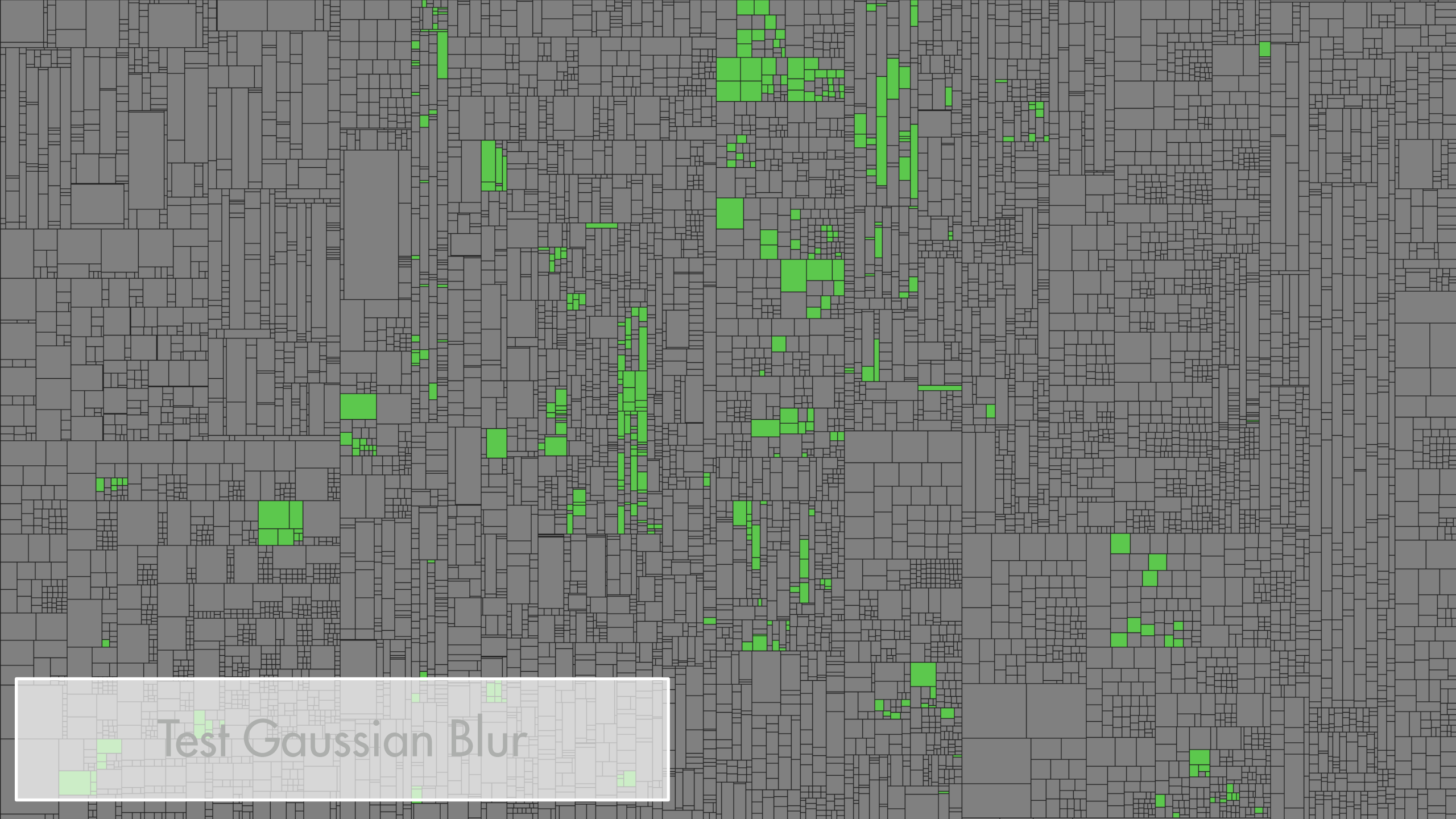










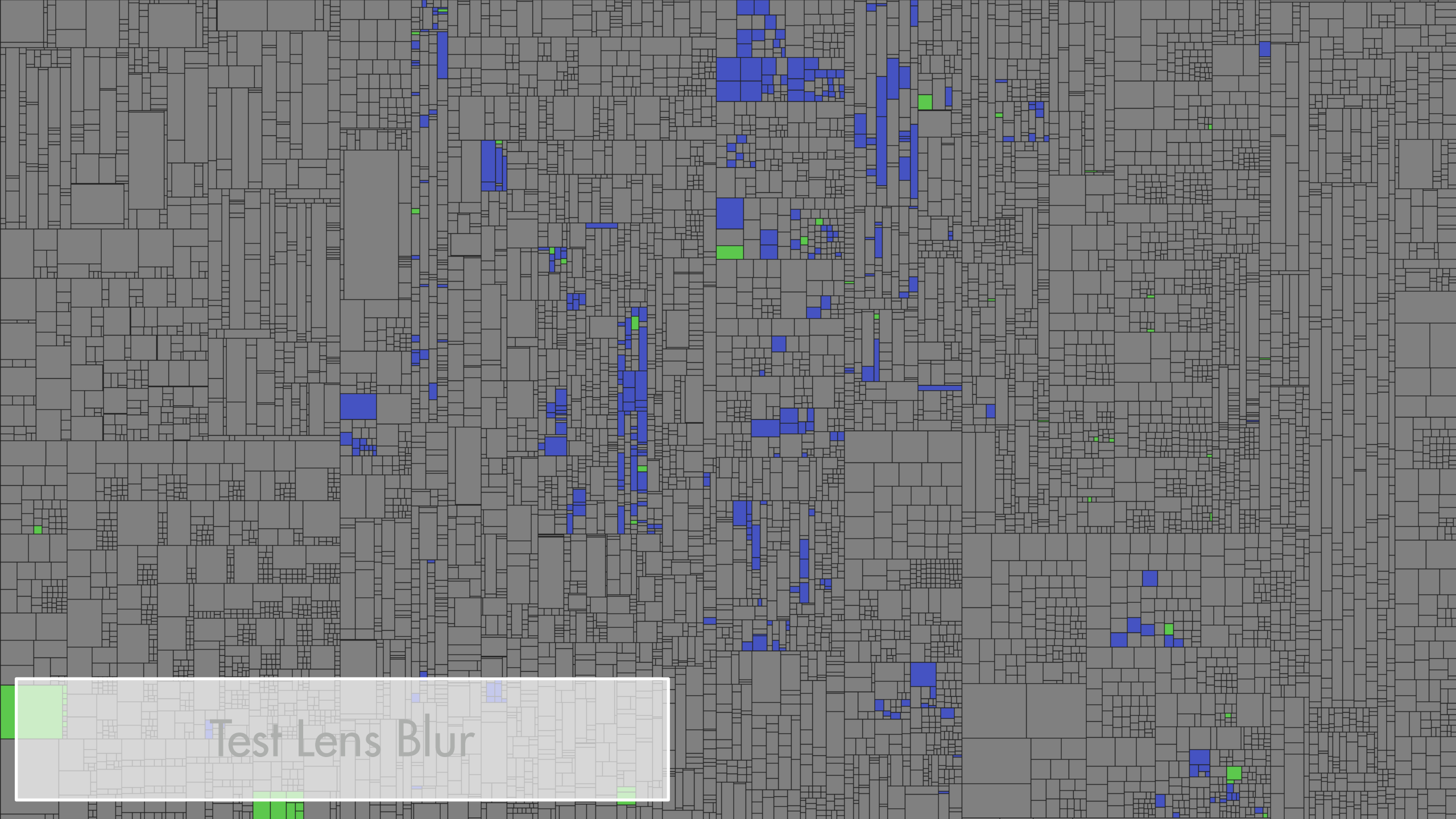


Test Gaussian Blur

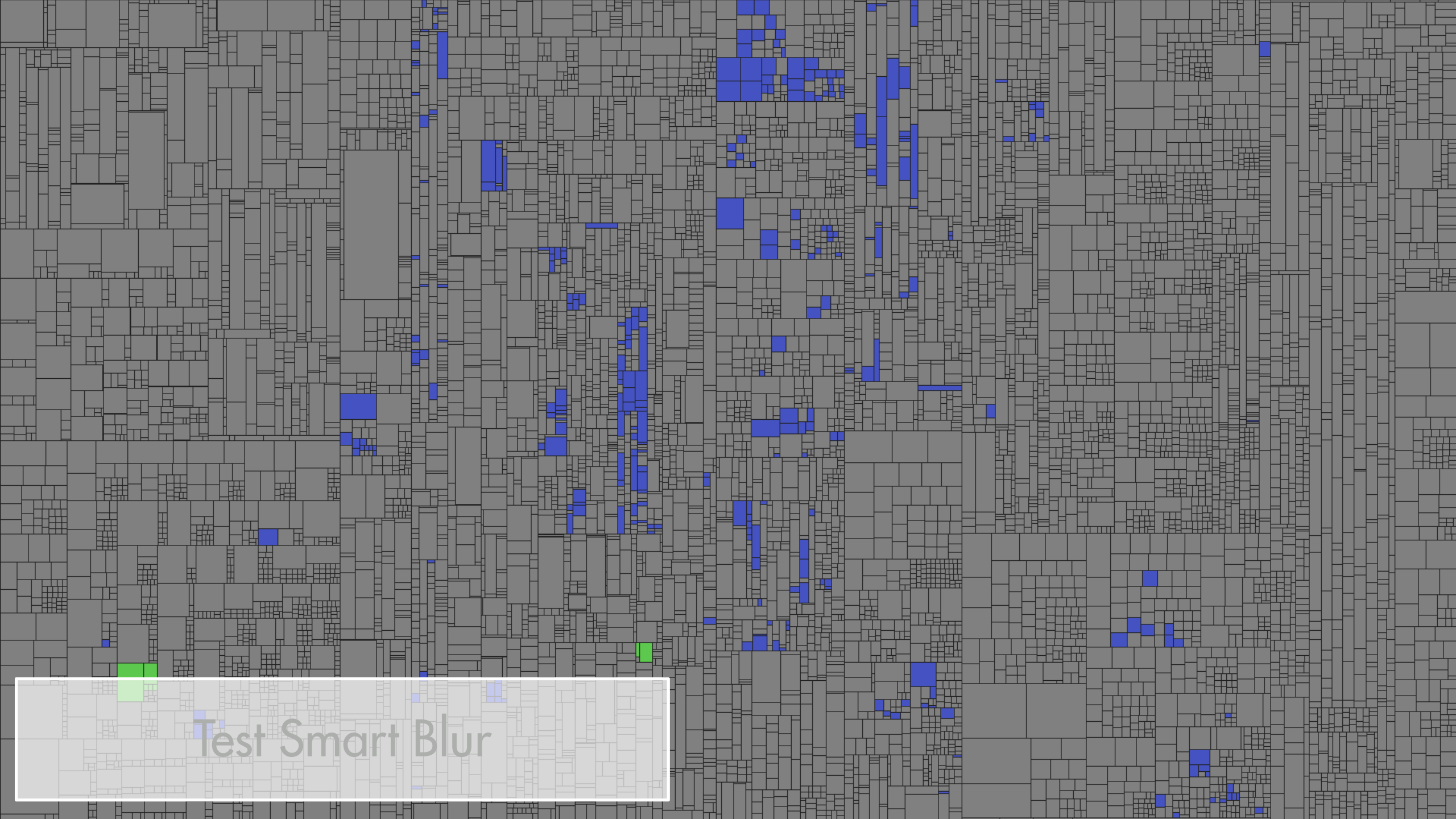




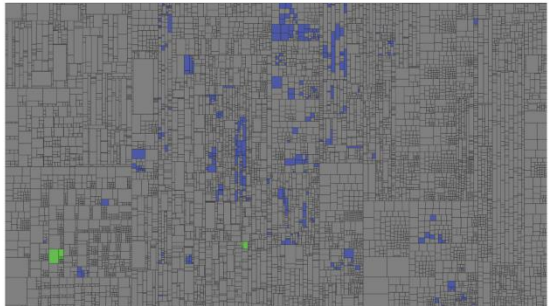
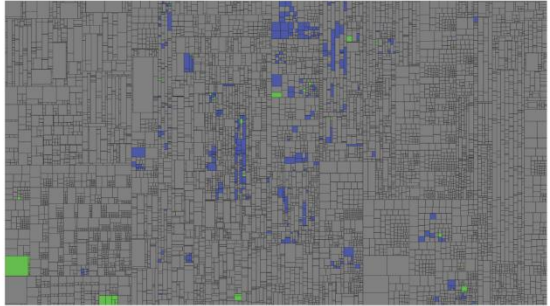
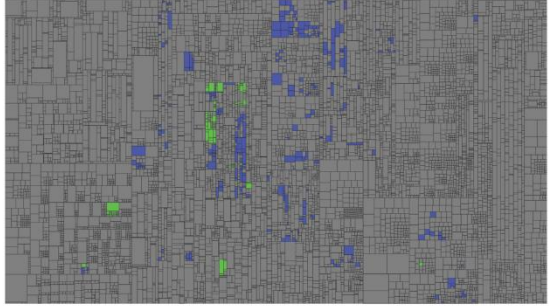
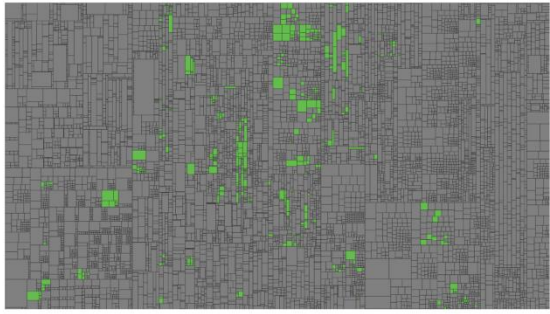
Test Motion Blur




Test Lens Blur



Test Smart Blur

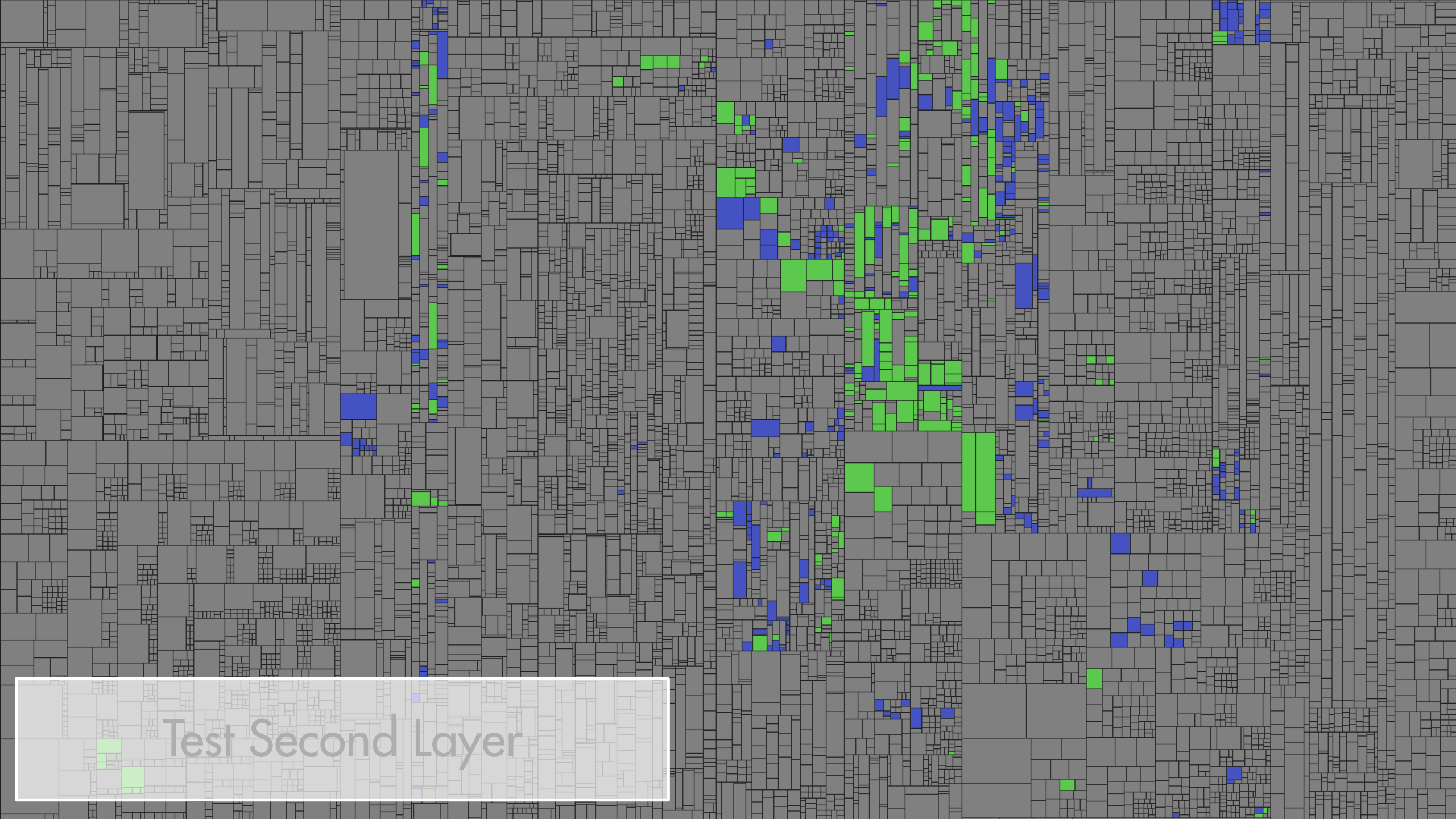




Test Create and Modify  
Selection

A treemap visualization showing a hierarchical structure of data. The background is a dense grid of grey rectangles. Overlaid on this are several clusters of smaller rectangles in green and blue. The green rectangles are scattered throughout, with notable concentrations in the upper-left, middle-right, and lower-right areas. The blue rectangles are also scattered, with some clusters in the upper-right and middle-left areas. The overall layout is complex and non-uniform.

Test Change View Settings

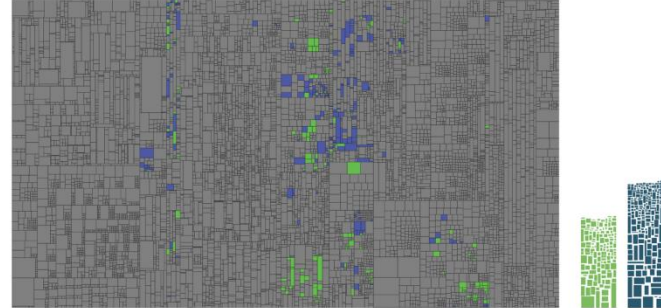
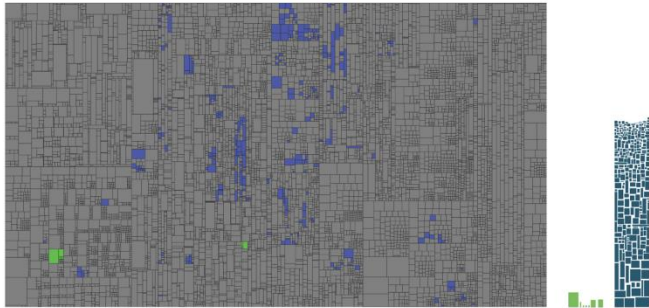
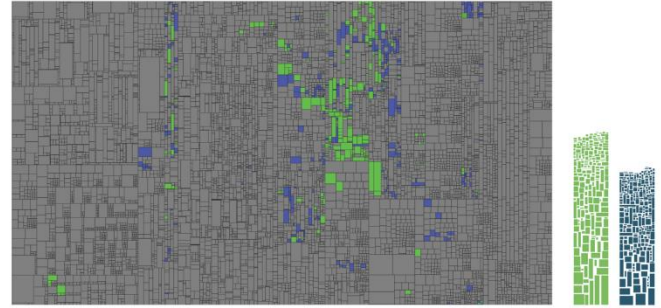
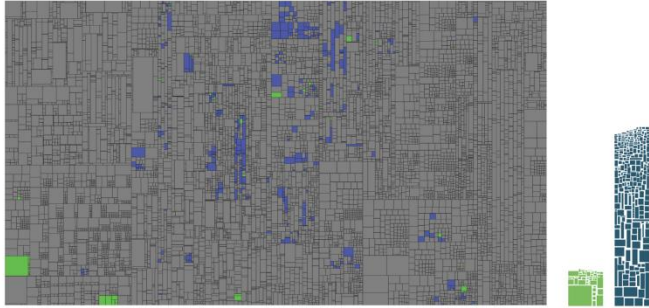
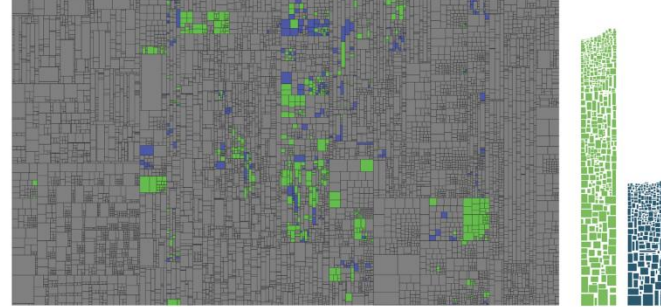
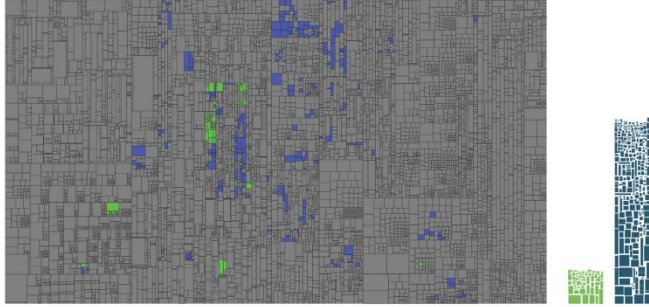
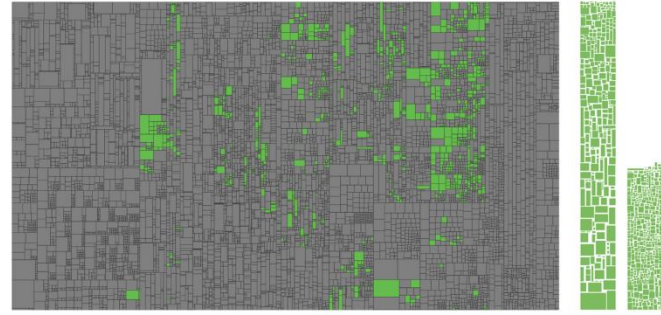
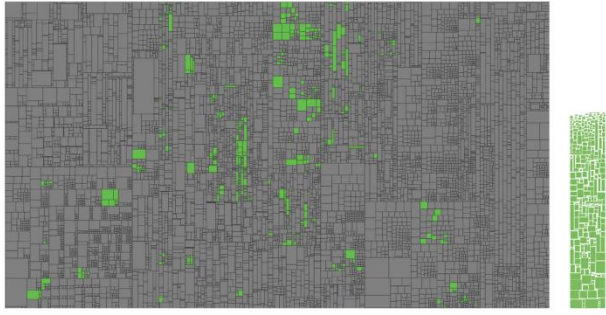


Test Second Layer

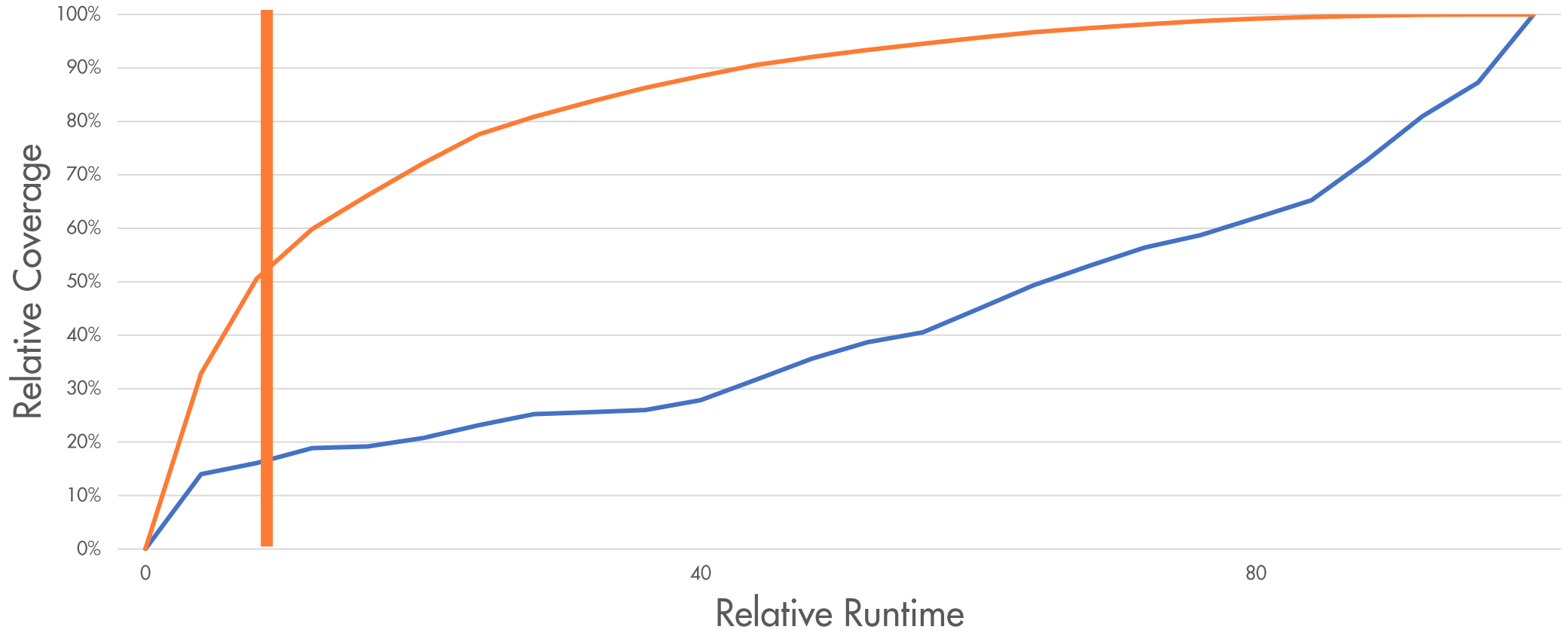
The image consists of a dense, overlapping grid of gray rectangles of various sizes and orientations. Scattered throughout this grid are several smaller rectangles in blue and green. These colored rectangles are concentrated in certain areas, such as the upper right and lower right quadrants, and are more sparsely distributed in the lower left. The overall effect is a complex, textured pattern of gray with occasional highlights of blue and green.

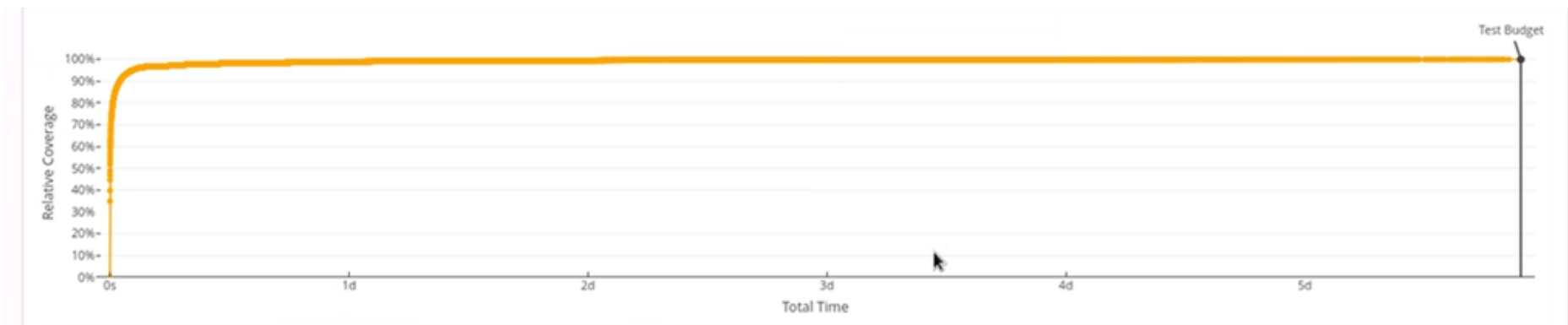
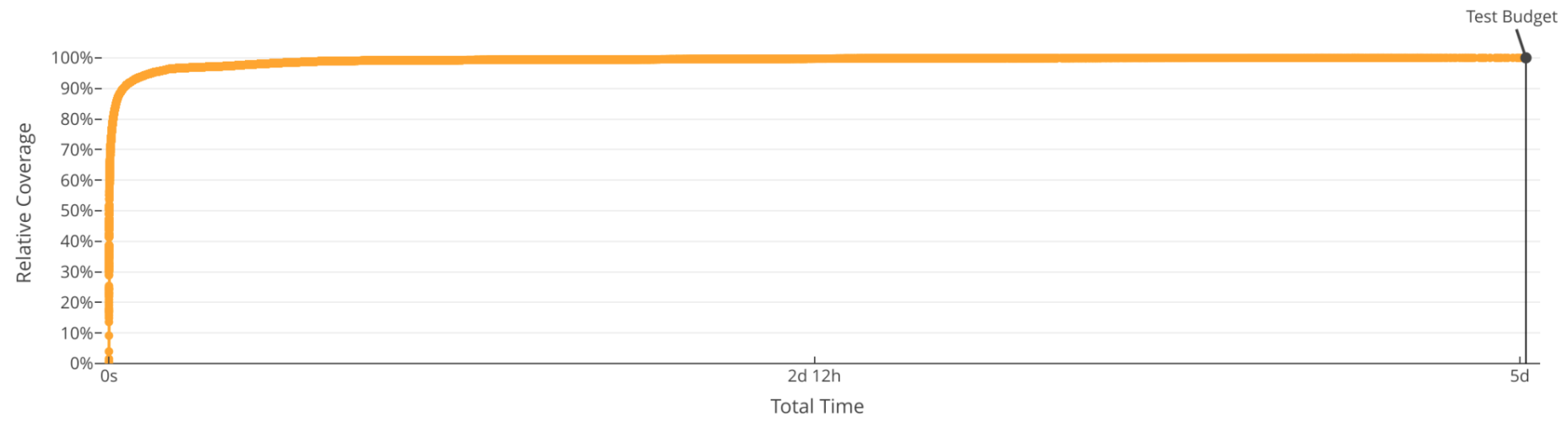
Test Save Image

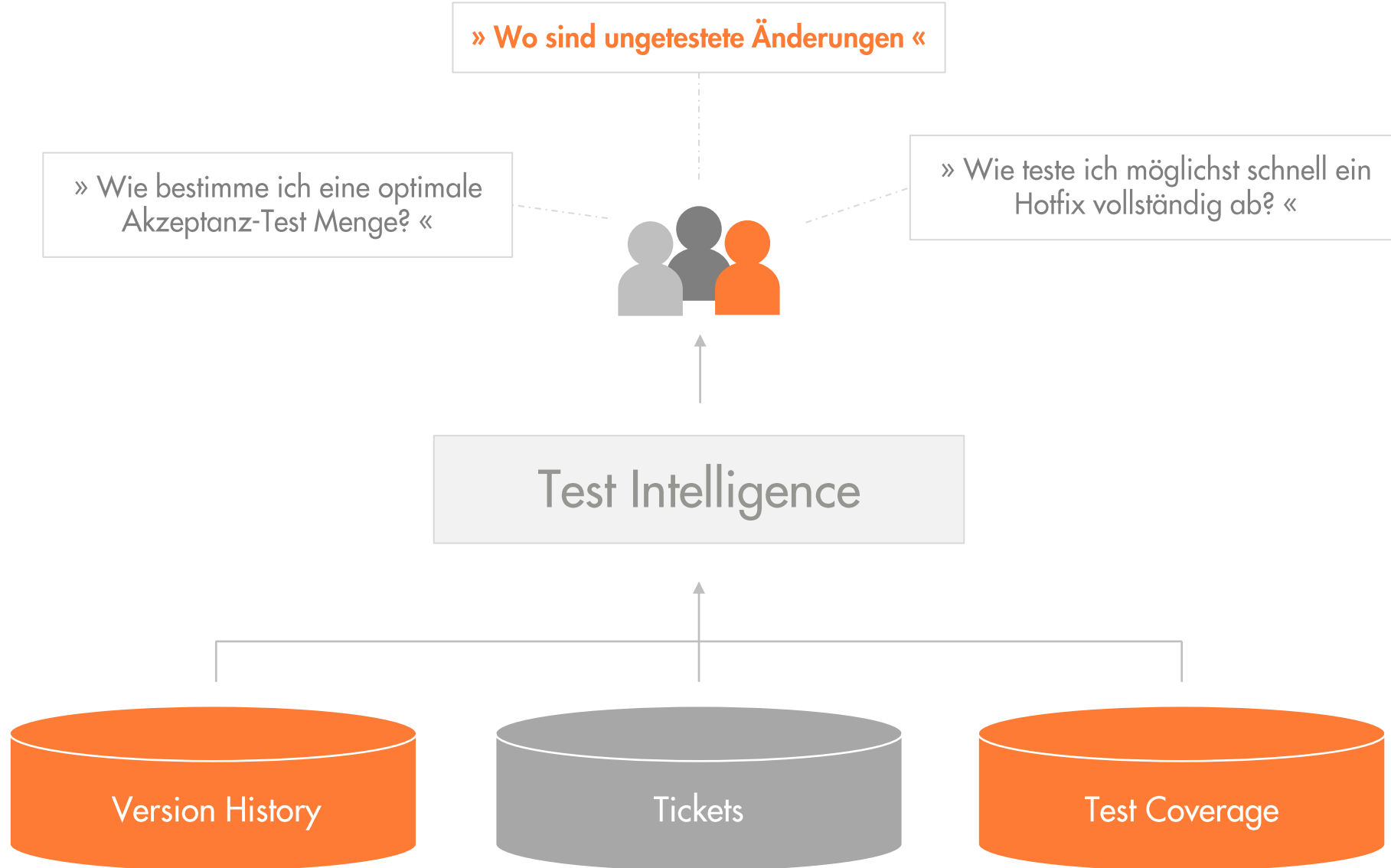


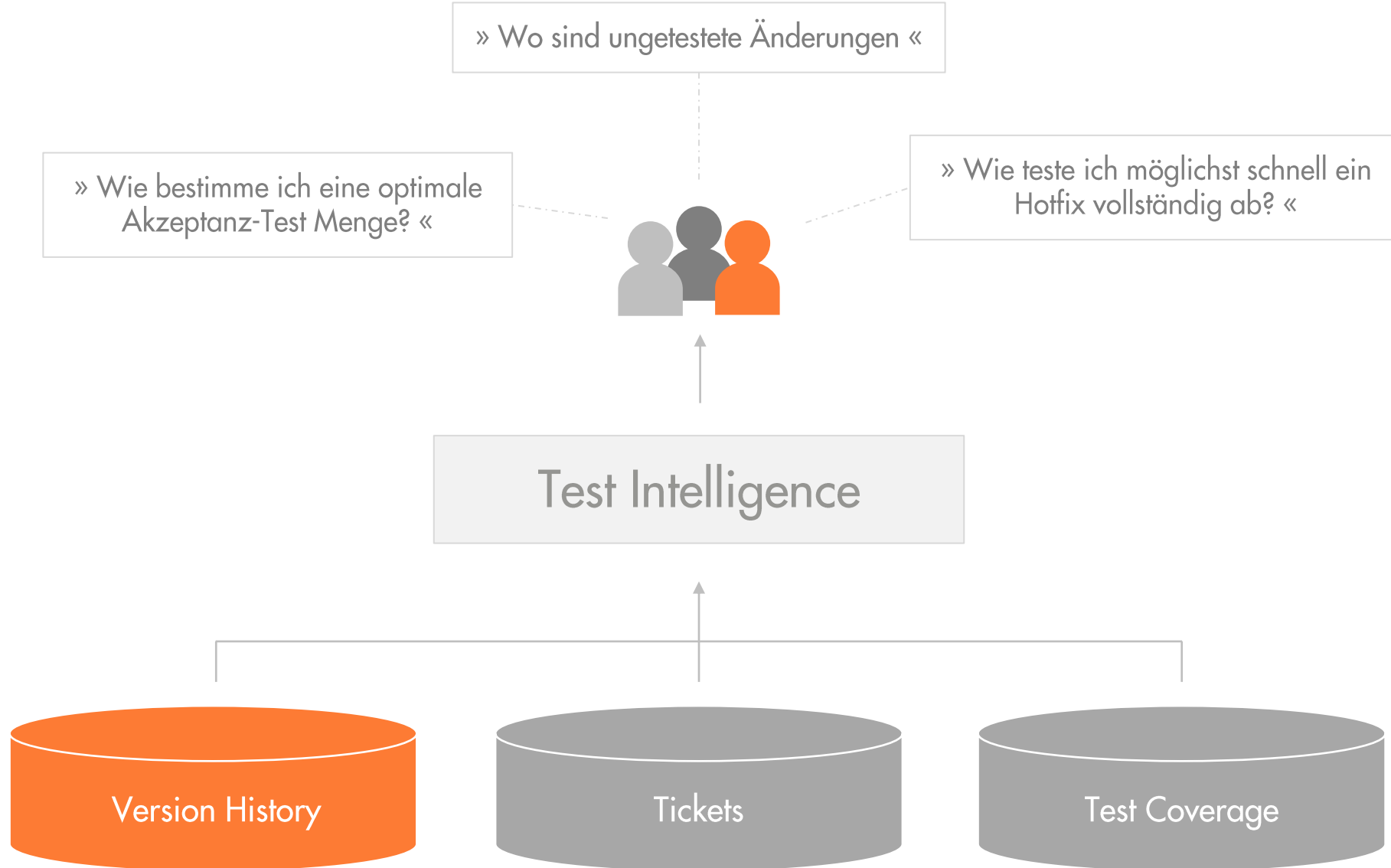


# Time vs Code Coverage









```

760     int canvasHeight = canvas.getHeight();
761
762     if ((imageWidth > canvasWidth) || (imageHeight > canvasHeight)) {
763         BufferedImage newImage = ImageUtils.crop(image, -getTX(), -getTY(), canvasWidth, canvasHeight);
764
765         BufferedImage tmp = image;
766         setImage(newImage);
767         tmp.flush();
768
769         setTranslation(0, 0);
770         return true;
771     }
772     return false;
773 }
774
775 @Override
776 public void enlargeCanvas(int north, int east, int south, int west) {
777     // all coordinates in this method are
778     // relative to the previous state of the canvas
779     Rectangle imageBounds = getImageBounds();
780     Rectangle canvasBounds = comp.getCanvasBounds();
781
782     Rectangle transformedCanvasBounds = new Rectangle(
783         canvasBounds.x - west,
784         canvasBounds.y - north,
785         canvasBounds.width + west + east,
786         canvasBounds.height + north + south);
787
788     if (imageBounds.contains(transformedCanvasBounds)) {
789         // even after the canvas enlargement, the image does not need to be enlarged
790         translationX += west;
791         translationY += north;
792     } else {
793         enlargeImage(transformedCanvasBounds);
794     }
795 }
796
797 @Override
798 ContentLayerMoveEdit createMovementEdit(int oldTX, int oldTY) {
799     ContentLayerMoveEdit edit;
800     boolean needsEnlarging = checkImageDoesNotCoverCanvas();
801     if (needsEnlarging) {
802         BufferedImage backupImage = getImage();
803         enlargeImage(comp.getCanvasBounds());
804         edit = new ContentLayerMoveEdit(this, backupImage, oldTX, oldTY);
805     } else {
806         edit = new ContentLayerMoveEdit(this, null, oldTX, oldTY);
807     }
808
809     return edit;
810 }
811

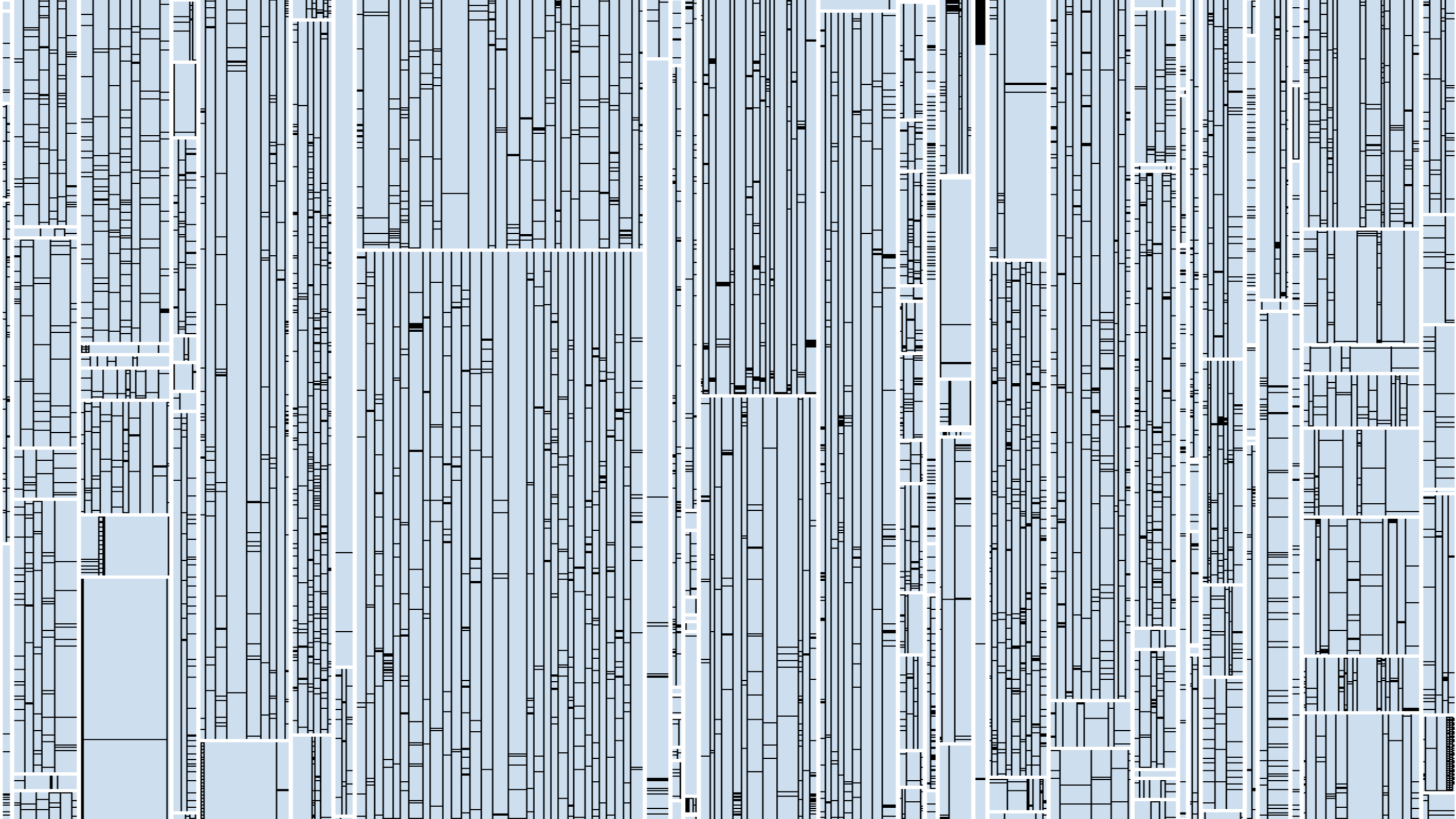
```

```

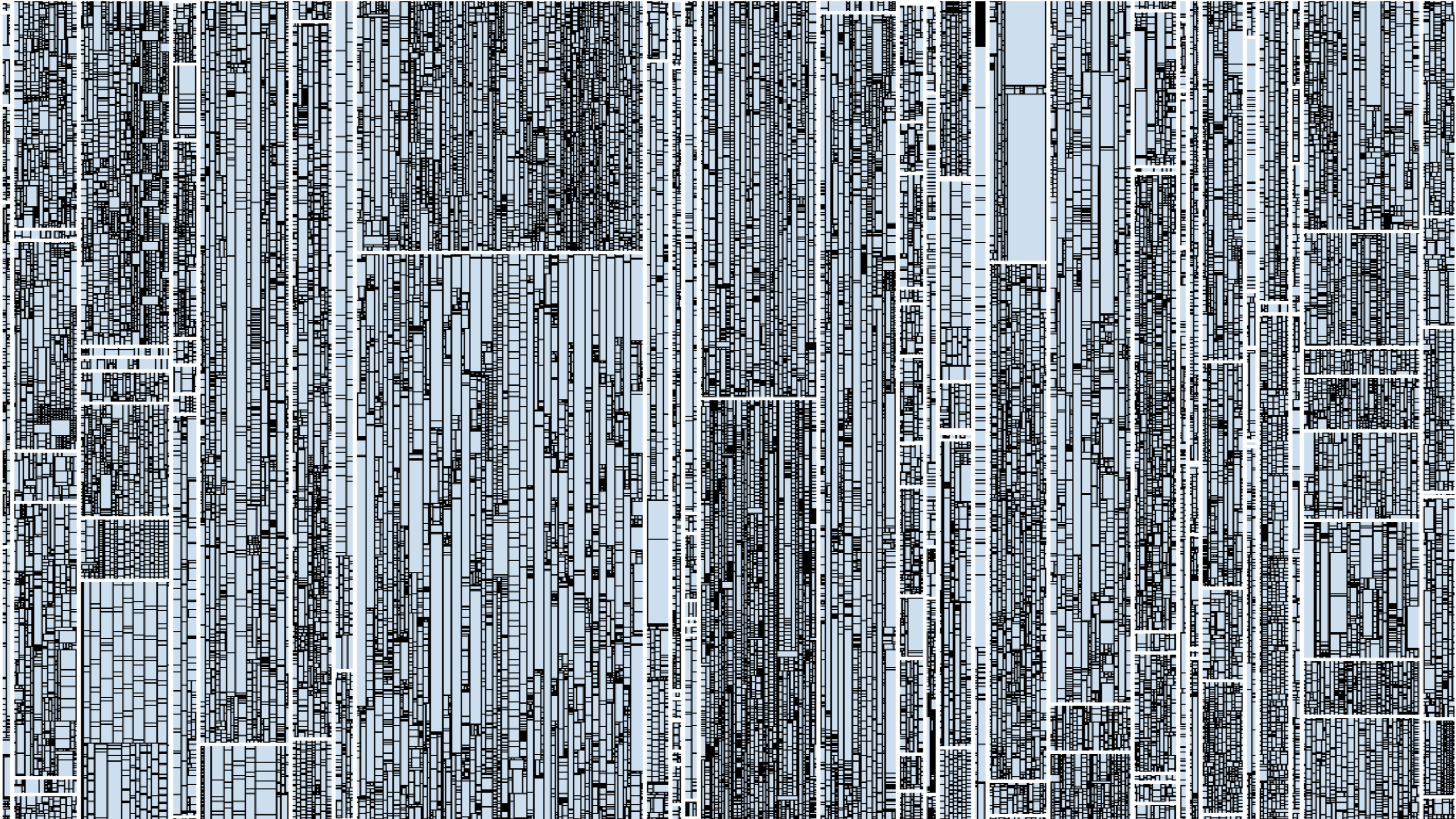
777     int canvasHeight = canvas.getHeight();
778
779     if ((imageWidth > canvasWidth) || (imageHeight > canvasHeight)) {
780         BufferedImage newImage = ImageUtils.crop(image, -getTX(), -getTY(), canvasWidth, canvasHeight);
781
782         BufferedImage tmp = image;
783         setImage(newImage);
784         tmp.flush();
785
786         setTranslation(0, 0);
787         return true;
788     }
789     return false;
790 }
791
792 @Override
793 public void enlargeCanvas(int north, int east, int south, int west) {
794     // all coordinates in this method are
795     // relative to the previous state of the canvas
796     Rectangle imageBounds = getImageBounds();
797     Rectangle canvasBounds = comp.getCanvasBounds();
798
799     int newX = canvasBounds.x - west;
800     int newY = canvasBounds.y - north;
801     int newWidth = canvasBounds.width + west + east;
802     int newHeight = canvasBounds.height + north + south;
803     Rectangle newCanvasBounds = new Rectangle(newX, newY, newWidth, newHeight);
804
805     if (imageBounds.contains(newCanvasBounds)) {
806         // even after the canvas enlargement, the image does not need to be enlarged
807         translationX += west;
808         translationY += north;
809     } else {
810         enlargeImage(newCanvasBounds);
811     }
812 }
813
814 @Override
815 ContentLayerMoveEdit createMovementEdit(int oldTX, int oldTY) {
816     ContentLayerMoveEdit edit;
817     boolean needsEnlarging = checkImageDoesNotCoverCanvas();
818     if (needsEnlarging) {
819         BufferedImage backupImage = getImage();
820         enlargeImage(comp.getCanvasBounds());
821         edit = new ContentLayerMoveEdit(this, backupImage, oldTX, oldTY);
822     } else {
823         edit = new ContentLayerMoveEdit(this, null, oldTX, oldTY);
824     }
825
826     return edit;
827 }
828

```

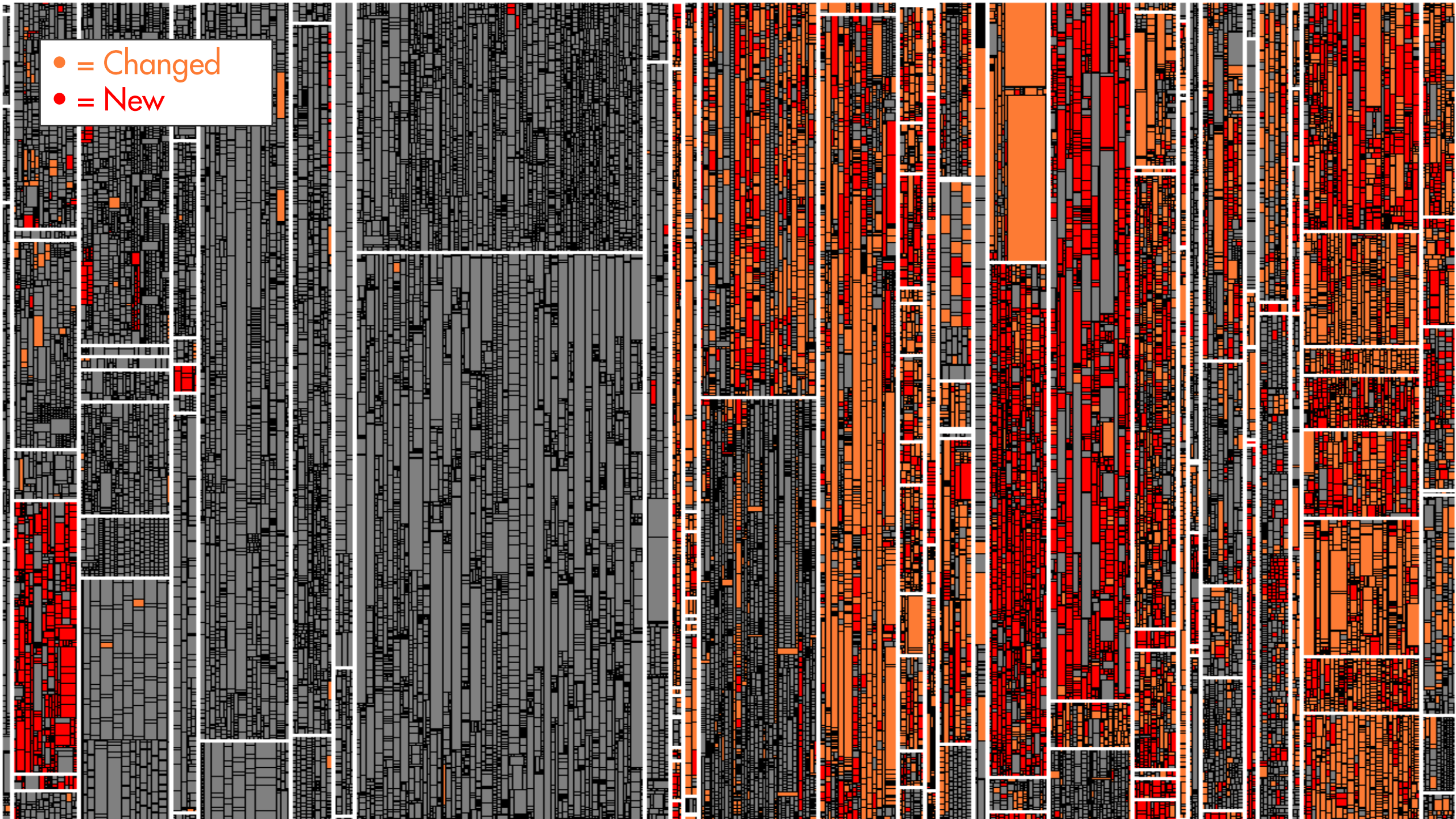


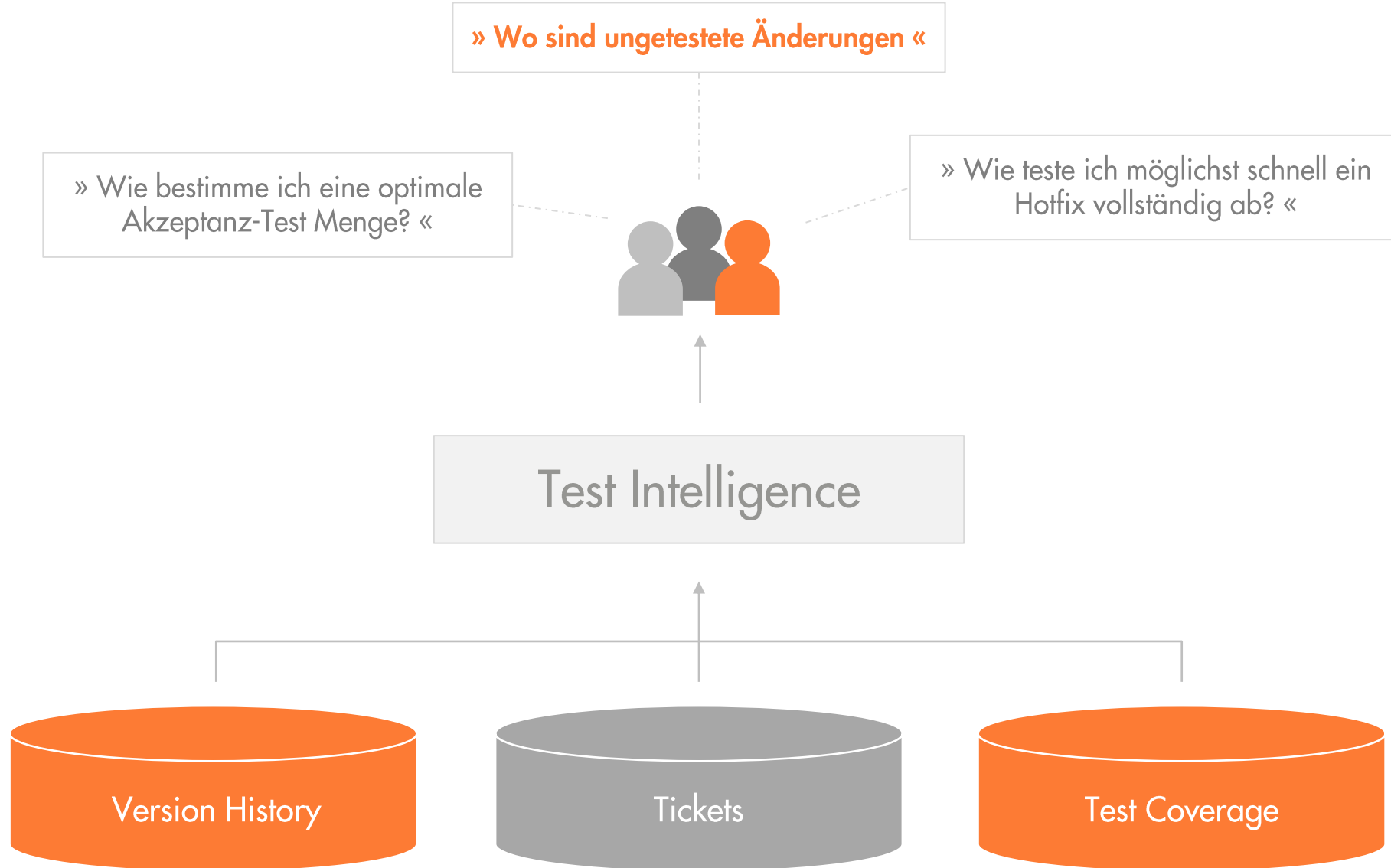




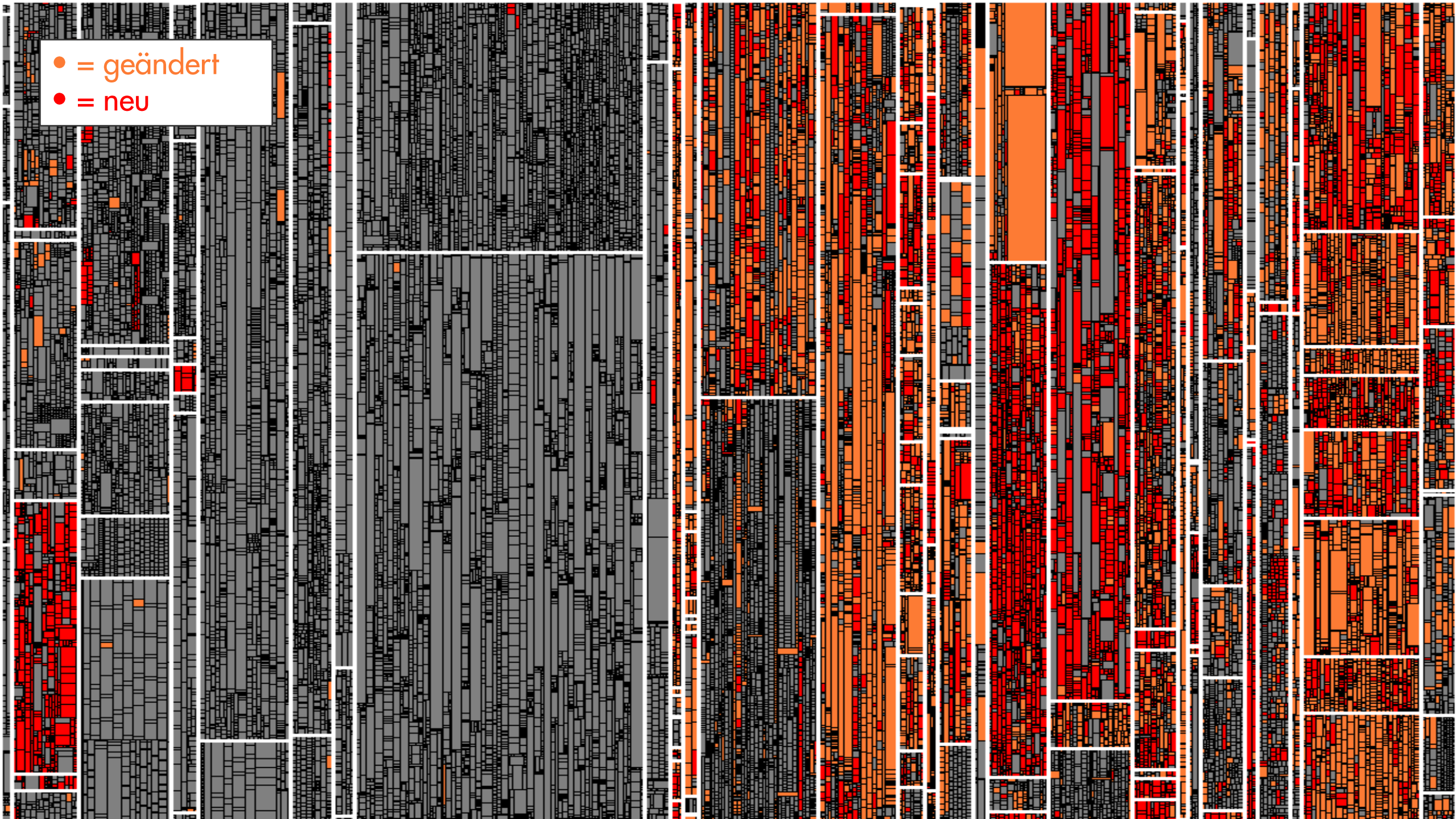


- = Changed
- = New



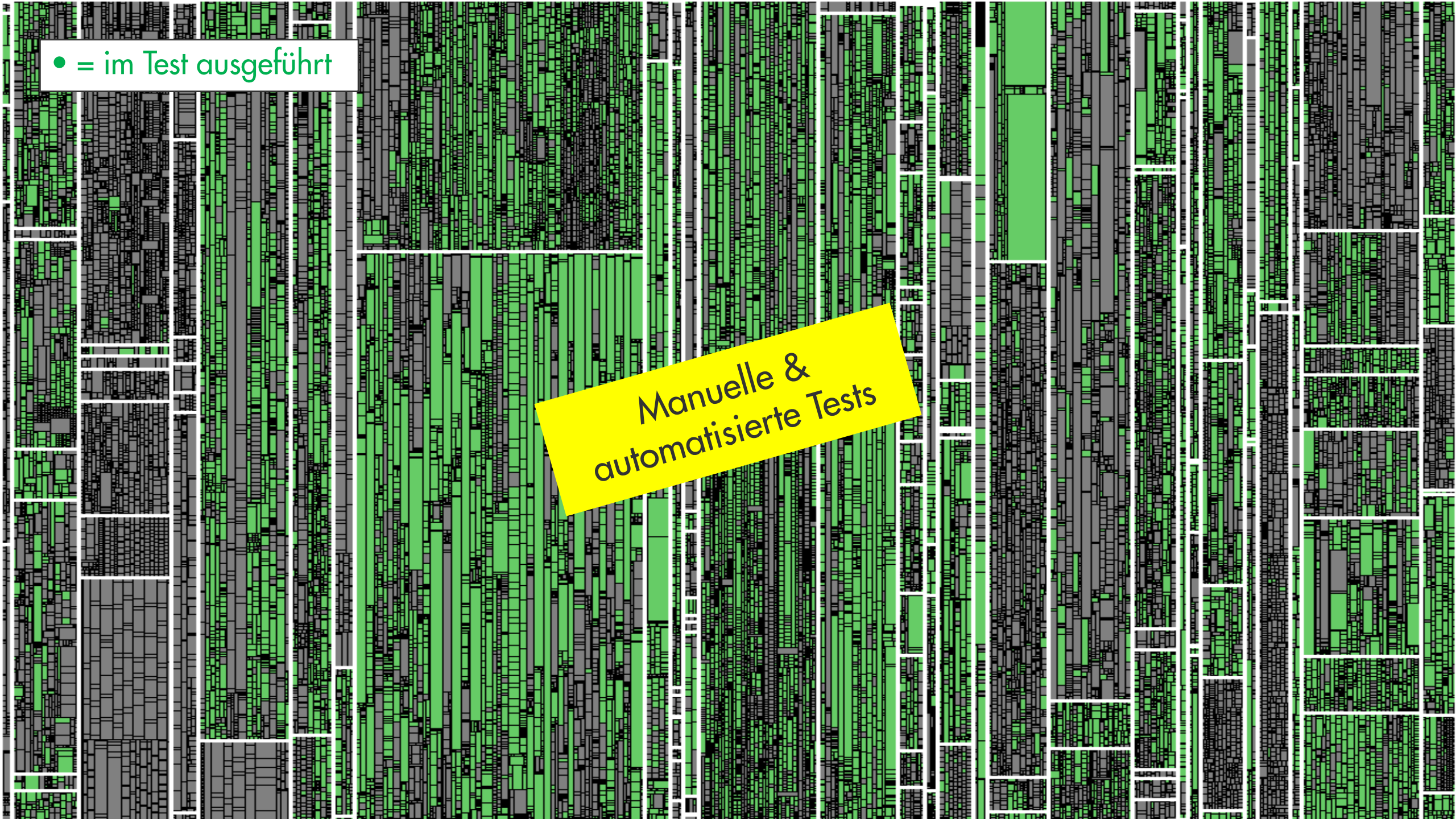


- = geändert
- = neu

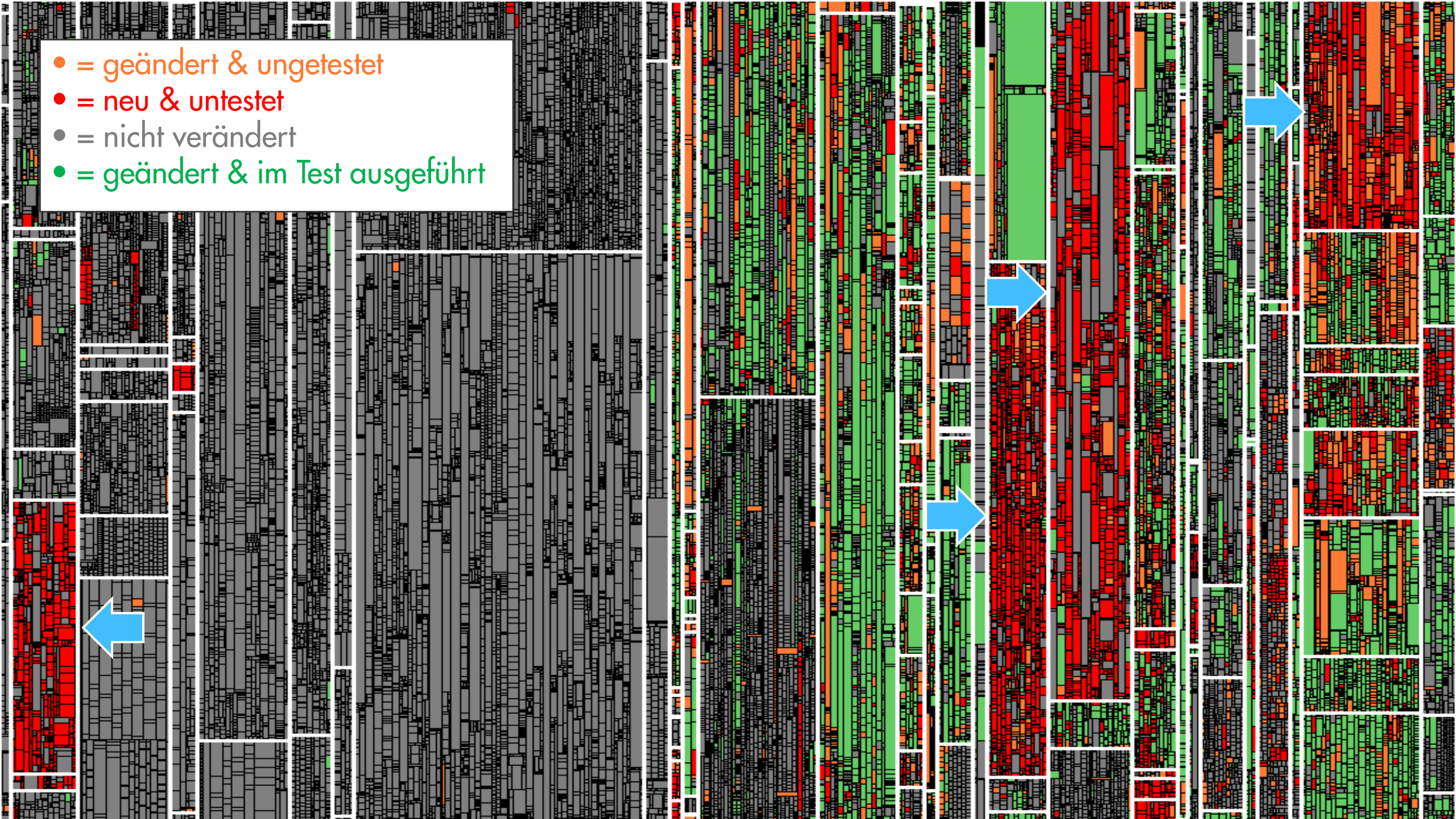


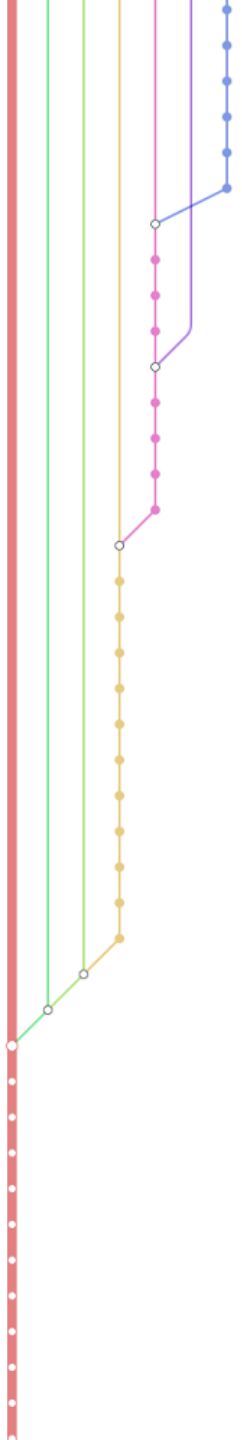
● = im Test ausgeführt

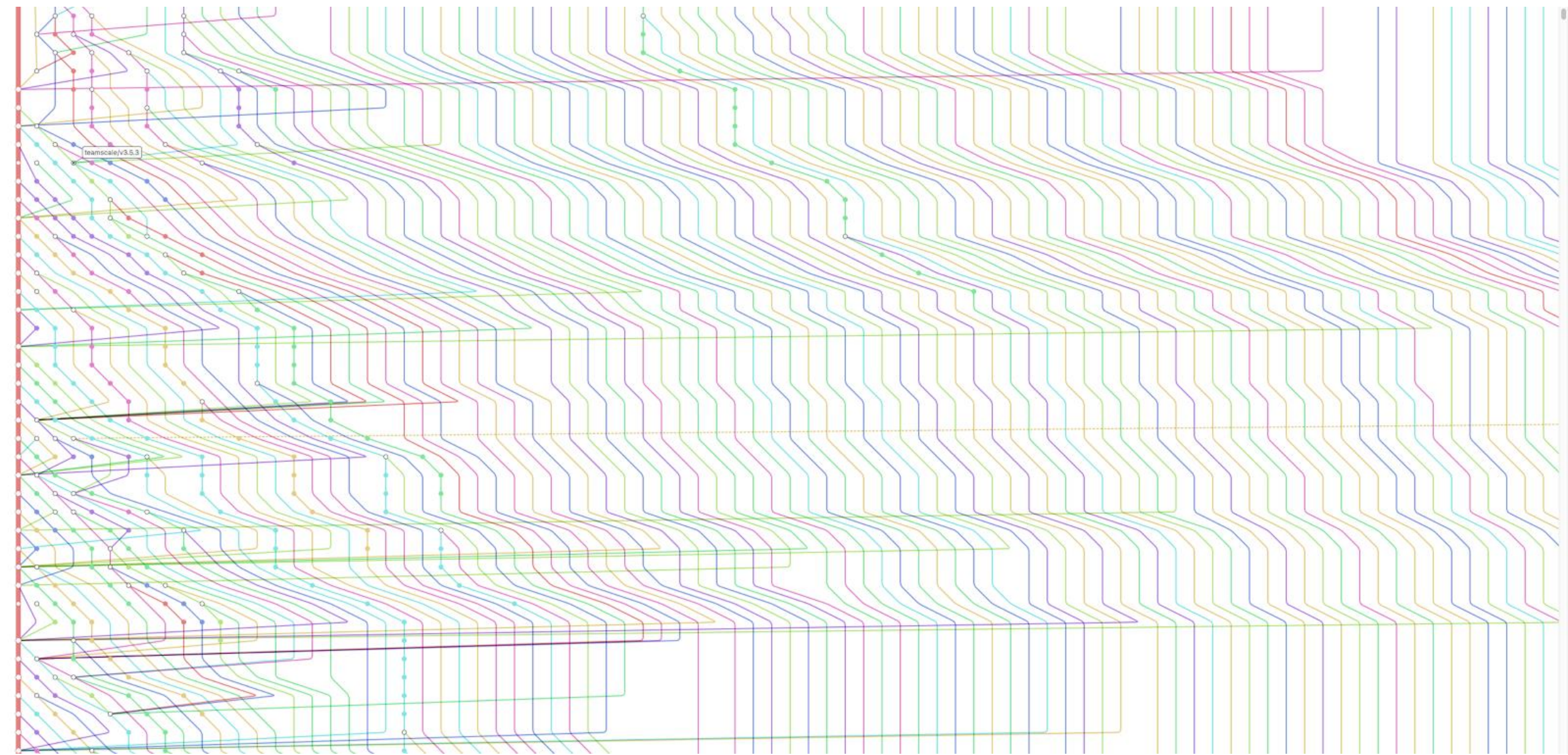
Manuelle &  
automatisierte Tests



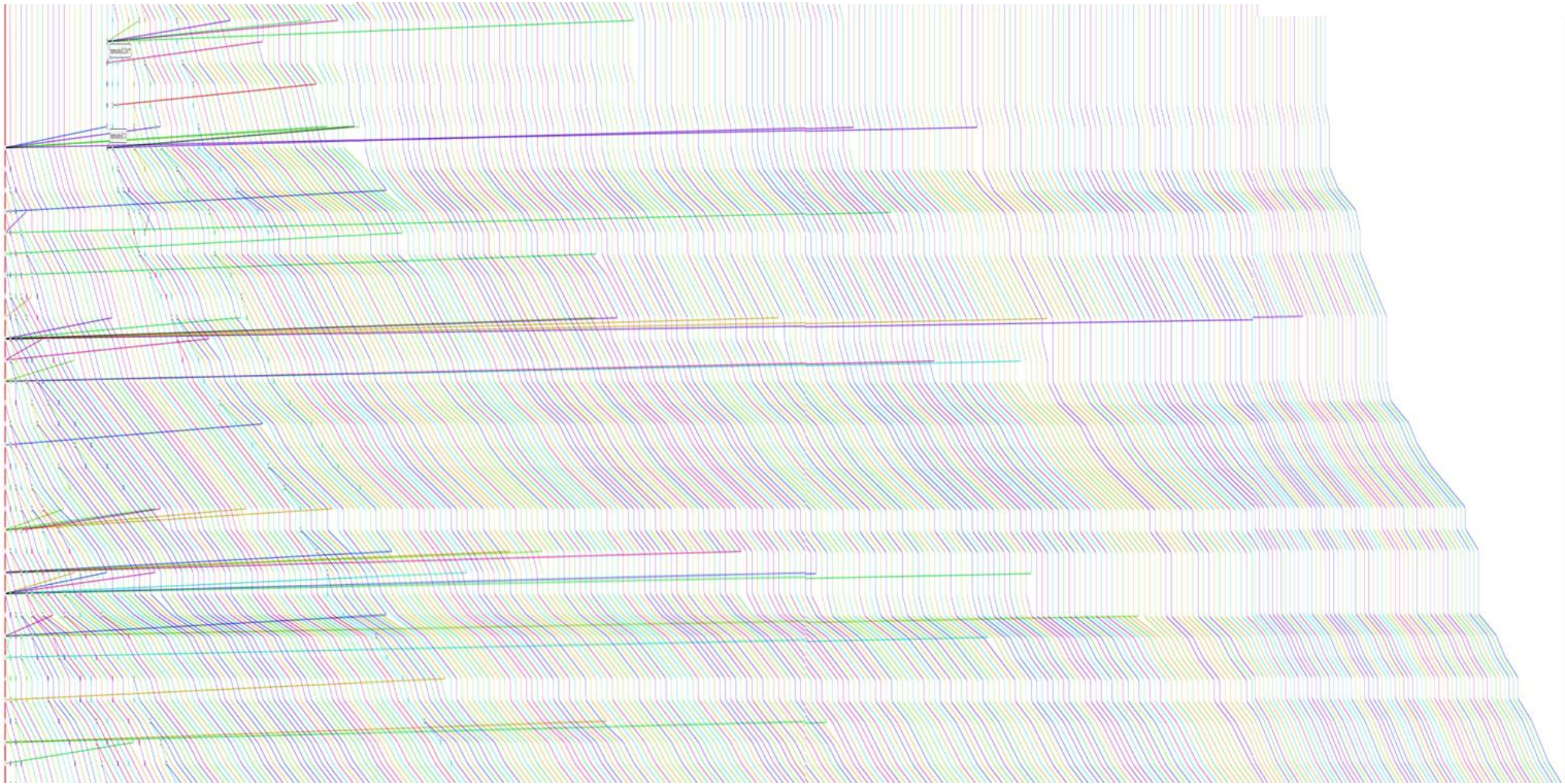
- = geändert & ungetestet
- = neu & untestet
- = nicht verändert
- = geändert & im Test ausgeführt

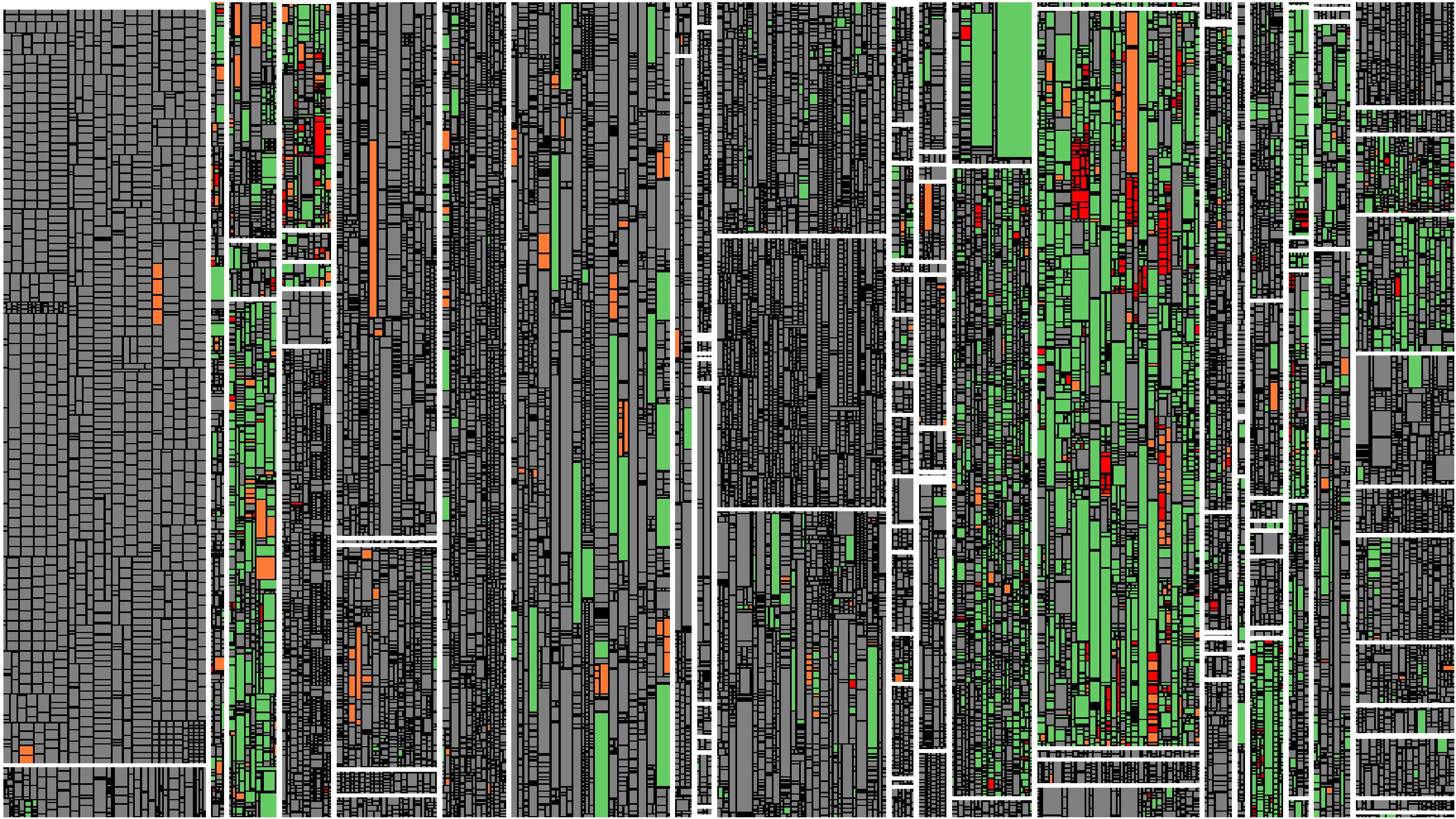


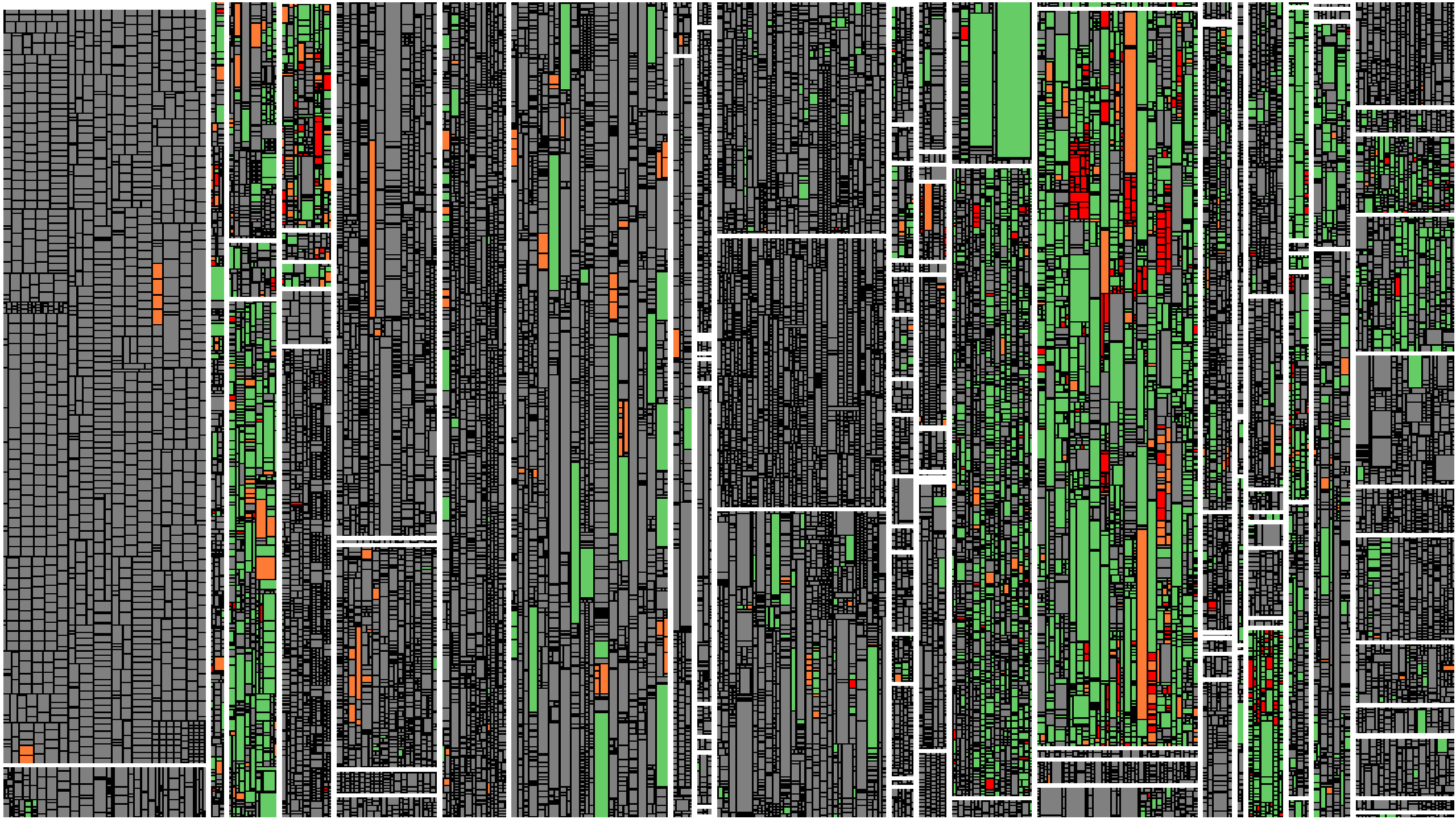


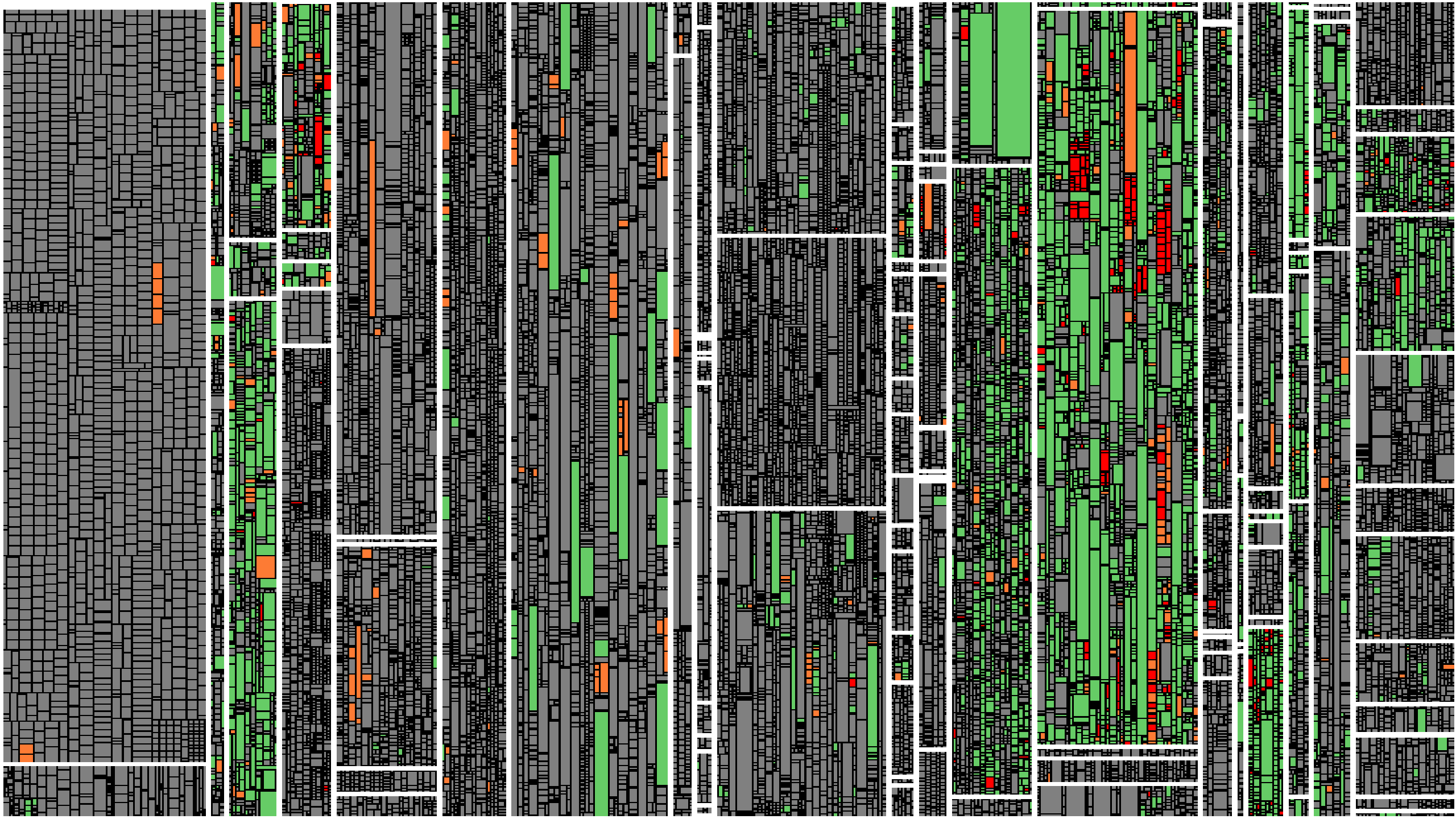


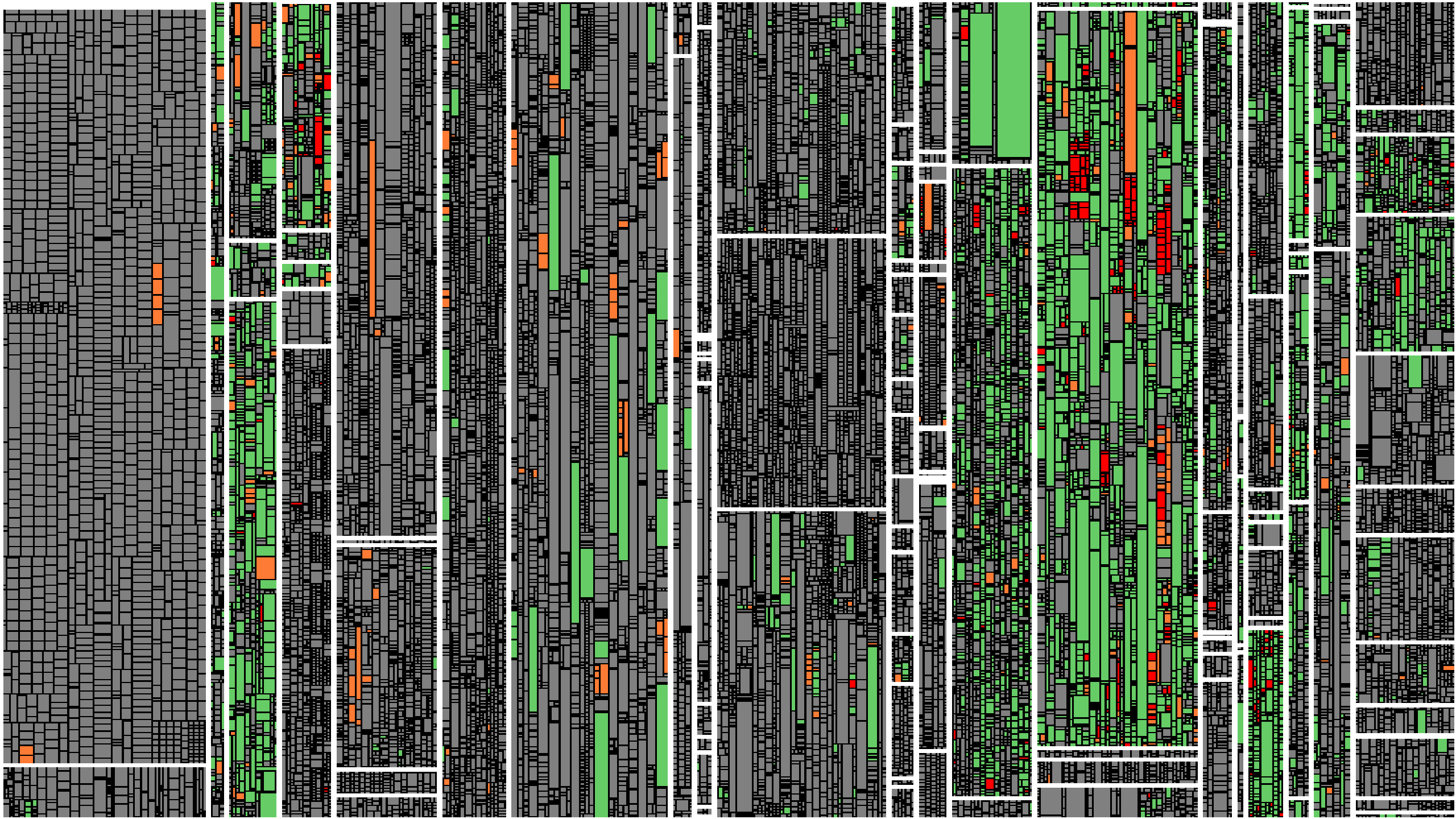


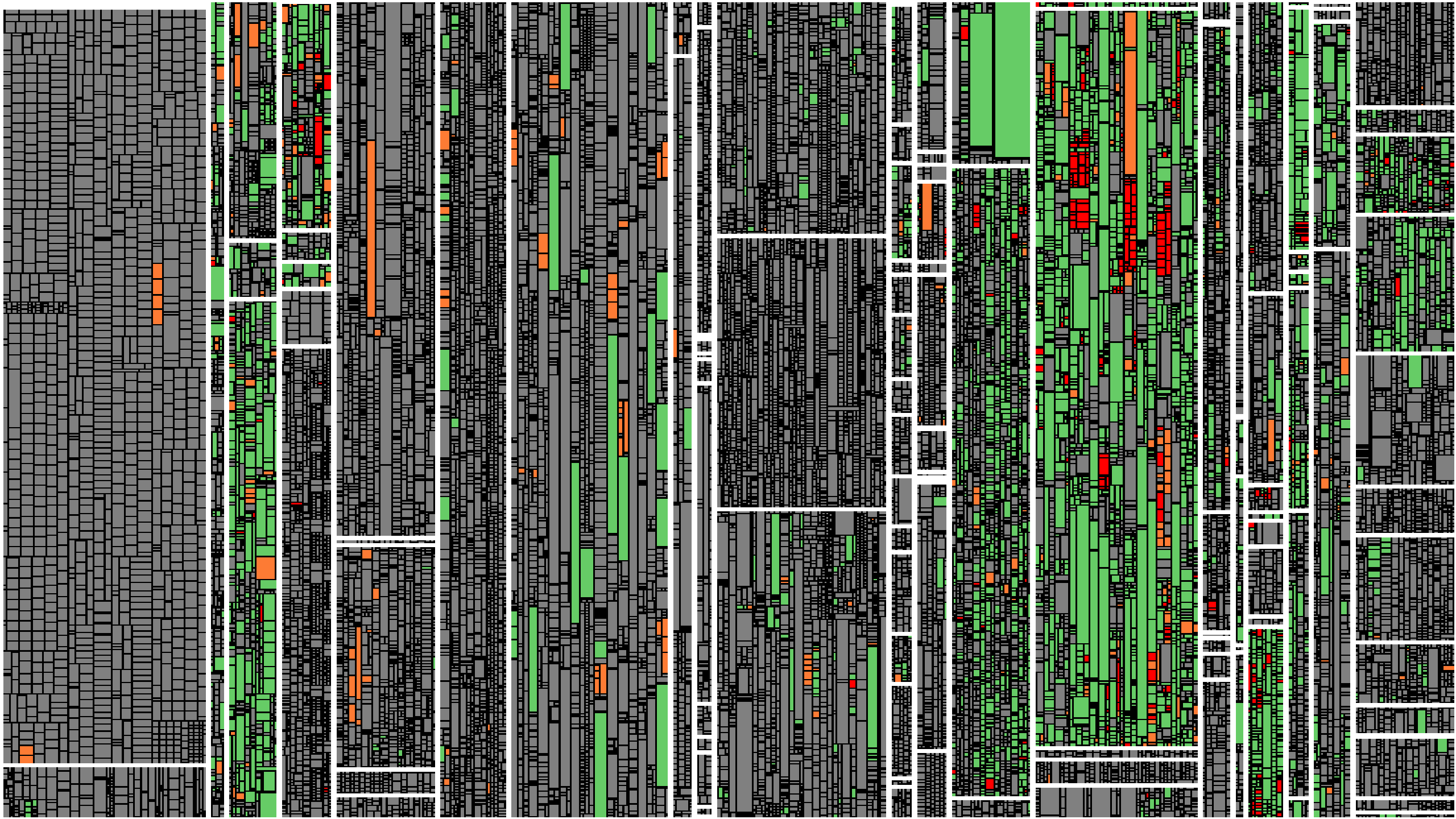


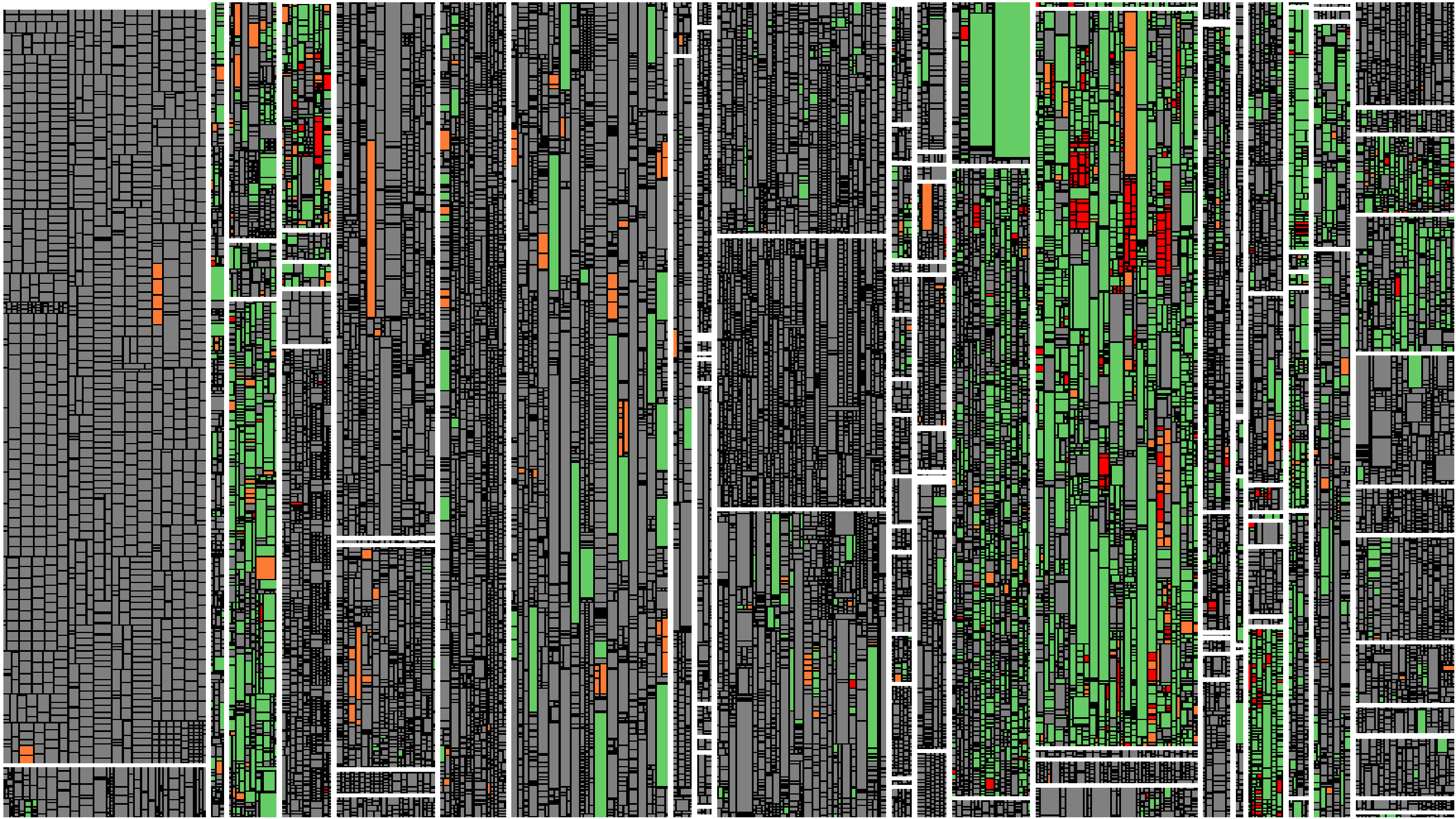


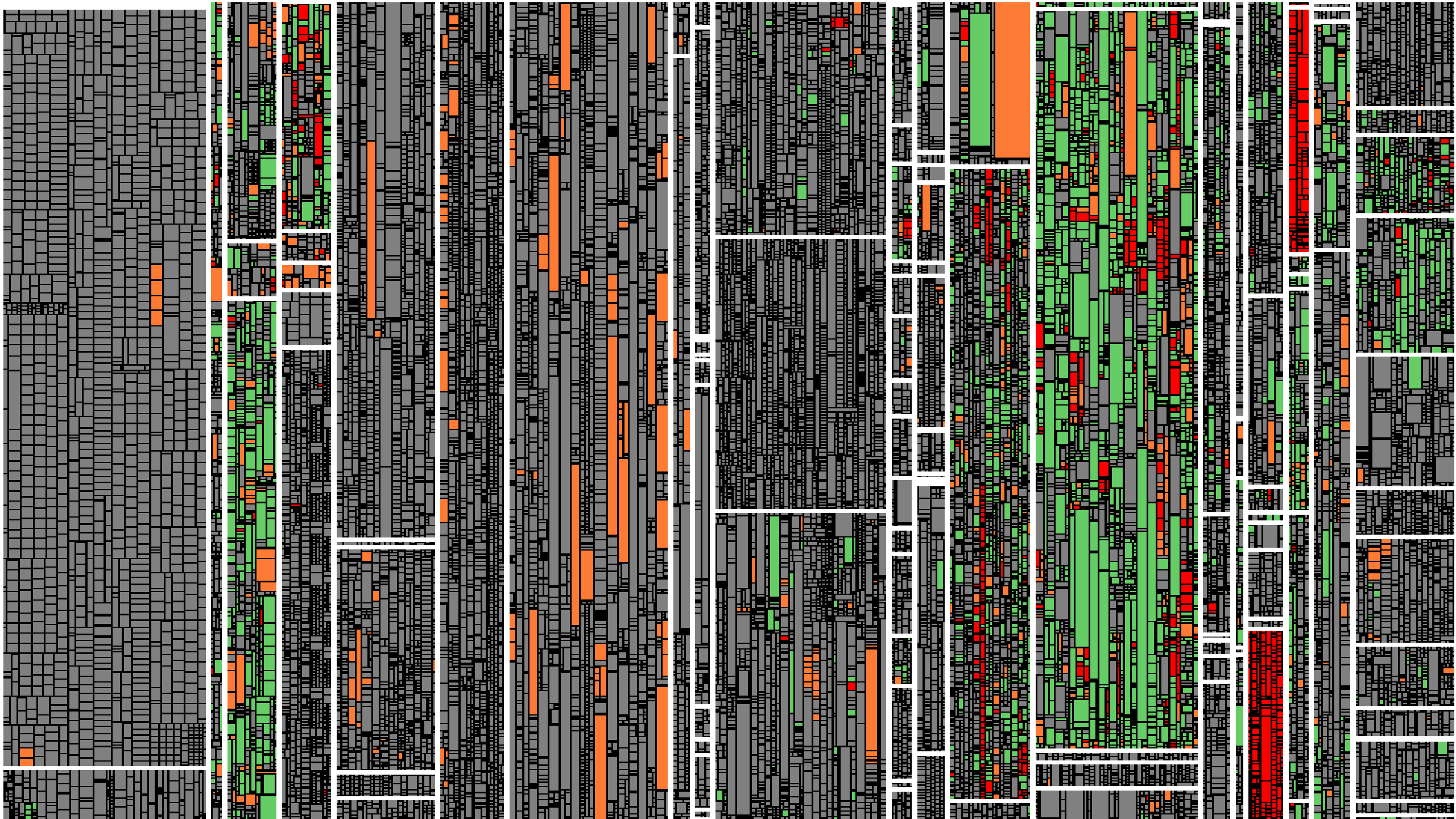




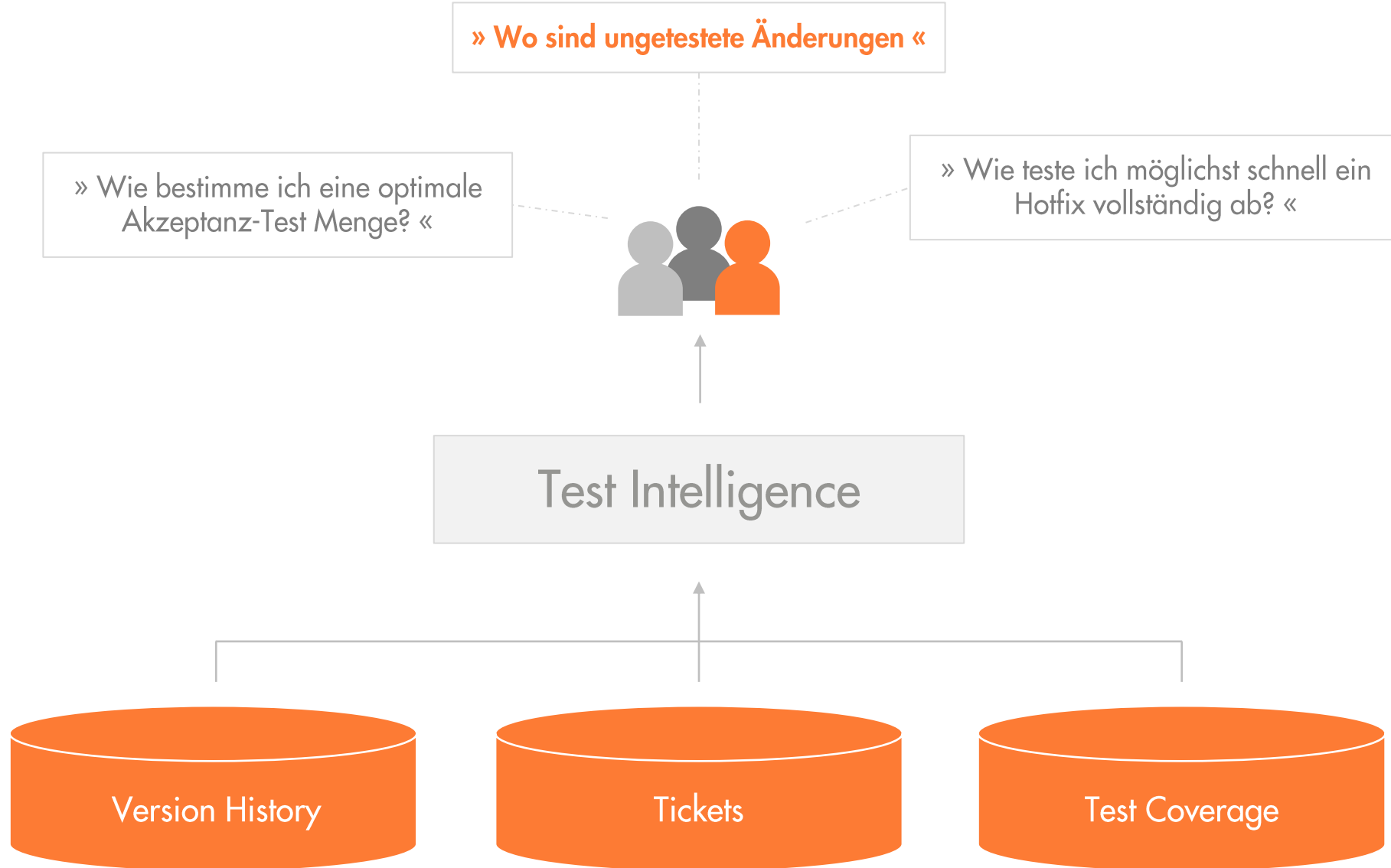


































Issue # <span>▼</span>	Subject	Done		Test Gap
<a href="#">🔗 TS-10549</a>	Undo/Redo for web-based architecture editor	Done		0% 
<a href="#">🔗 TS-10784</a>	Fix long method finding in TaintAnalysisRunner	Done		0% 
<a href="#">🔗 TS-10923</a>	Implement metric 'Nesting Depth' for Simulink	Done		29% 
<a href="#">🔗 TS-11364</a>	External findings are not registered during first upload	Done		14% 
<a href="#">🔗 TS-11942</a>	Manual test coverage upload during development	Done		43% 
<a href="#">🔗 TS-12050</a>	Tool for transferring findings blacklists and tasks	Done		50% 
<a href="#">🔗 TS-12262</a>	Cannot set or alter alias without reanalysis	Done		0% 
<a href="#">🔗 TS-13151</a>	Fetch parent relationship of TFS work items	Done		0% 

Issue # ▾	Subject		Test Gap
<a href="#">TS-14421</a>	Get rid of TestGapSynchronizer block	Done 	0% 
<a href="#">TS-14733</a>	Remove Dataflow blocks	Done 	22% 

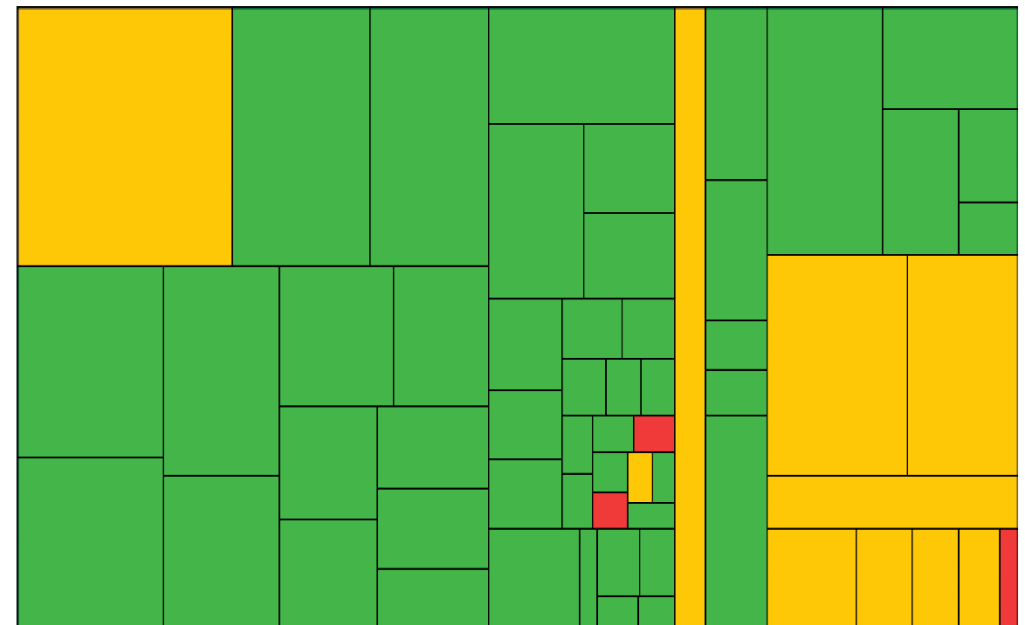
**Done Issue TS-14733 - Remove Dataflow blocks**

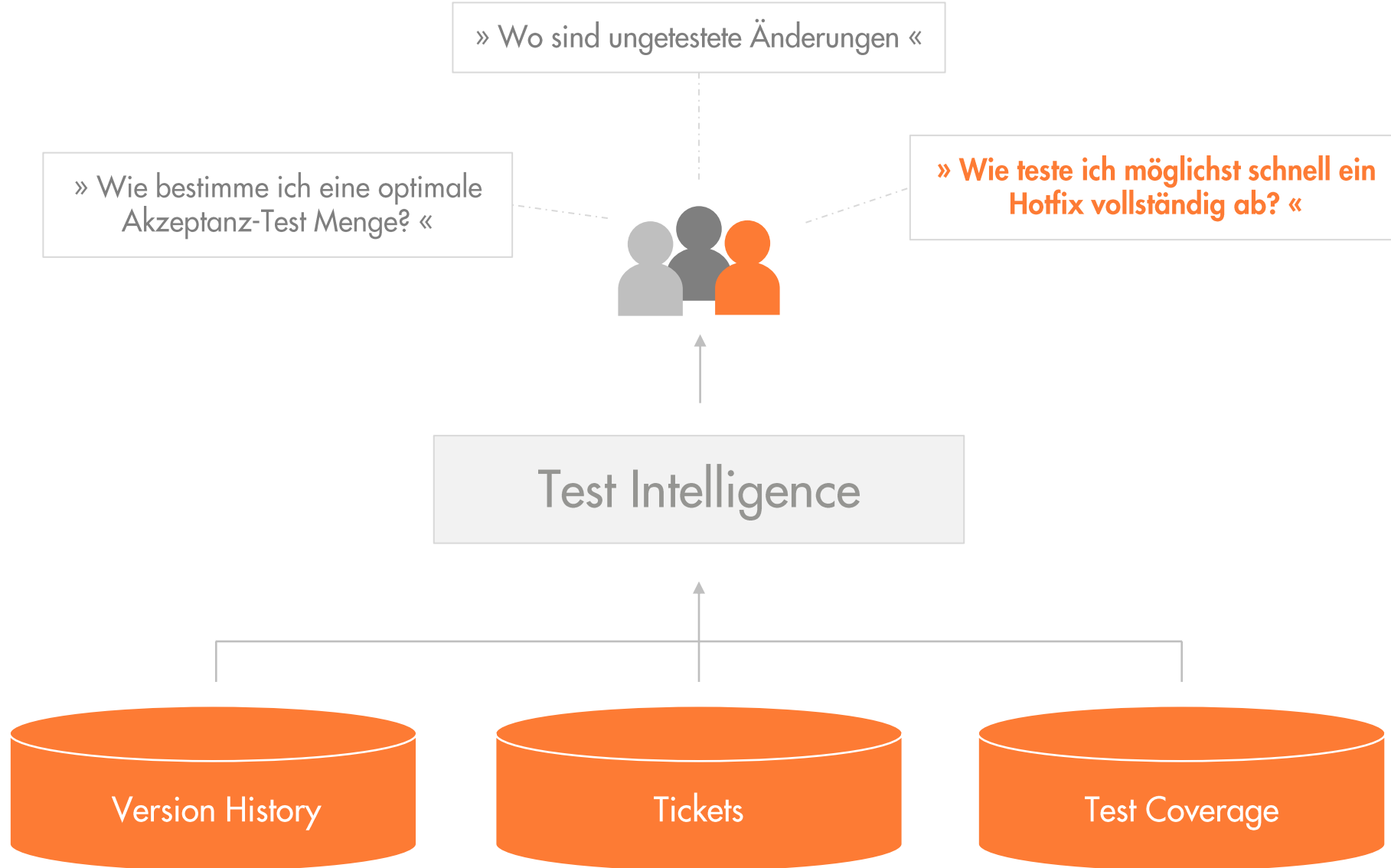
Creator:  (on Apr 06 2018 19:44) Last update: Aug 24 2018 09:32

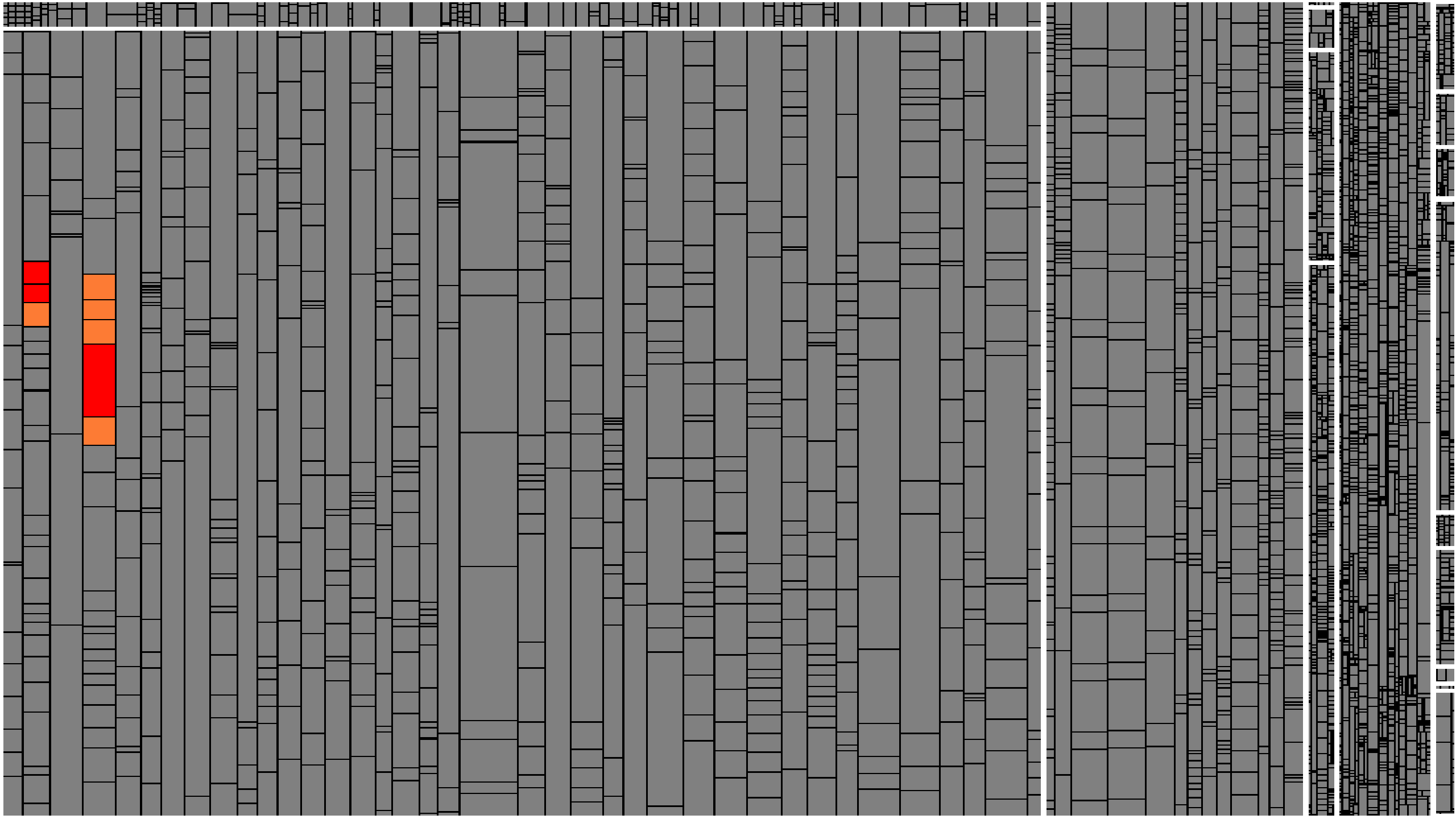
Assignee: 

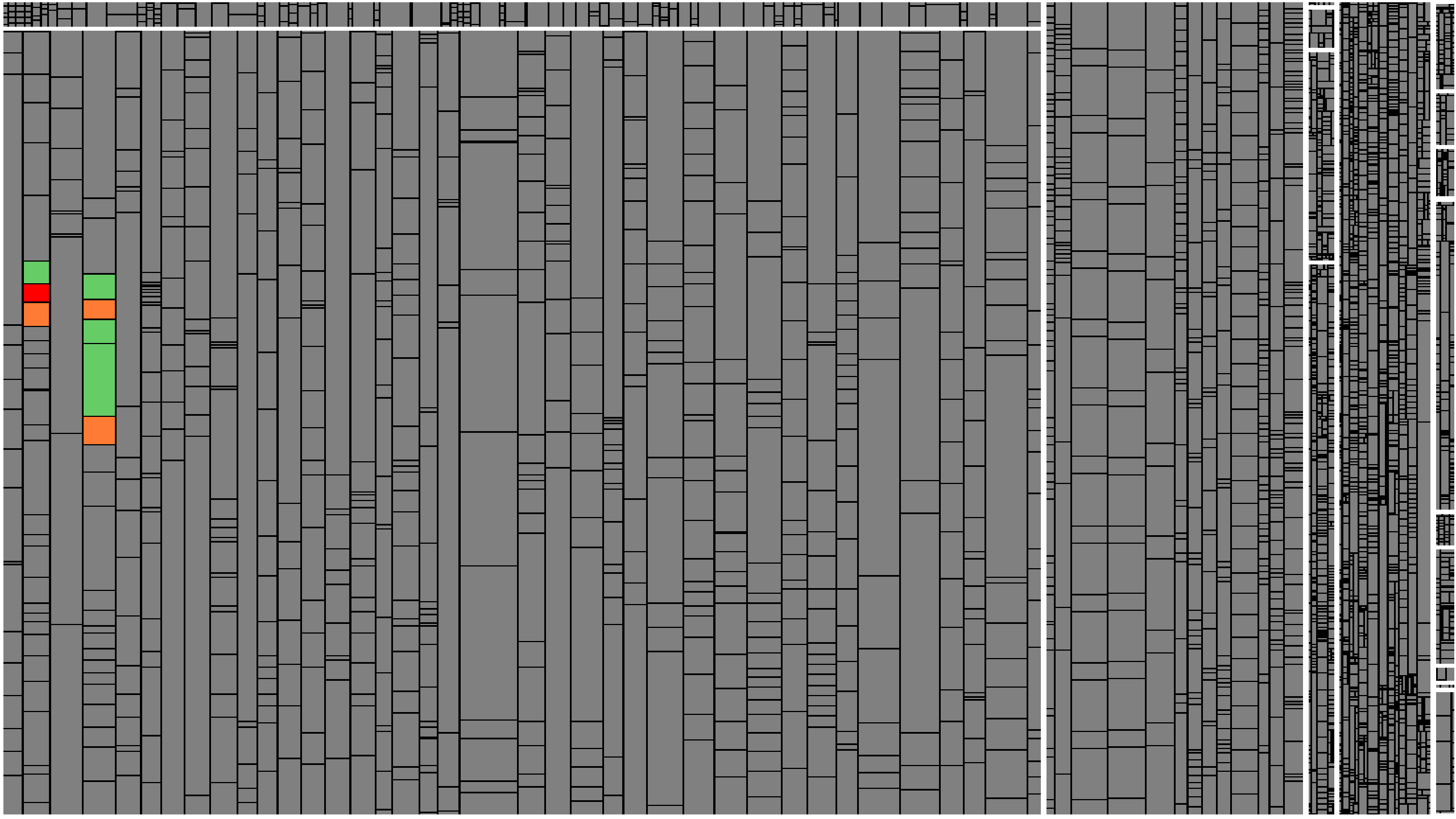
Project	Type	Priority	Resolution	Fix Version
TS	Maintenance	Normal	Green	Teamscale 4.5
Component	Labels	Affected Version	Customer	Customer Issue
Backend	Performance			
Epic Name	Freshdesk URL	Merge Request		
		<a href="https://git.cqse.eu/cqse/teamscale/3621">https://git.cqse.eu/cqse/teamscale/3621</a>		

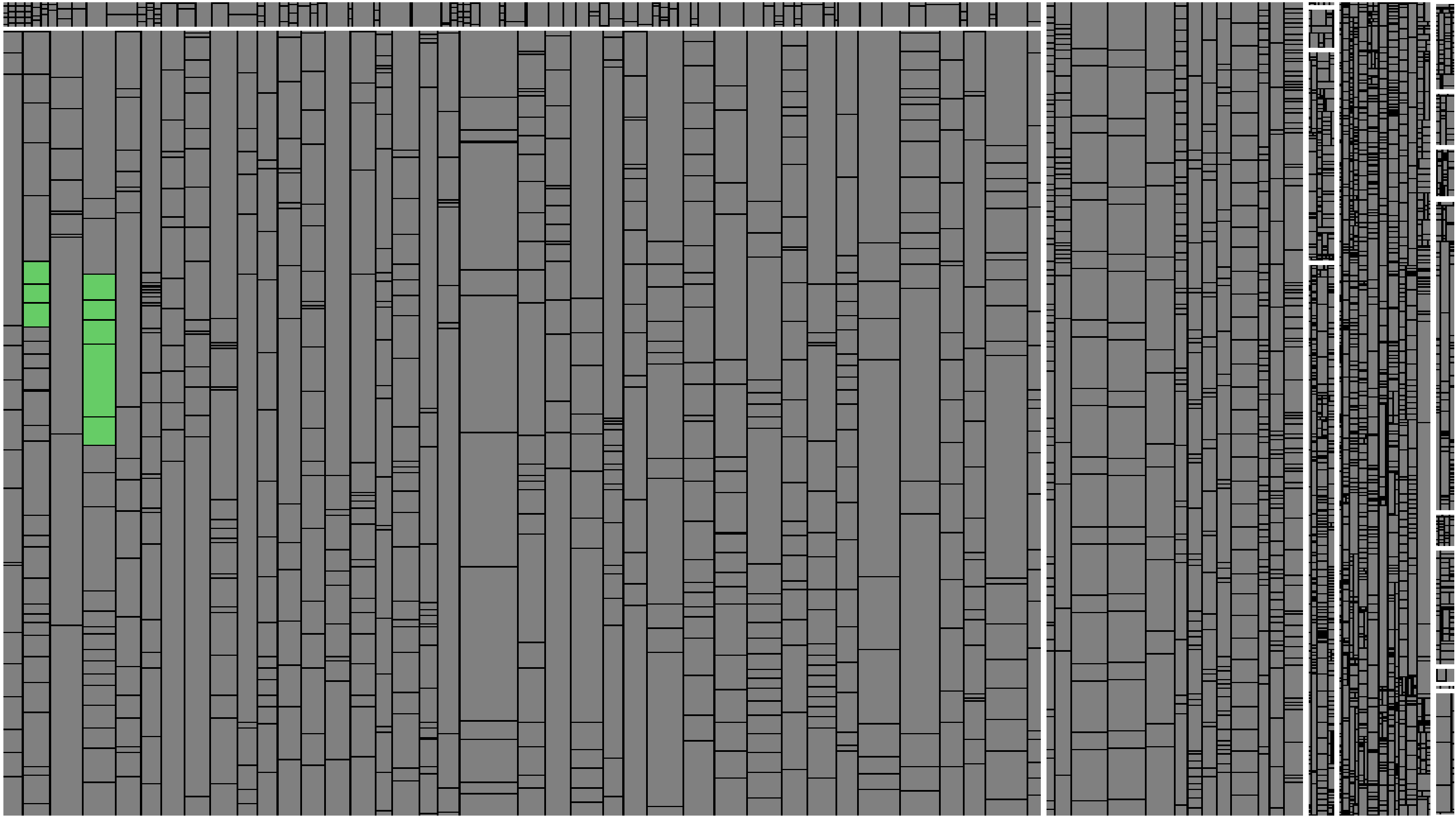
Aug 15 2018 12:37–Now | Test Gap: 22%

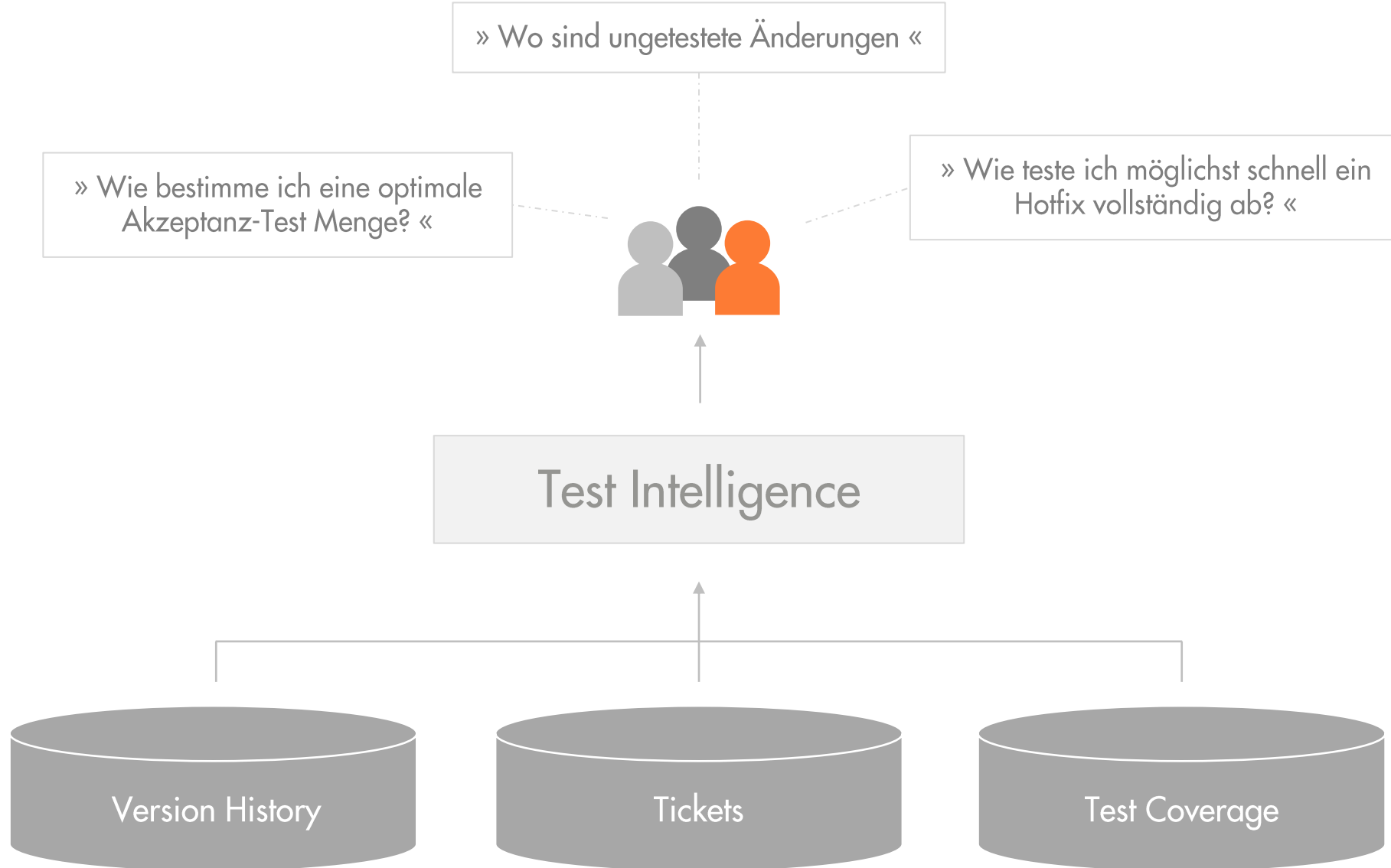














## Test-Gap-Analyse

Ungetestete Änderungen im Quelltext aufdecken



15. Februar  
10:30-12:00 Uhr  
[cqse.eu/tga-232-oopd](https://cqse.eu/tga-232-oopd)



25. April  
10:30-12:00 Uhr  
[cqse.eu/tga-234-oopd](https://cqse.eu/tga-234-oopd)



## Schnelles Feedback trotz langsamer Tests

Testselektion für historisch gewachsene Test-Suites



15. März  
10:30-12:00 Uhr  
[cqse.eu/ts-233-oopd](https://cqse.eu/ts-233-oopd)



14. Juni  
10:30-12:00 Uhr  
[cqse.eu/ts-236-oopc](https://cqse.eu/ts-236-oopc)



# Kontakt – Ich freue mich auf Fragen 😊



Dr. Elmar Jürgens · [juergens@cqse.eu](mailto:juergens@cqse.eu) · +49 179 675 3863

CQSE GmbH  
Centa-Hafenbrädl-Str 59  
81249 München  
[www.cqse.eu](http://www.cqse.eu)

