

Did We Test Our Changes?

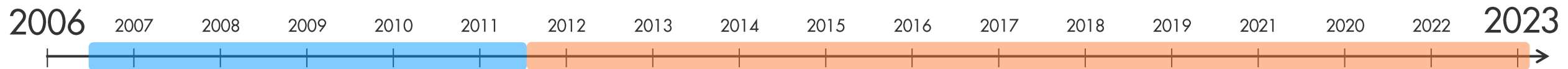
Preventing the Release of Error-Prone Code



TUM



CQSE



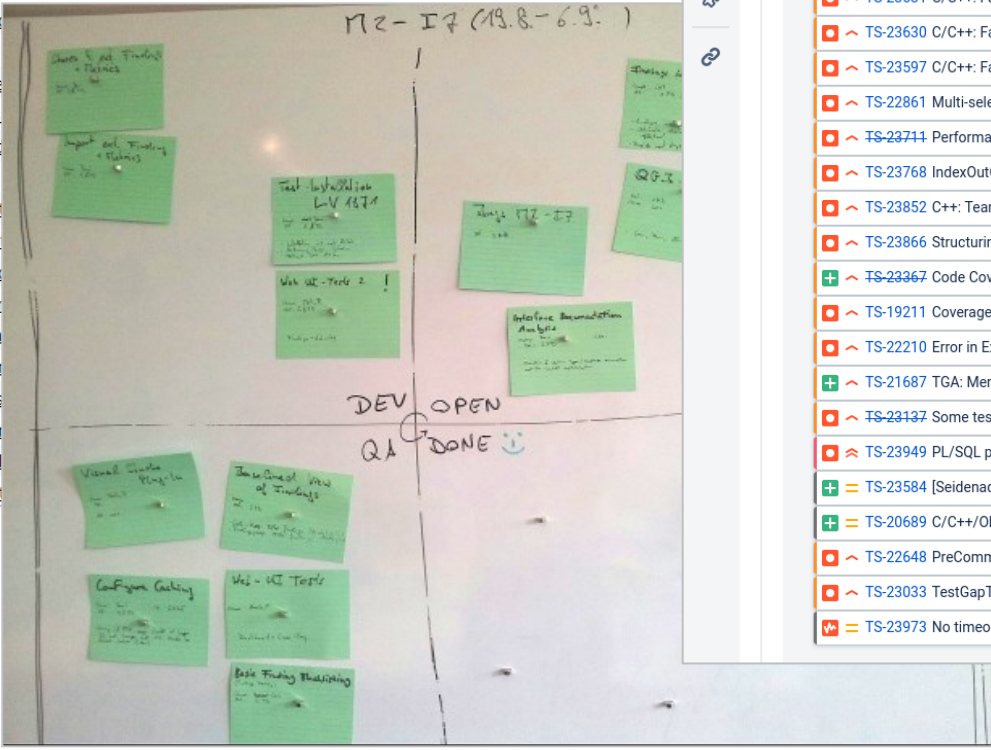
Given my limited resources, how do I get the most out of my tests?

Hide Search Description

Status: UNCONFIRMED, NEW, ASSIGNED, REOPENED Product: clone Component: clone Alias: clone

83 bugs found.

ID	Summary
3382	Remove old filtering framework
4608	Improve progress monitoring for clone detection
3613	Make MetricOverview more flexible w.r.t. to database table names
3811	Incorrect offset reported by two-phased scanners
4610	Clone compare view does not open clone pair at correct position
1000	Integrate ConQAT and Clone Detection to allow for flexible filtering
1013	Rebuild CloneDetective Pipeline as ConQAT processors
1014	Integrate cdct, clonetective and bundles into single source
872	Remove ClassFileCollector
2397	Review cd_core
1824	More fine grained file e
1945	Consolidate identifiers
2199	Automatically close old
2216	Clone Tracking
2221	Consolidate normaliza
2223	Problem with gap posi
2327	Clean up old clone trac
2328	Add clone tracing for r
2367	Compute type of chan
2417	Create batch file gener
2428	Implement constraints
2431	Create Clone Detection
2439	Unload detection resul
2476	Improve parameteriza



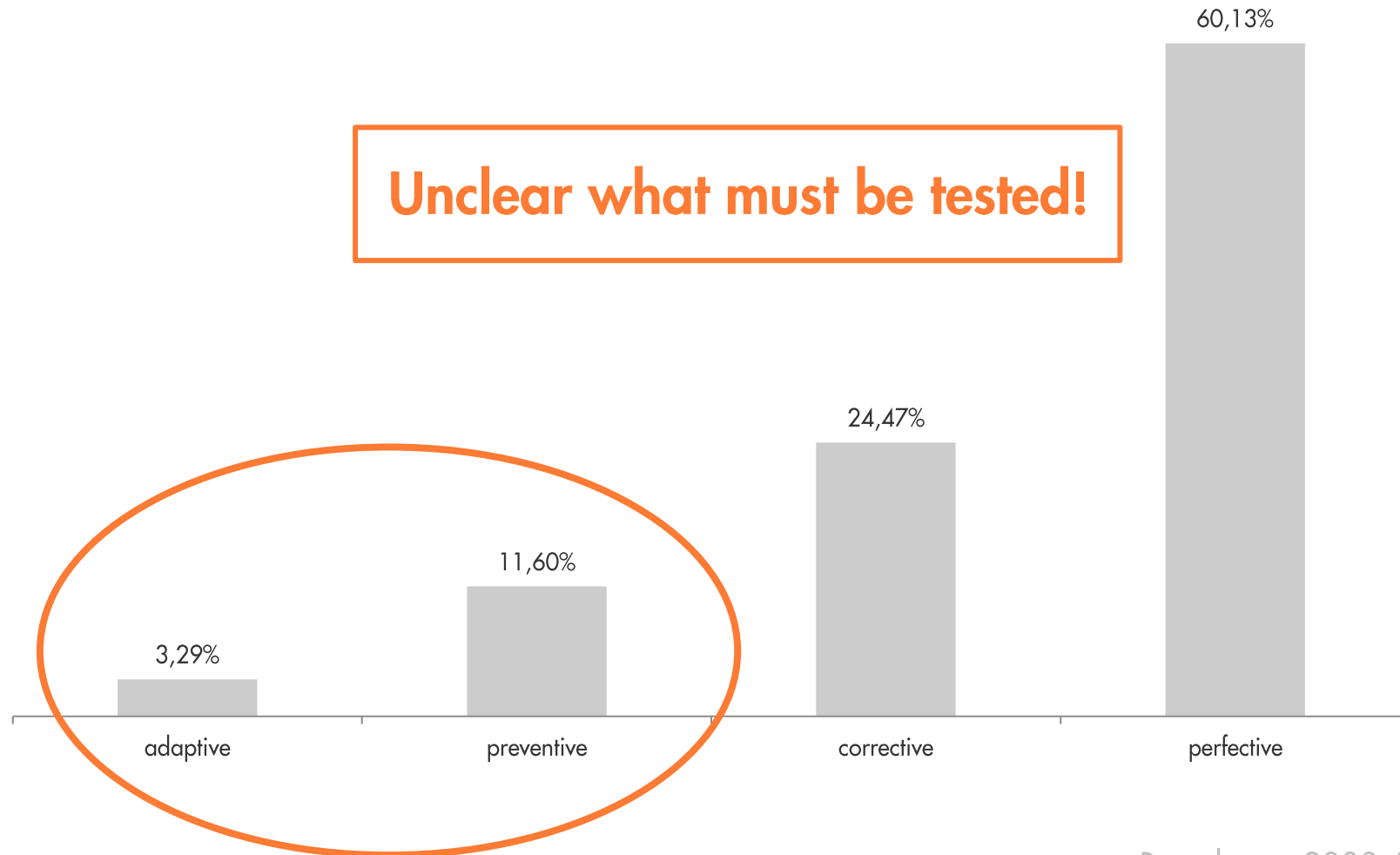
Jira Software

Denali Next Backlog

Denali Sprint 4 20 issues ACTIVE

- TS-21509 TGA Trend numbers do not match TGA treemap and Test perspective
- TS-23631 C/C++: False-positive with "Null pointer dereference" check when variables have the same names as struct members
- TS-23630 C/C++: False-positive with Unused parameter (dataflow) check when using __unused
- TS-23597 C/C++: False-positive with "Unused variable or parameter" check (dataflow) when writing
- TS-22861 Multi-selection pre-commit analysis (all projects in same SVN checkout) sometimes misses files
- TS-23714 Performance of the process-based clang-tidy integration is low
- TS-23768 IndexOutOfBoundsException in check 'Don't assign to wider essential type'
- TS-23852 C++: Teamscale preprocessor confuses macro call with constructor call
- TS-23866 StructuringAnalysis: Expecting metric value for edited path
- TS-23367 Code Coverage for Go (implement report parser)
- TS-19211 Coverage upload not integrated when backup is downloaded during running analysis and before upload is processed
- TS-22210 Error in ExternalAnalysisResultsPersister: Predecessor commits [...] can't come after commit ...
- TS-21687 TGA: Merge Commits should show method changes
- TS-23137 Some tests fail with "Global Role Assignments" tab not clickable
- TS-23949 PL/SQL parser: define not parsed correctly
- TS-23584 [Seidenader EPS] Arch Editor Implement "Focus on transitive dependencies"
- TS-20689 C/C++/Objective-C: Check for correct formatting string in log statement
- TS-22648 PreCommitServiceTest.testDeletingFile fails/flickers
- TS-23033 TestGapTreeMapWidgetTest.testCR20004 is flickering consistently
- TS-23973 No timeouts for clang-tidy processes

CRs in Maintenance

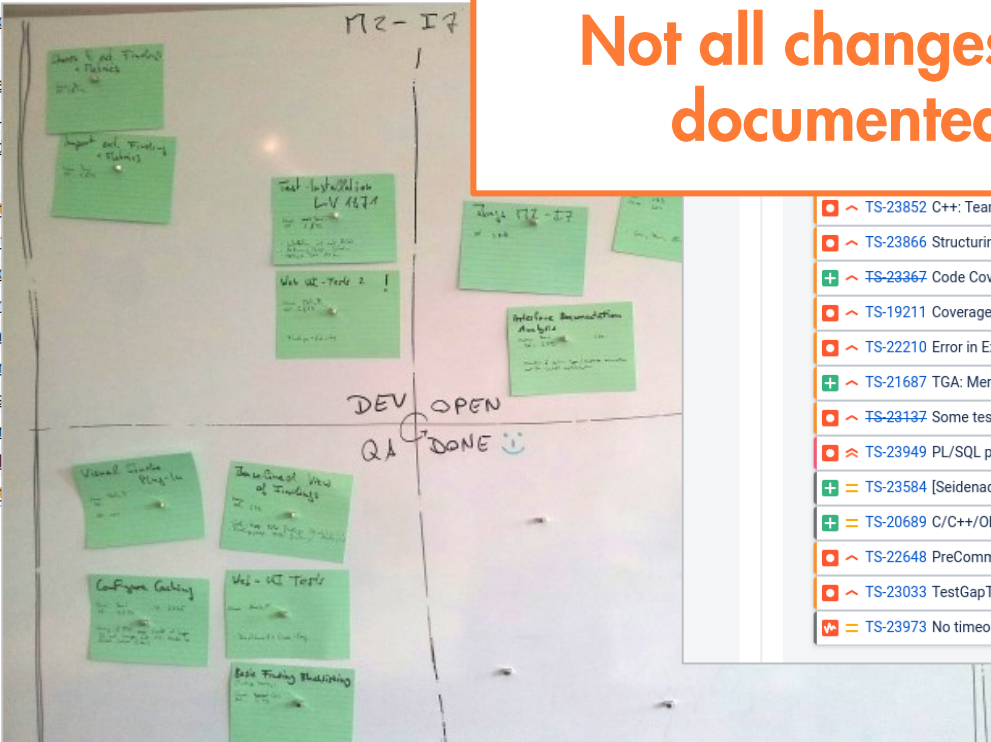


Based on ~ 2000 tickets from our issue tracker

Hide Search Description
Status: UNCONFIRMED, NEW, ASSIGNED, REOPENED Product: clone Component: clone Alias: clone

83 bugs found.

ID	Summary
3382	Remove old filtering framework
4608	Improve progress monitoring for clone detection
3613	Make MetricOverview more flexible w.r.t. to database table names
3811	Incorrect offset reported by two-phased scanners
4610	Clone compare view does not open clone pair at correct position
1000	Integrate ConQAT and Clone Detection to allow for flexible filtering
1013	Rebuild CloneDetective Pipeline as ConQAT processors
1014	Integrate cdct, clonetective and bundles into single source
872	Remove ClassFileCollector
2397	Review cd_core
1824	More fine grained file e
1945	Consolidate identifiers
2199	Automatically close old
2216	Clone Tracking
2221	Consolidate normaliza
2223	Problem with gap posi
2327	Clean up old clone trac
2328	Add clone tracing for r
2367	Compute type of chan
2417	Create batch file gener
2428	Implement constraints
2431	Create Clone Detection
2439	Unload detection resul
2476	Improve parameteriza



Not all changes are documented!

Jira Software Dashboards Projects Issues Boards Create

Denali Next Backlog

Denali Sprint 4 20 issues ACTIVE

- TS-21509 TGA Trend numbers do not match TGA treemap and Test perspective
- TS-23852 C++: Teamscale preprocessor confuses macro call with constructor call
- TS-23866 StructuringAnalysis: Expecting metric value for edited path
- TS-23367 Code Coverage for Go (implement report parser)
- TS-19211 Coverage upload not integrated when backup is downloaded during running analysis and before upload is processed
- TS-22210 Error in ExternalAnalysisResultsPersister: Predecessor commits [...] can't come after commit ...
- TS-21687 TGA: Merge Commits should show method changes
- TS-23137 Some tests fail with "Global Role Assignments" tab not clickable
- TS-23949 PL/SQL parser: define not parsed correctly
- TS-23584 [Seidenader EPS] Arch Editor Implement "Focus on transitive dependencies"
- TS-20689 C/C++/Objective-C: Check for correct formatting string in log statement
- TS-22648 PreCommitServiceTest.testDeletingFile fails/flickers
- TS-23033 TestGapTreeMapWidgetTest.testCR20004 is flickering consistently
- TS-23973 No timeouts for clang-tidy processes

How many changes are untested?

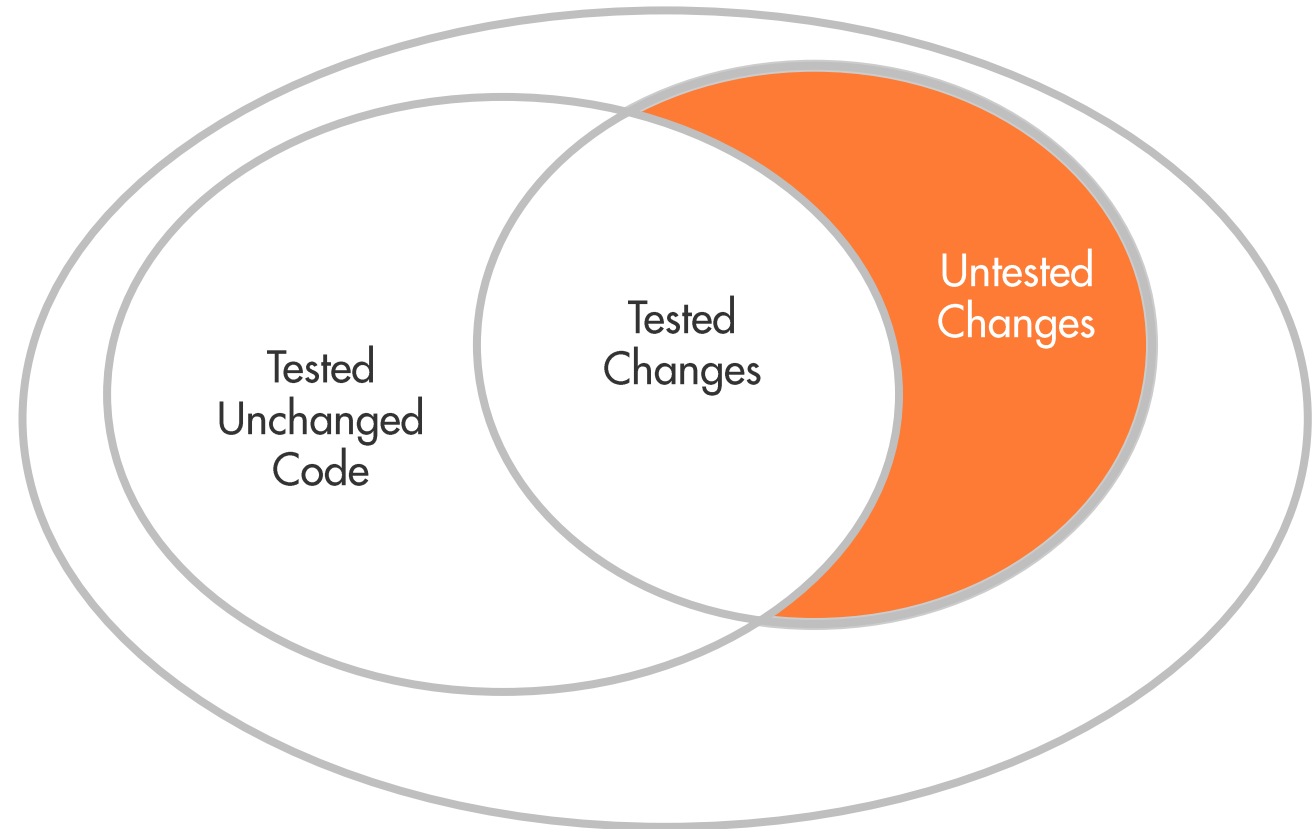
Study: C# System @MunichRe

Release A:

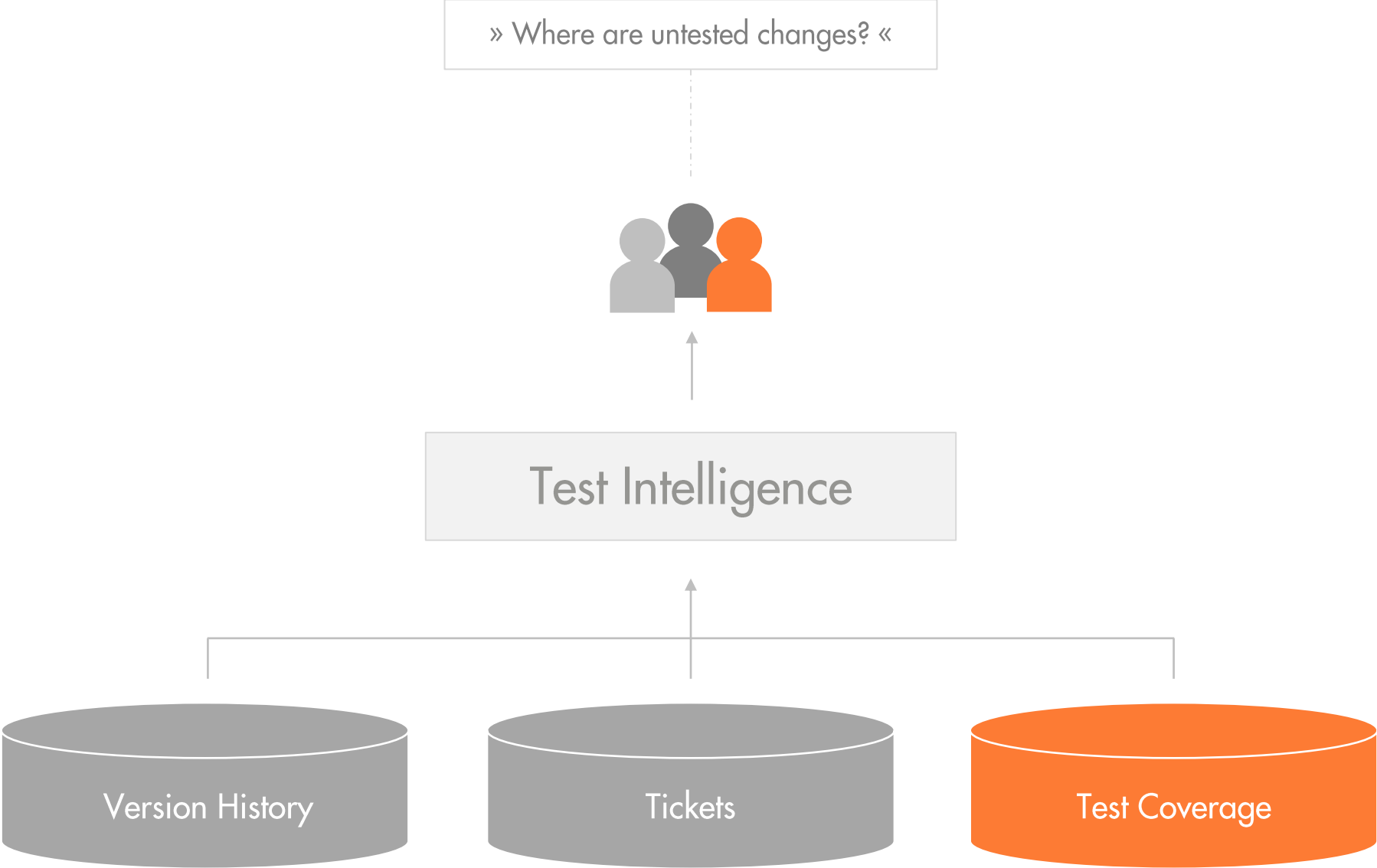
15% Code new/changed,
>50% untested

Release B:

15% Code new/changed,
>60% untested



Probability of production bugs is 5x higher for untested changes!



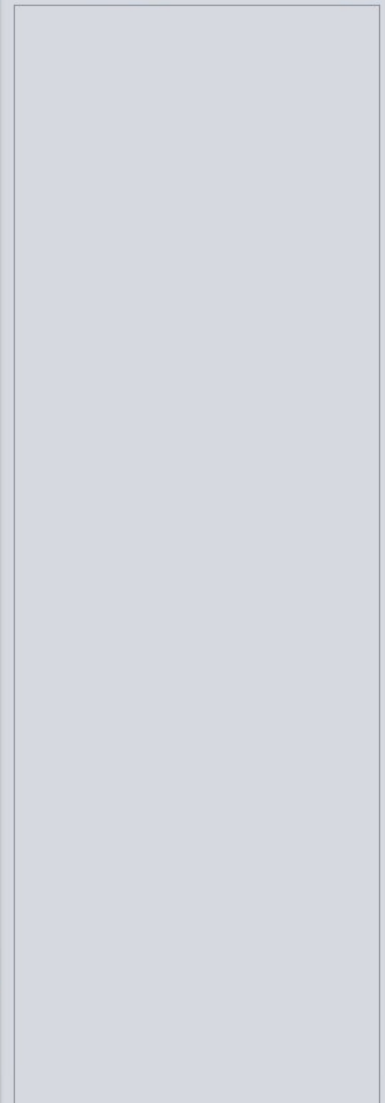
Sample Only the Active Layer/Mask

Untitled1 x Picture1.png x



Layers

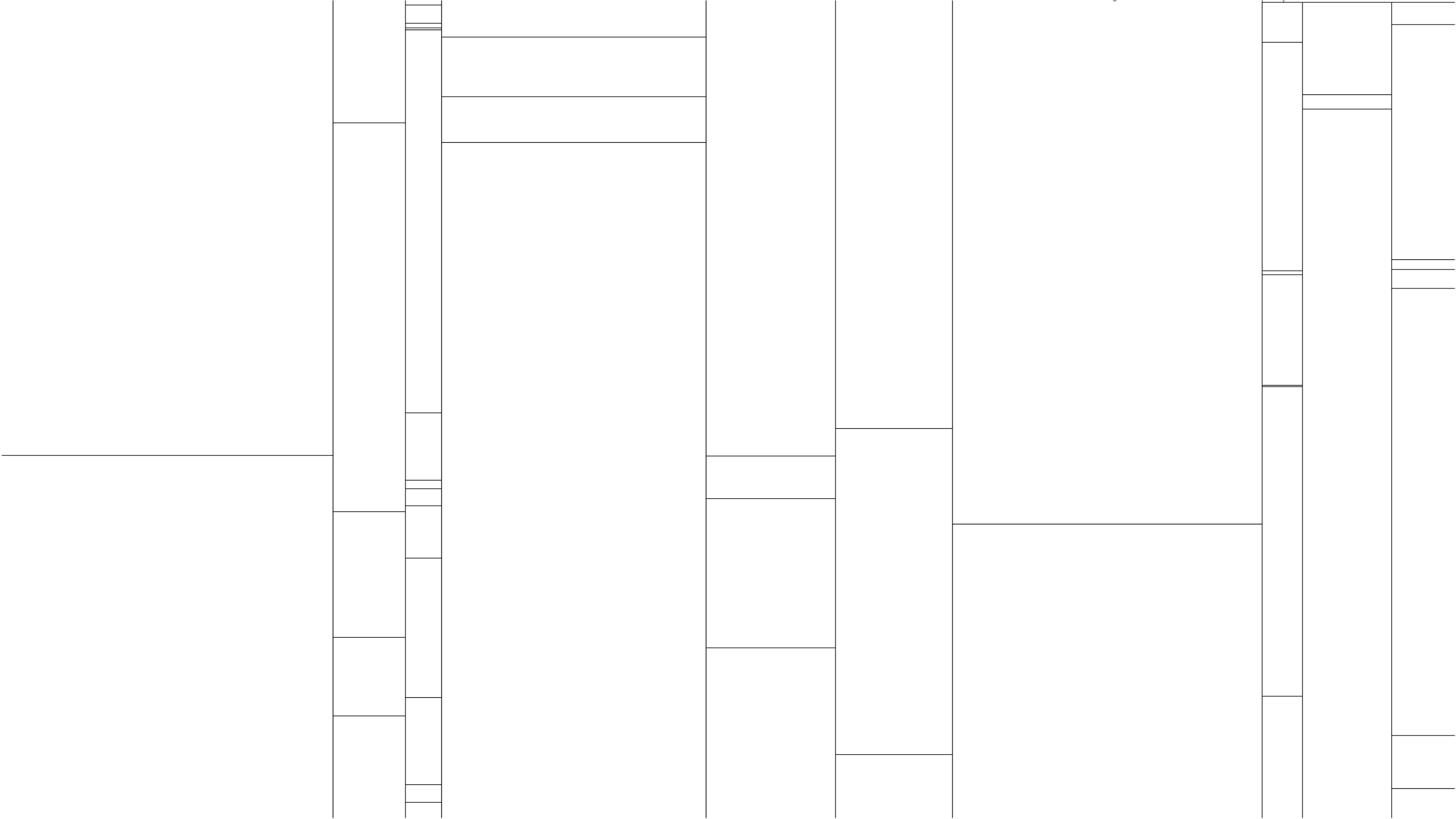
Opacity: 100 % Normal

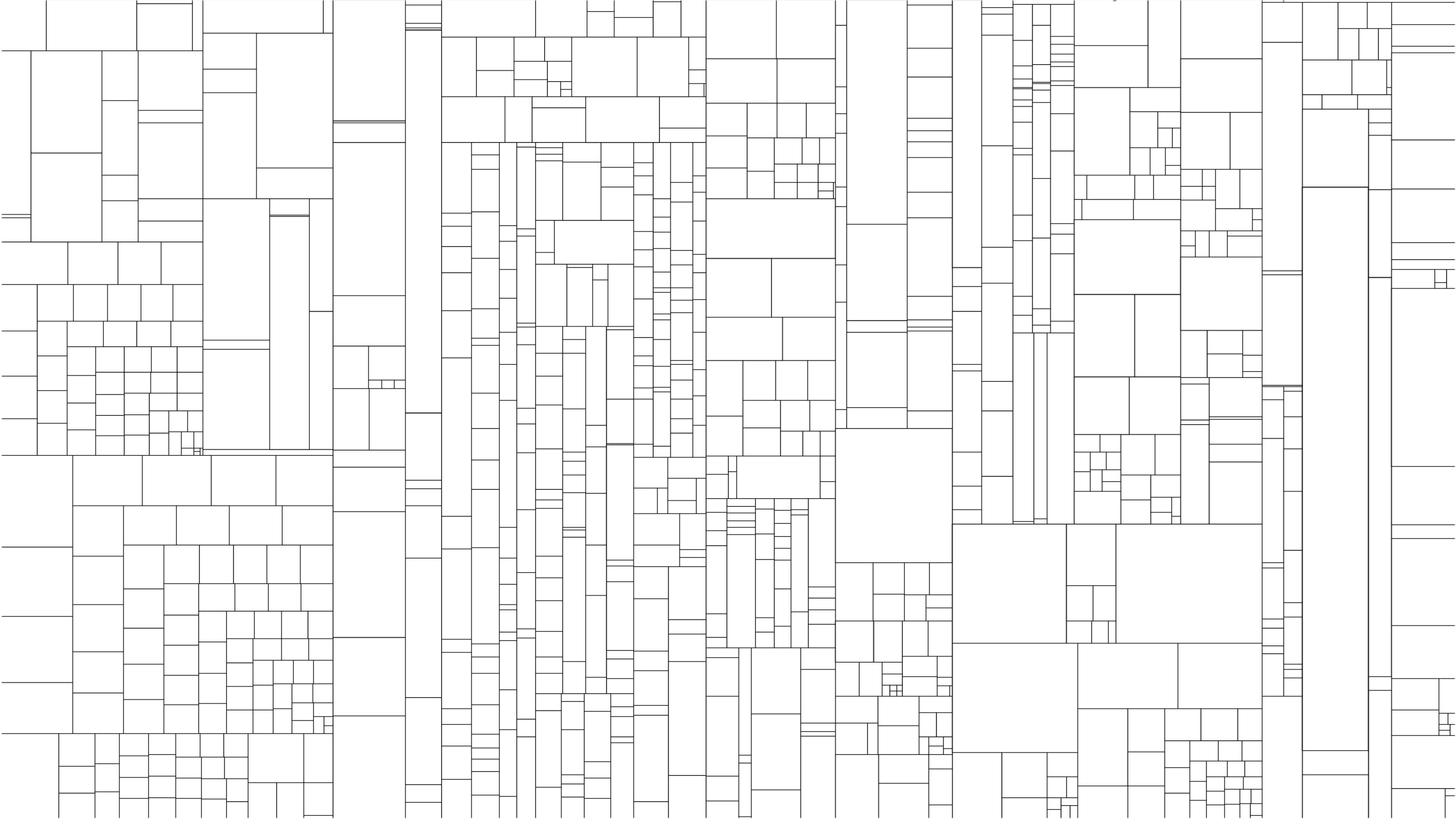


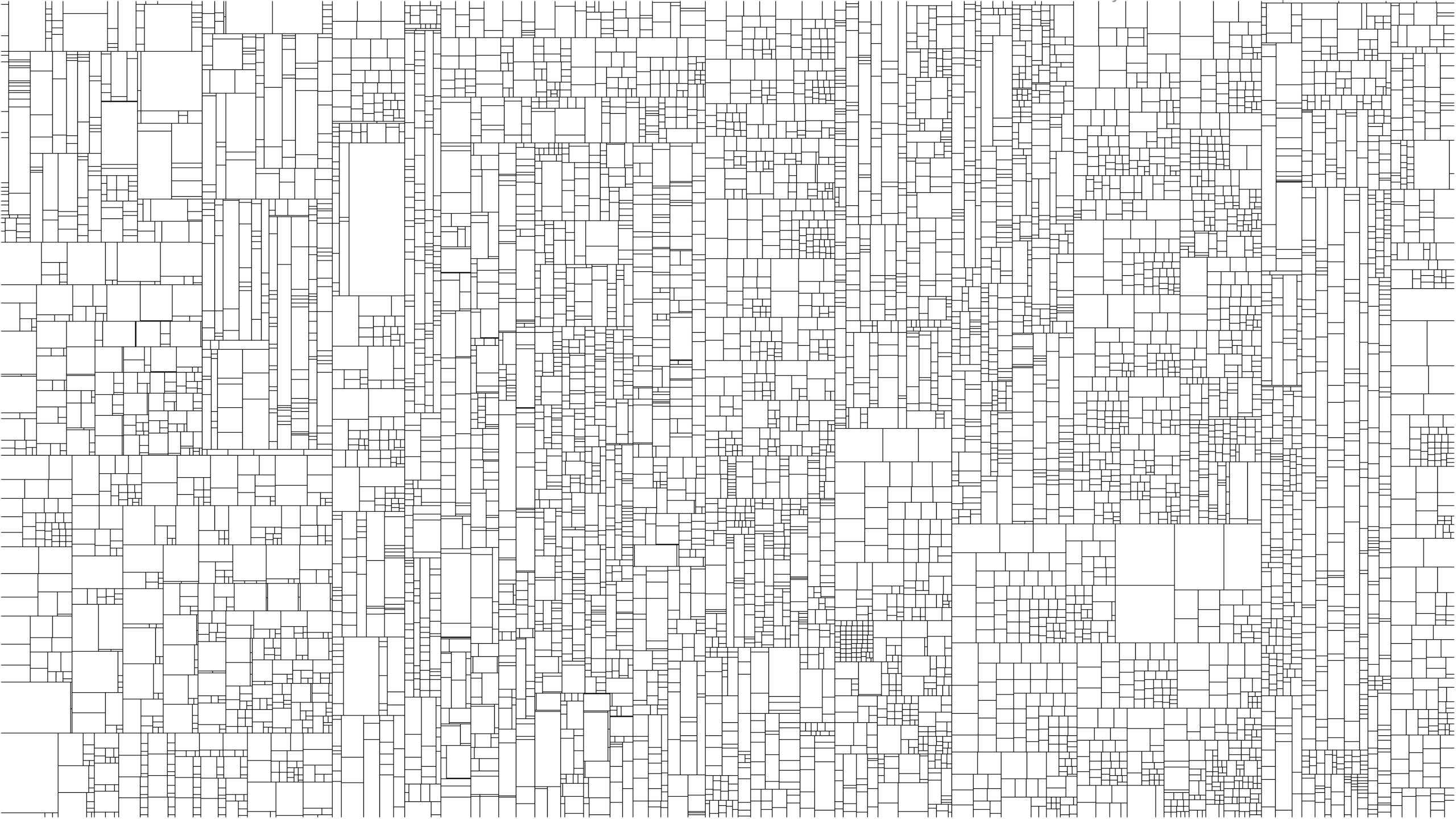
layer 1

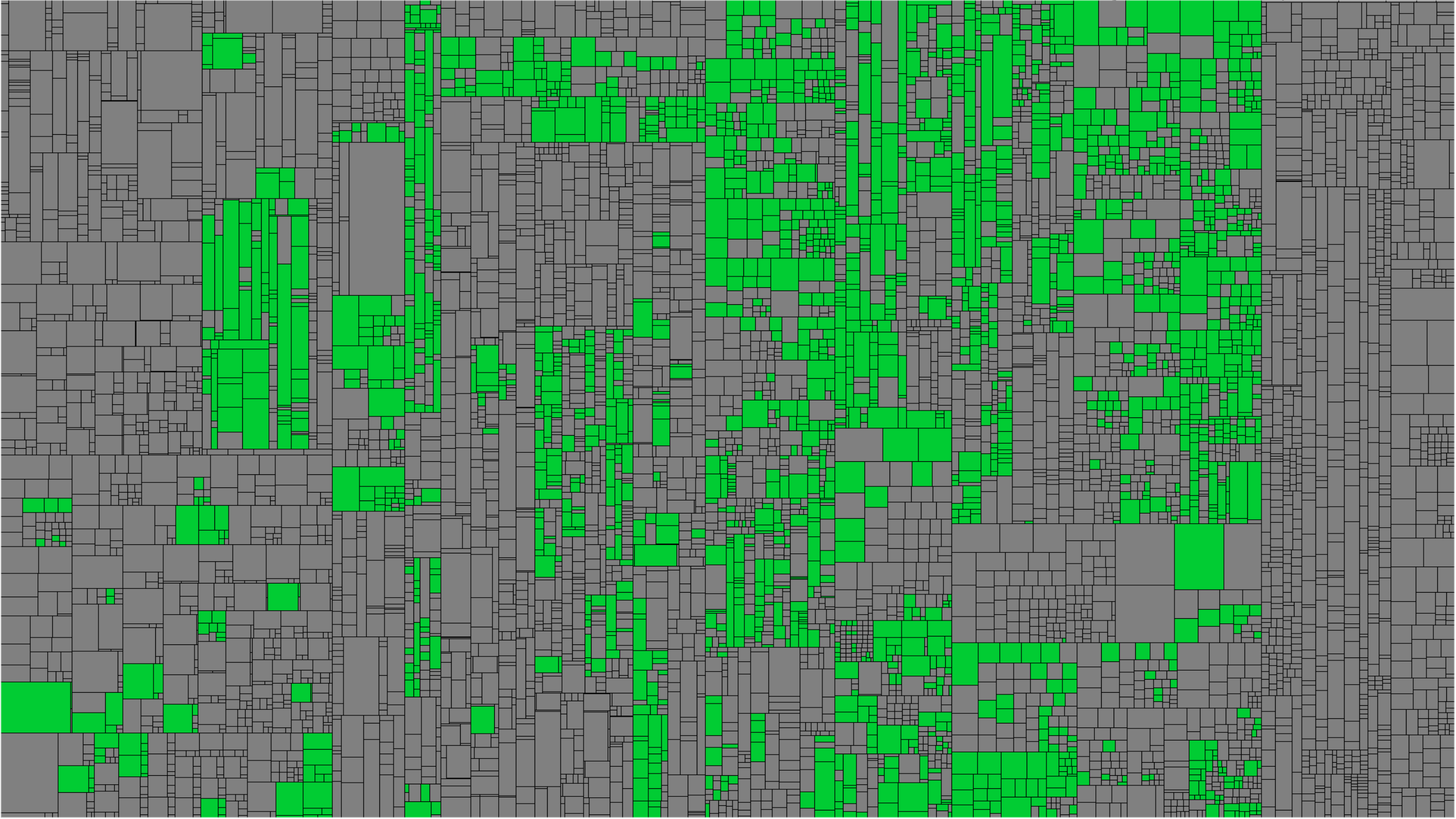
+ - [] T

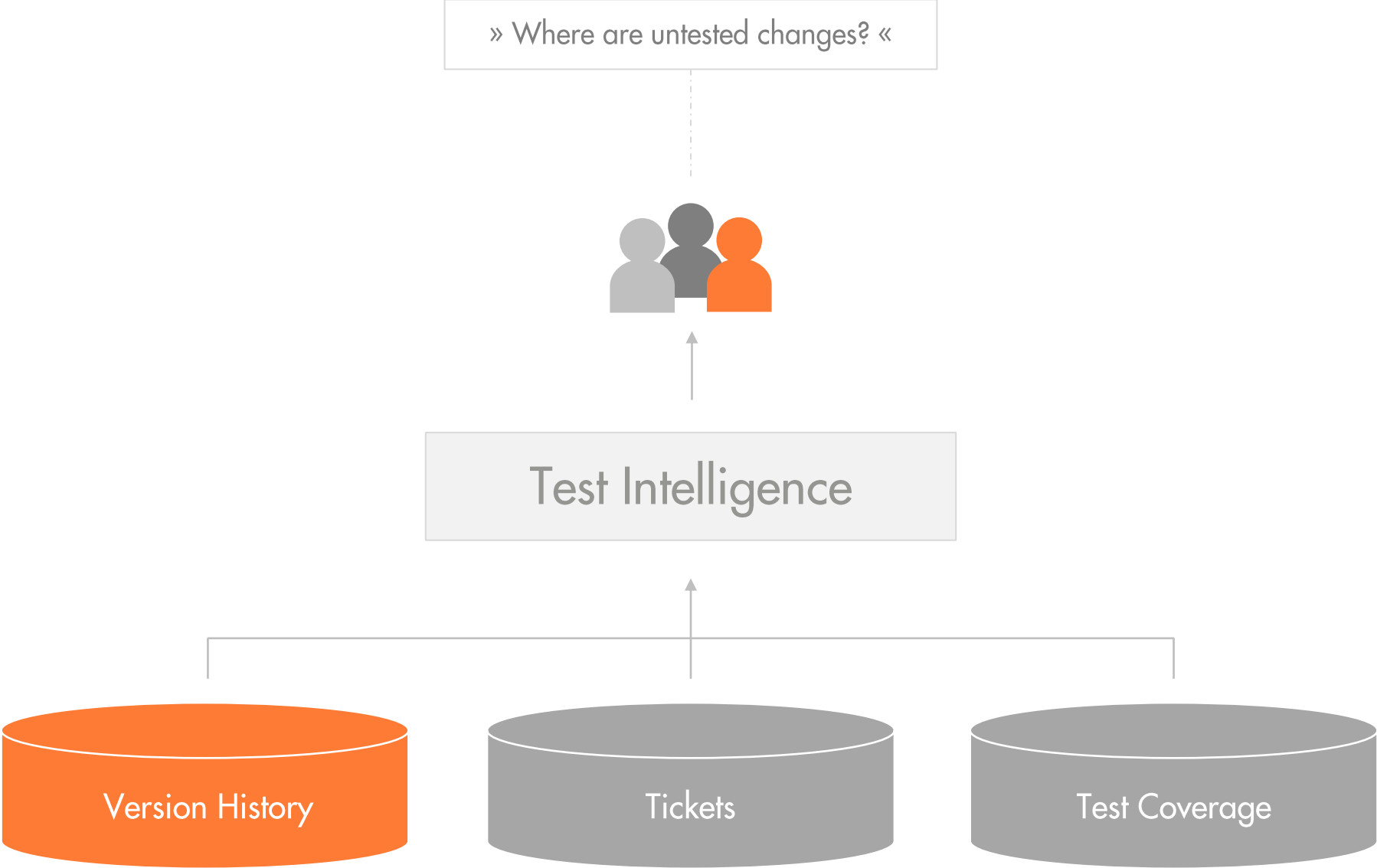
```
114
115i     private static void createAndShowGUI(String[] args) {
116         assert calledOnEDT() : threadInfo();
117
118         Messages.setMsgHandler(new GUIMessageHandler());
119
120         // GlobalKeyboardWatch.showEventsSlowerThan(100, TimeUnit.MILLISECONDS);
121
122         Theme theme = 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
```

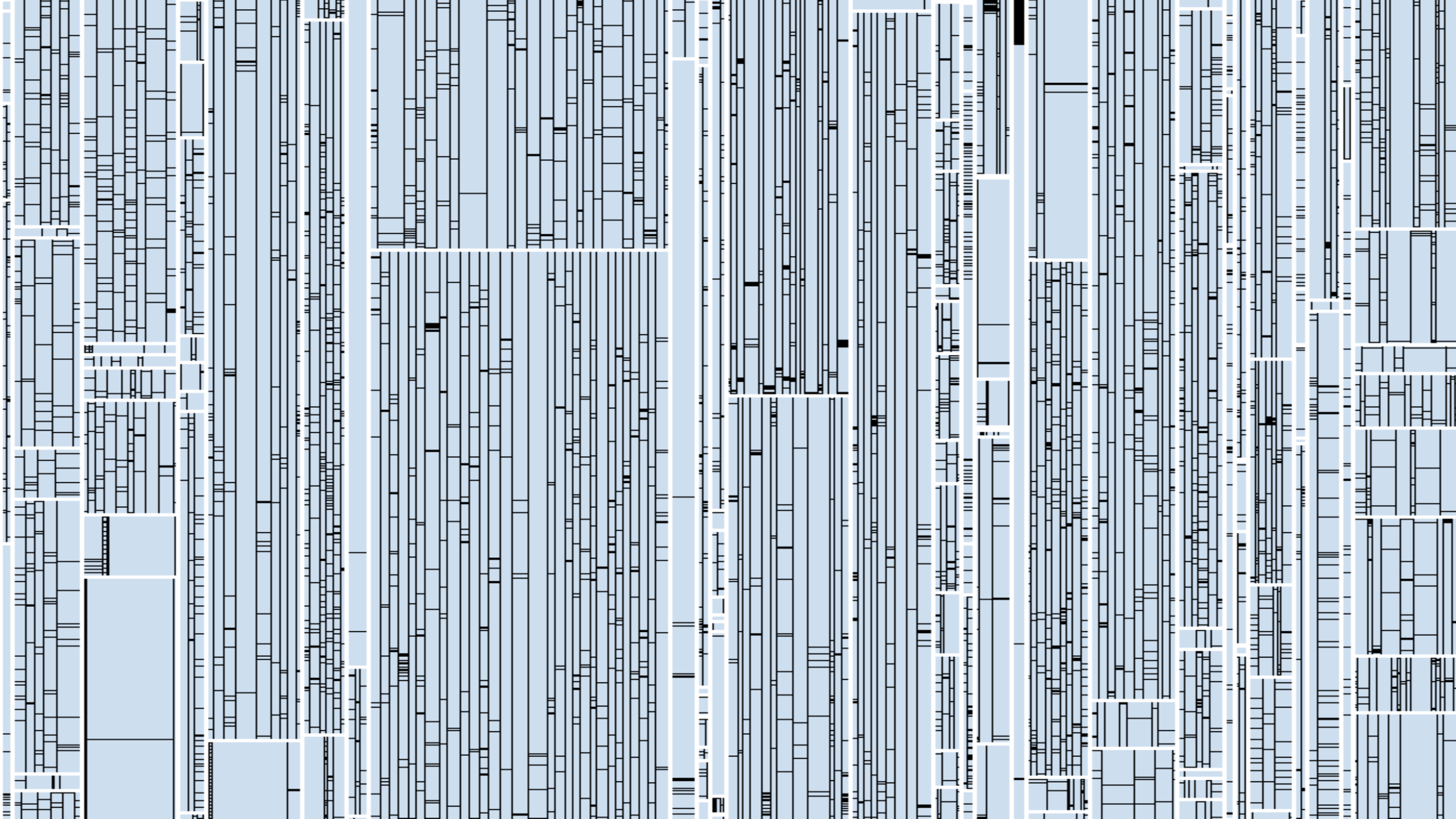


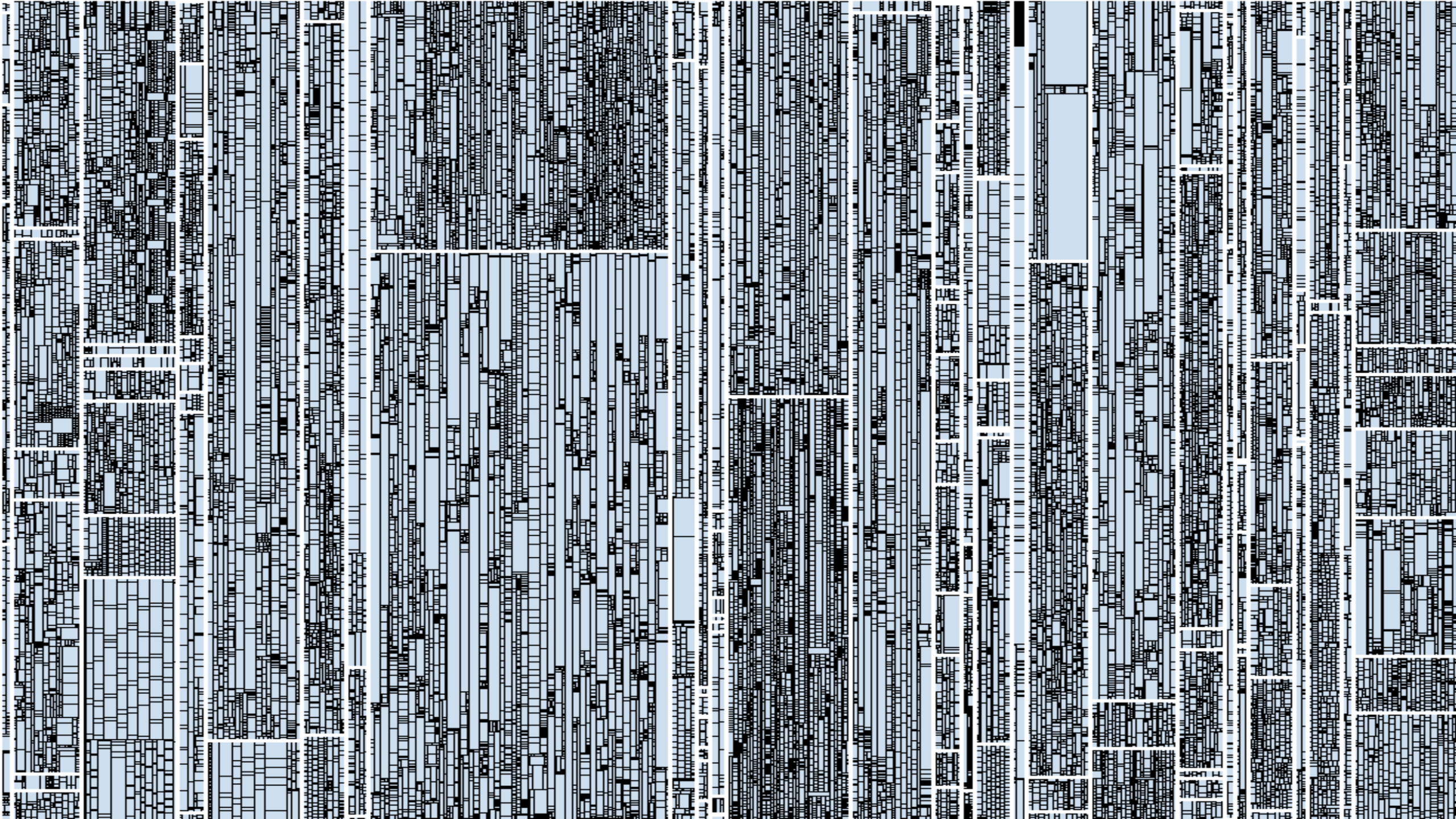




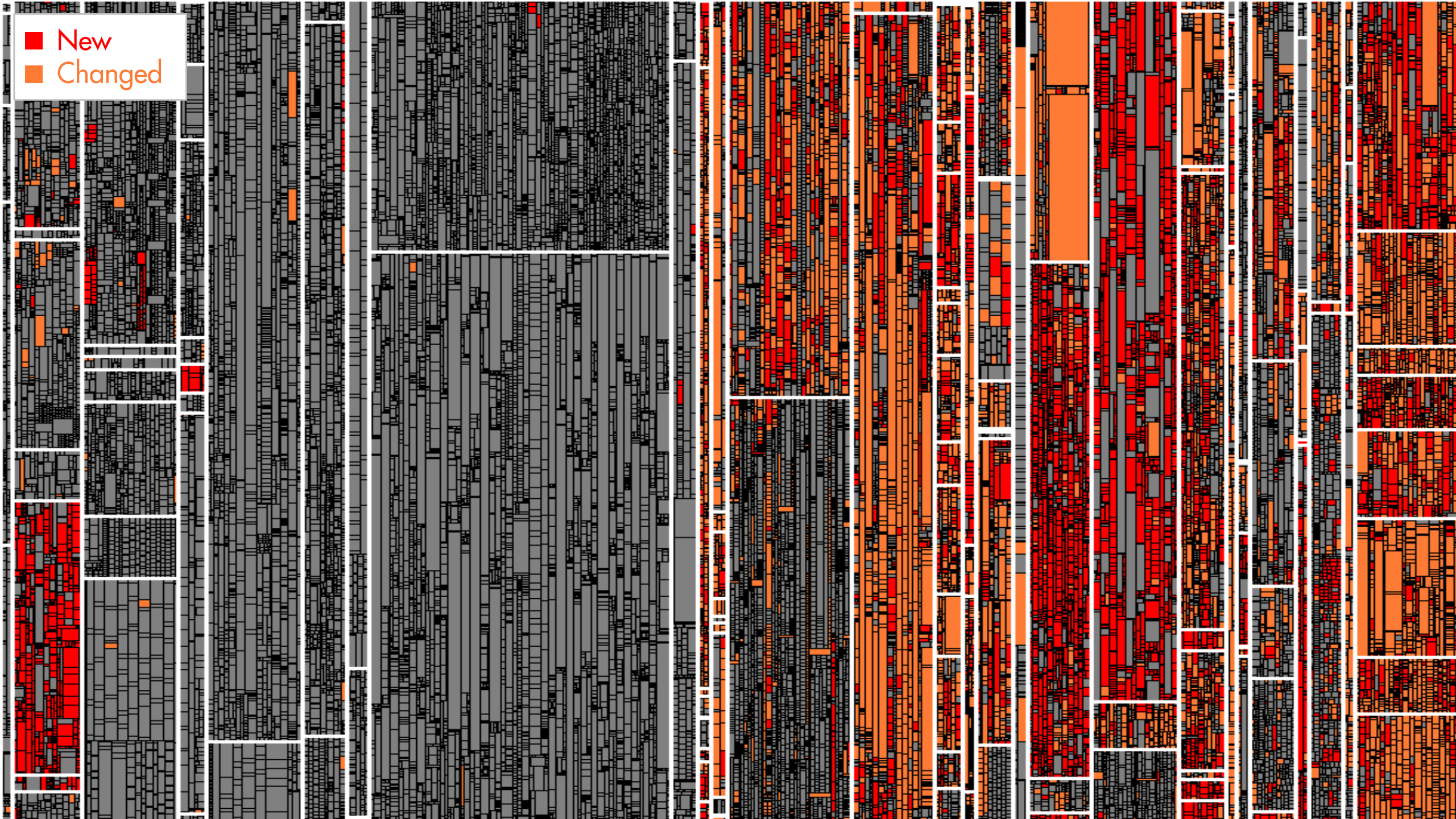


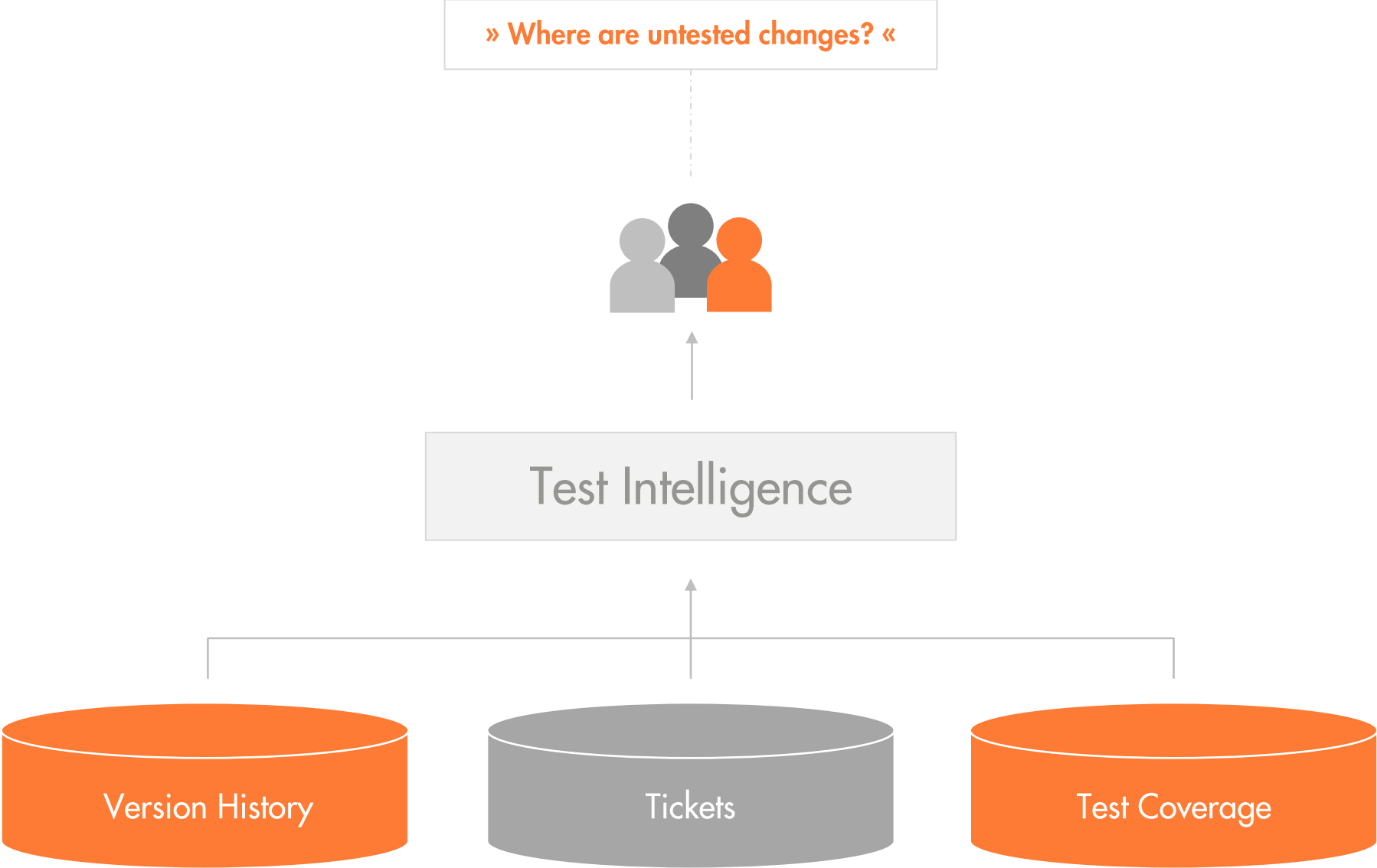






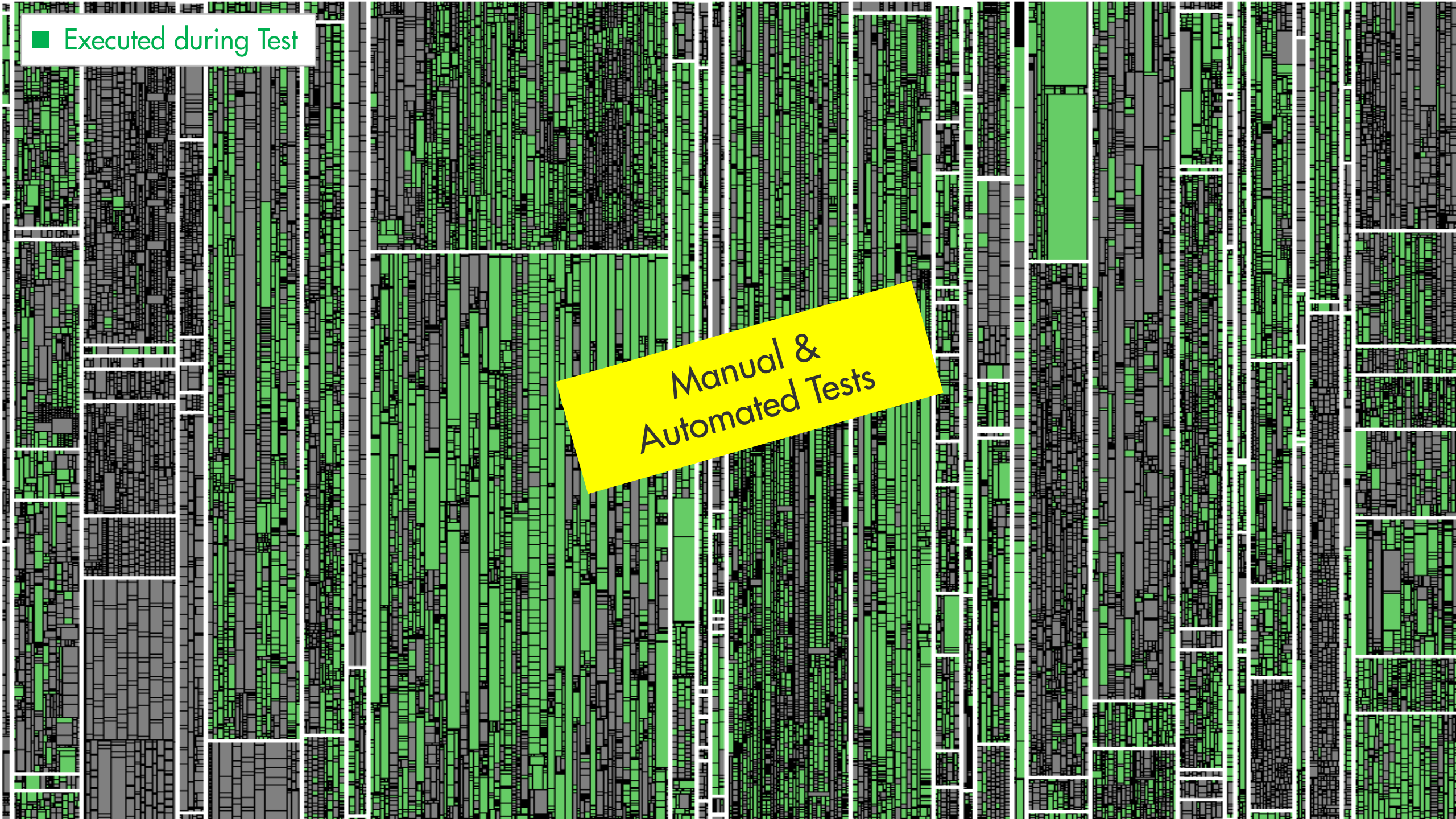
■ New
■ Changed



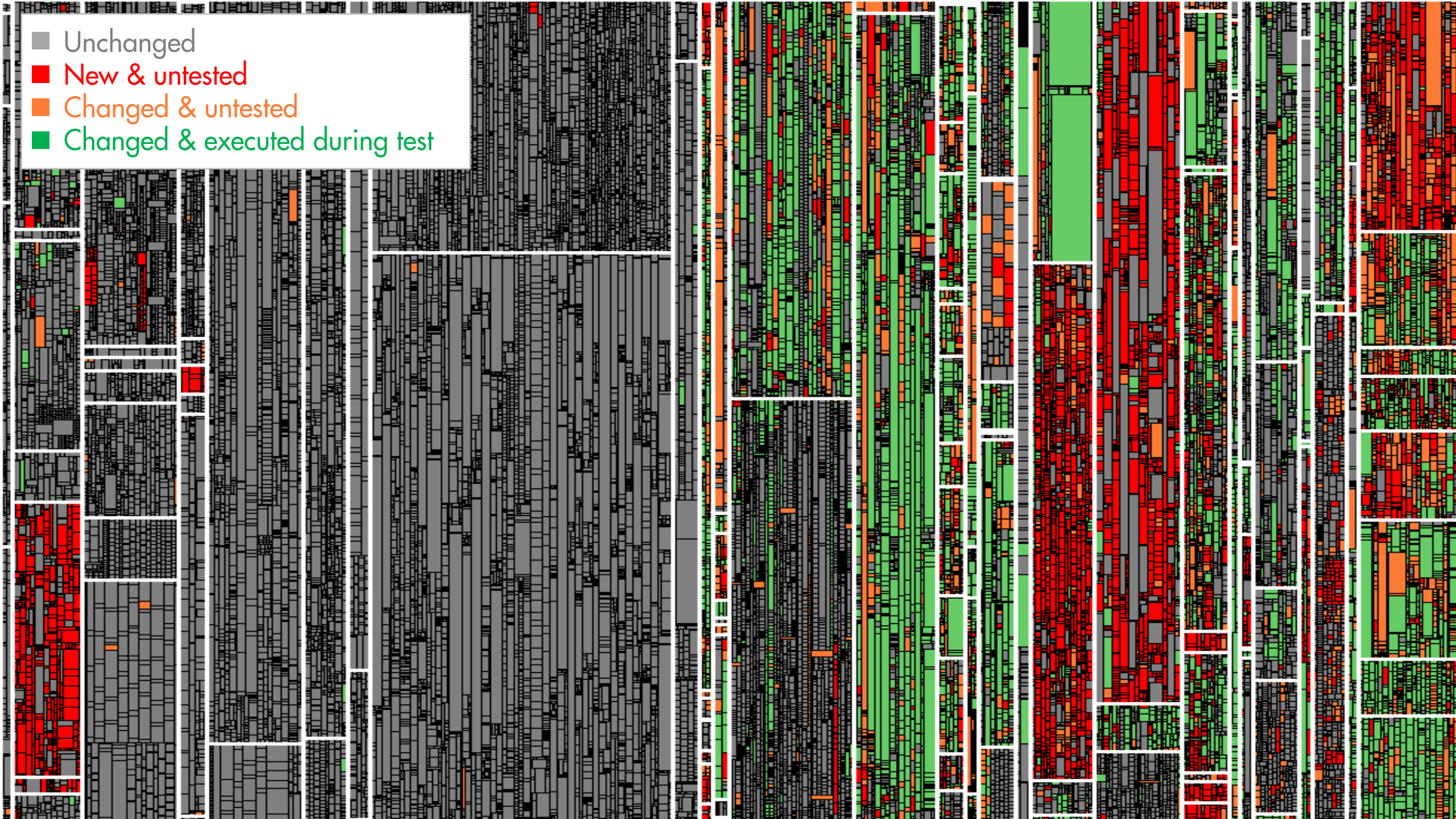


■ Executed during Test

Manual &
Automated Tests



- Unchanged
- New & untested
- Changed & untested
- Changed & executed during test



Changes



Test Gap Analysis

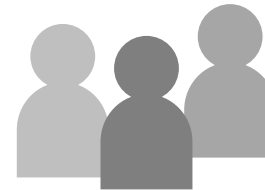
Execution



Untested Changes

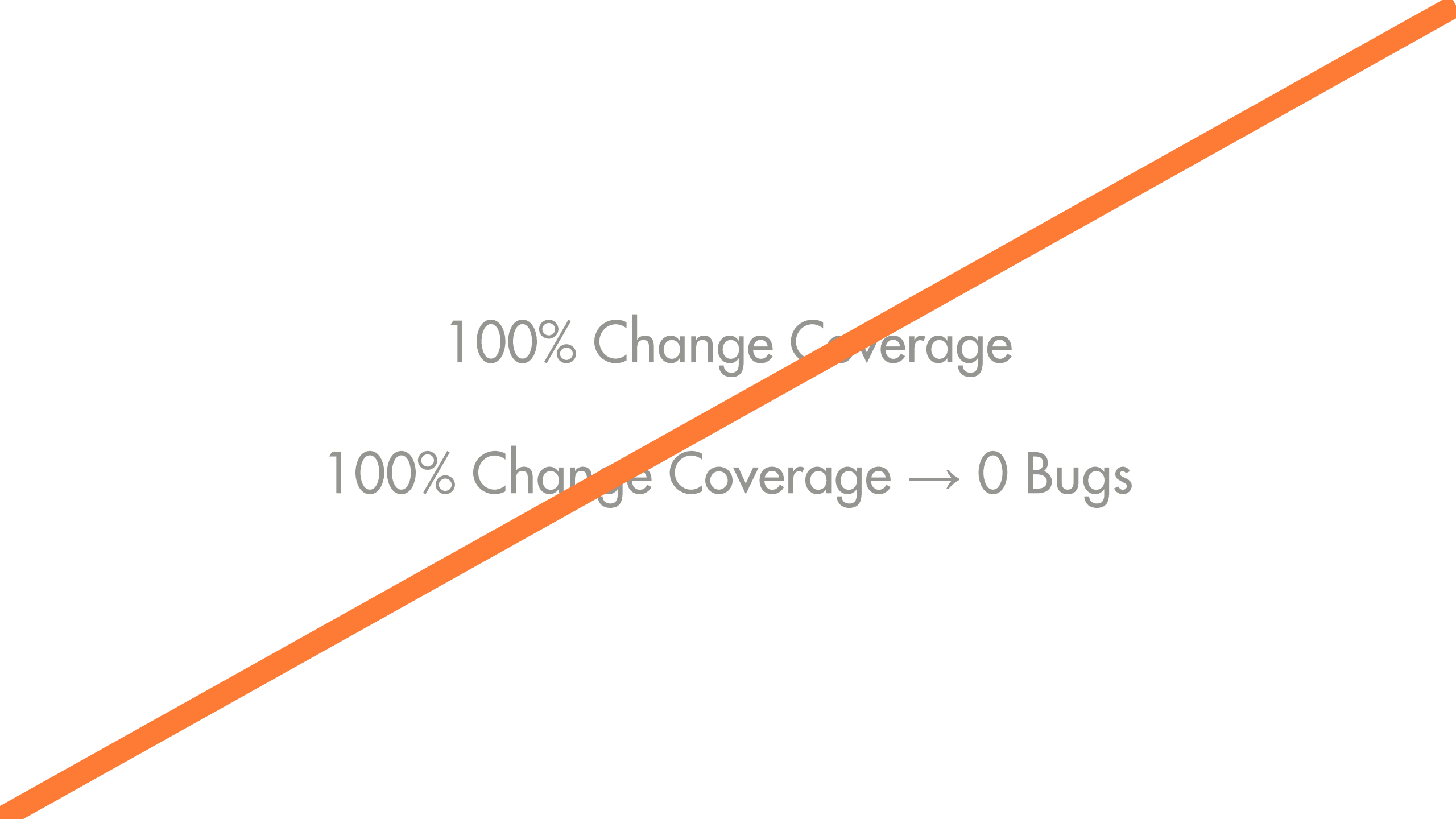


Test Cases



Developers, Testers, Test Managers

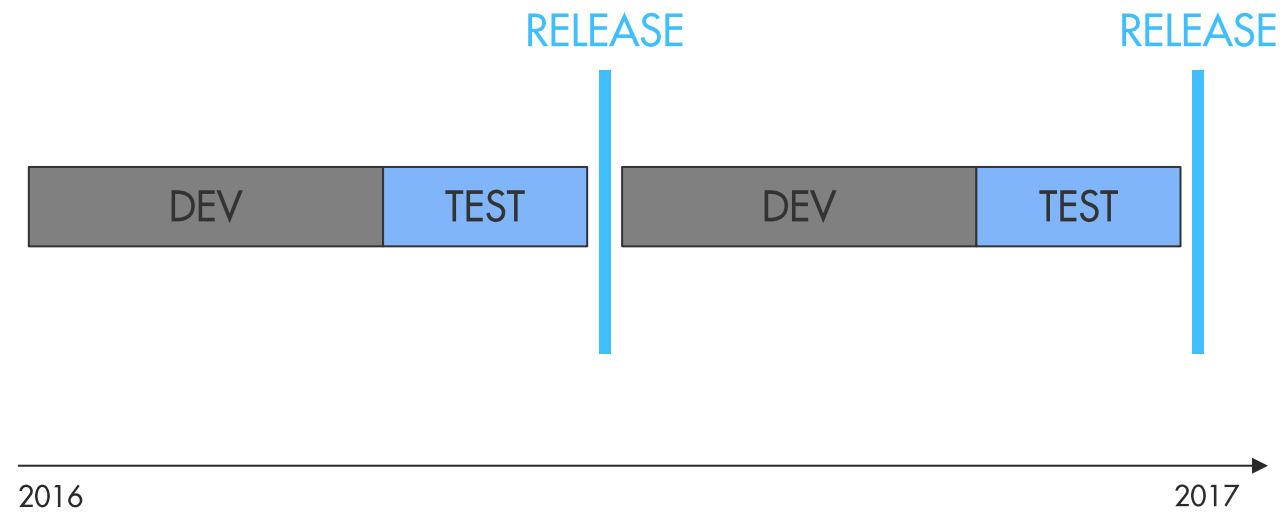




100% Change Coverage

100% Change Coverage → 0 Bugs

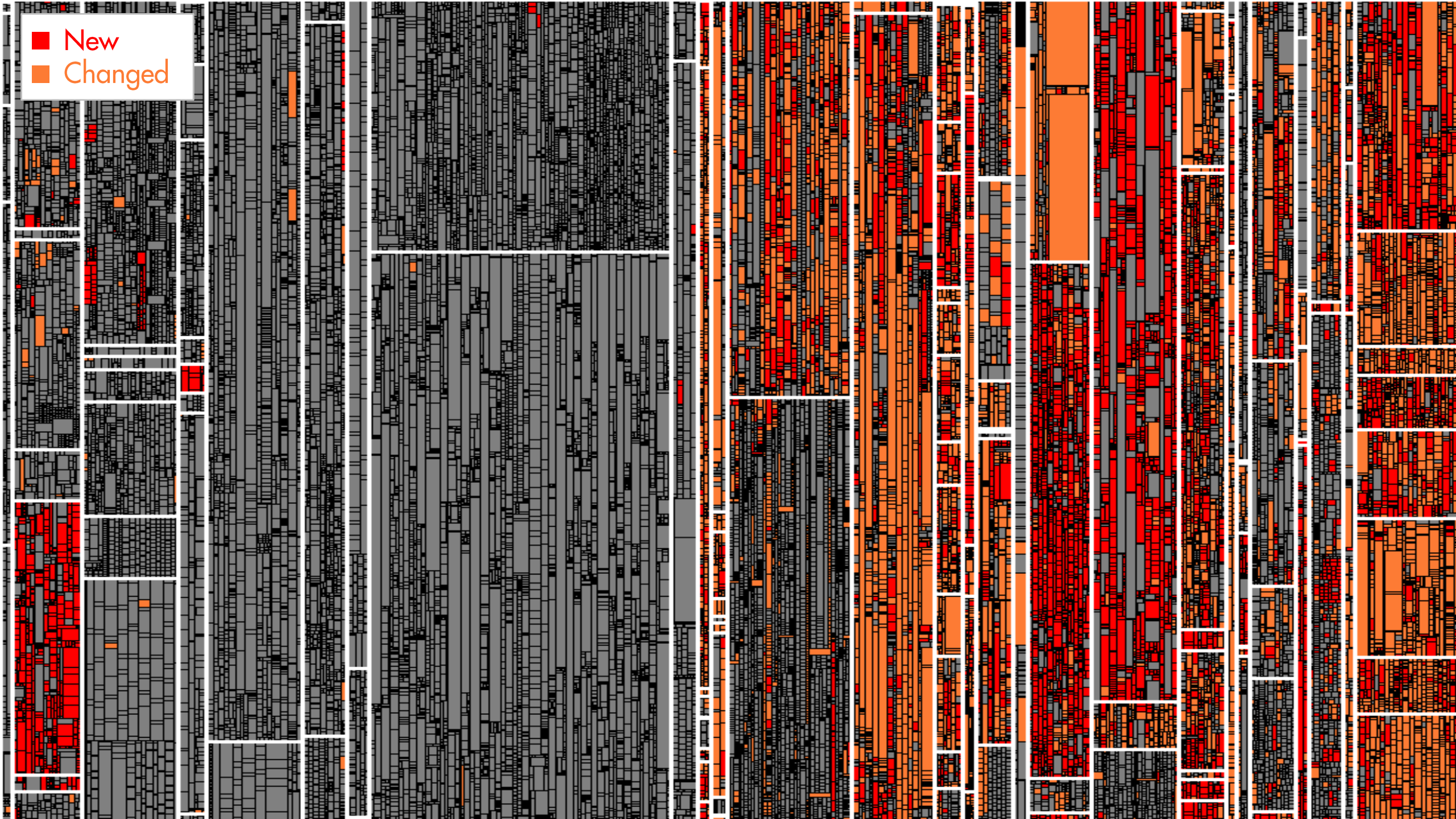
Release Test



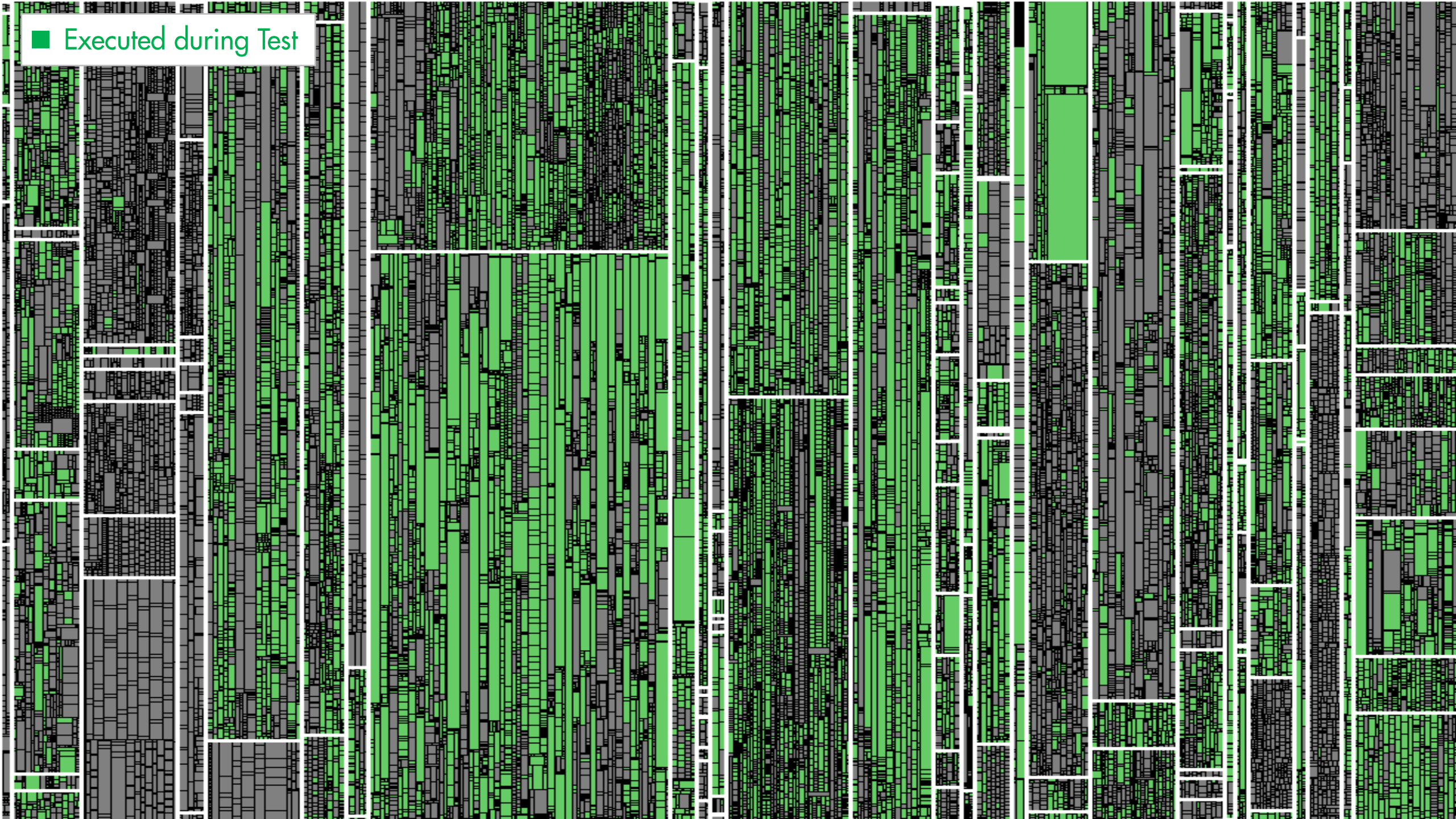
Dedicated Release Test

- Test Factory
- 5 testers in India
- Tests over 2 weeks
- 143 test sessions
- 6 versions of the system

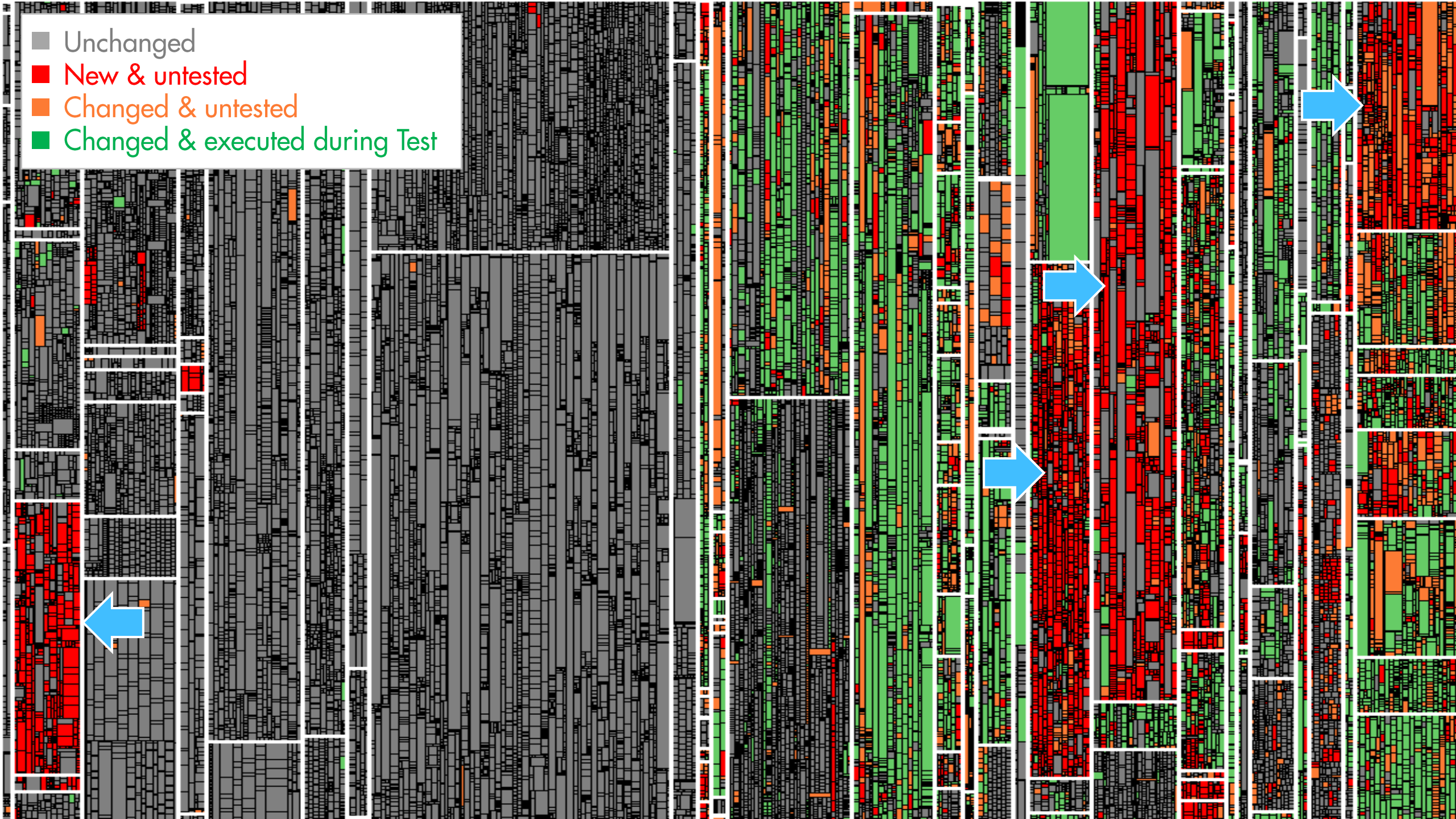
■ New
■ Changed



■ Executed during Test



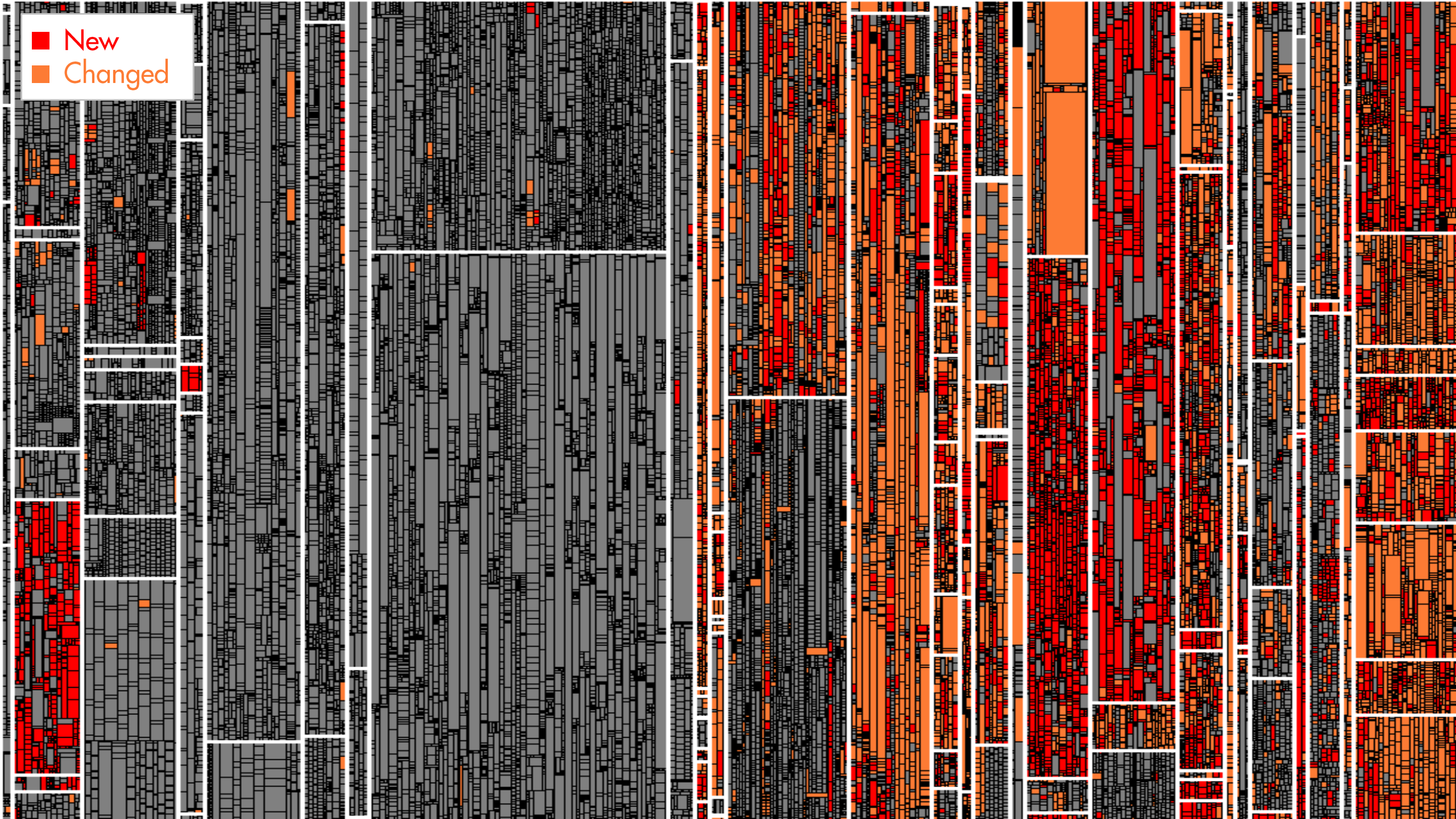
- Unchanged
- New & untested
- Changed & untested
- Changed & executed during Test



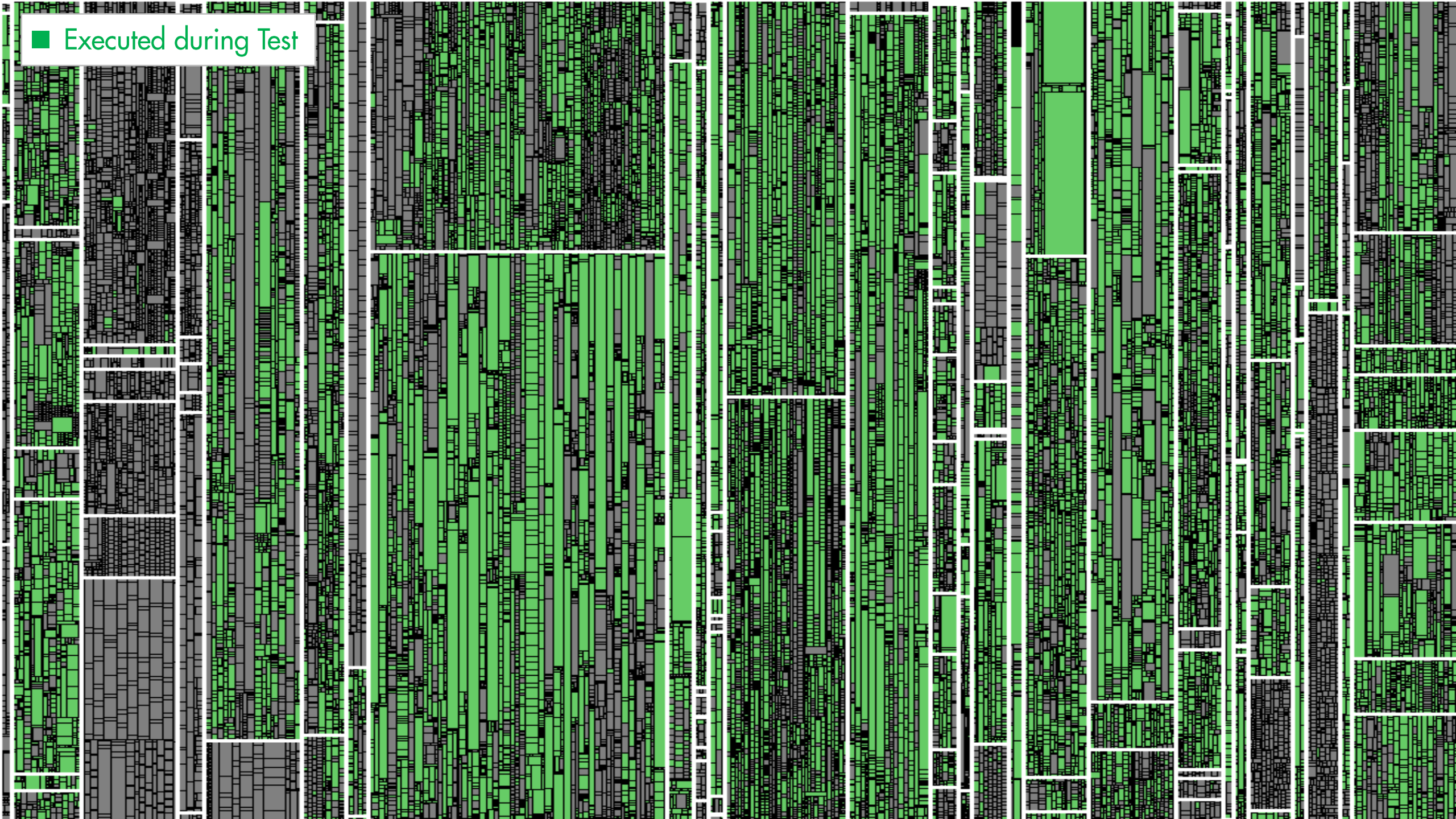
Explorative Test

- 23 testers
 - Requirements engineers
 - Business
 - End users
- Tests over 2 weeks
- 307 test sessions
- 8 versions of the system

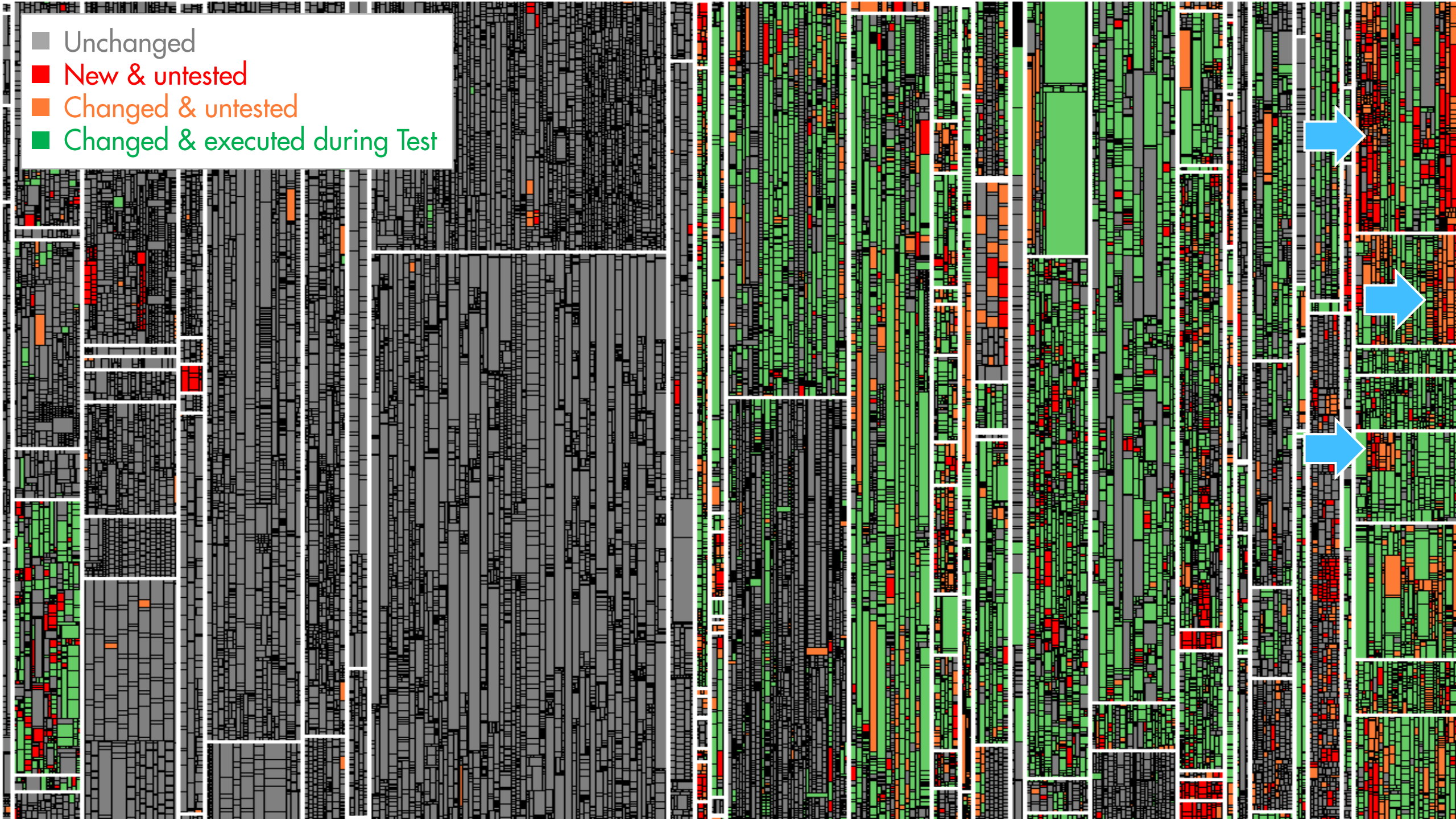
■ New
■ Changed



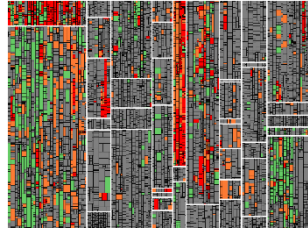
■ Executed during Test



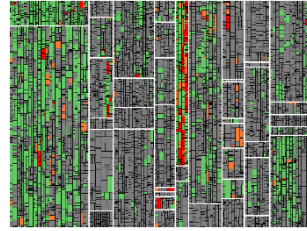
- Unchanged
- New & untested
- Changed & untested
- Changed & executed during Test



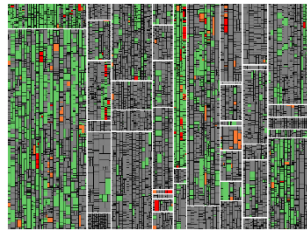
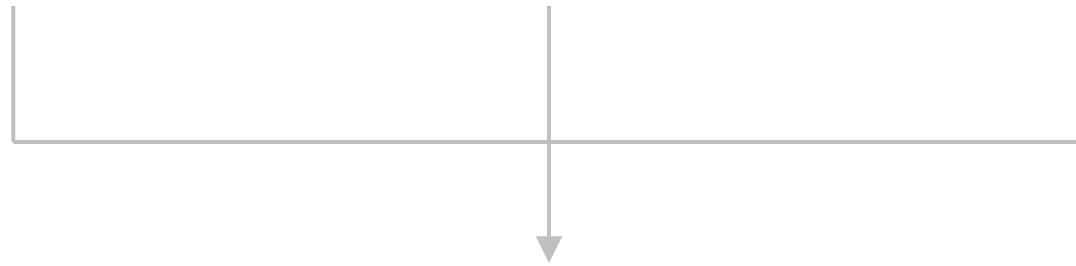
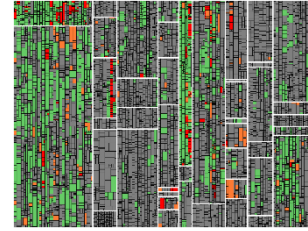
Component Test



Integration Test

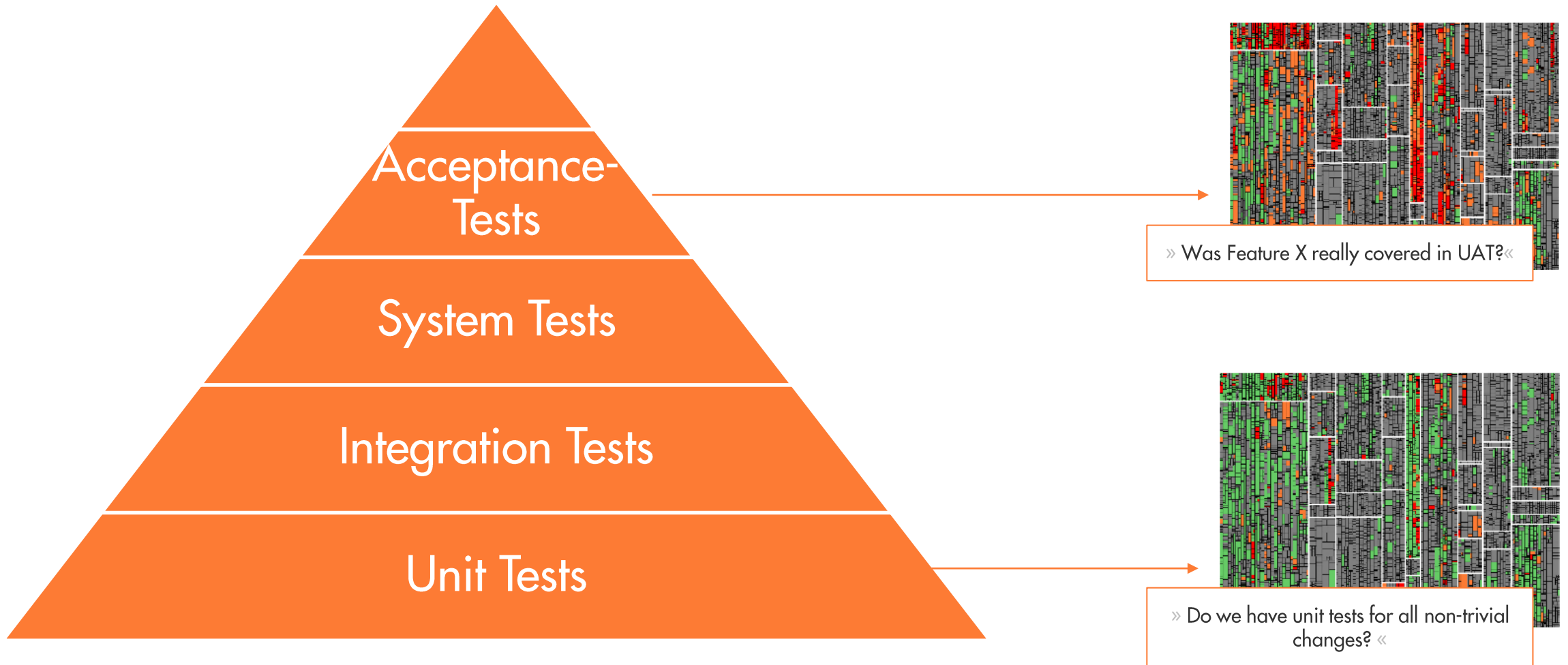


System Test



Aggregation

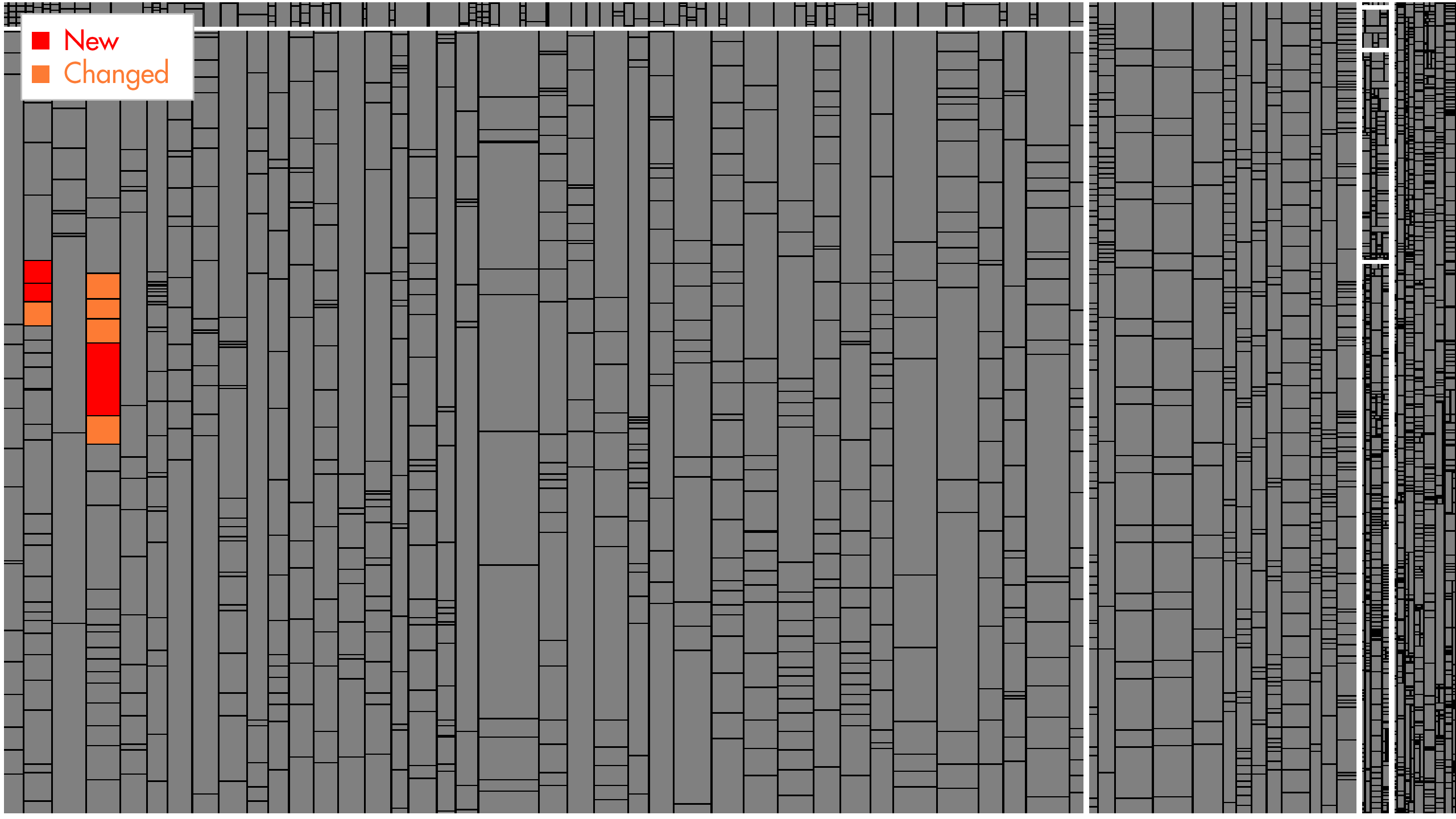
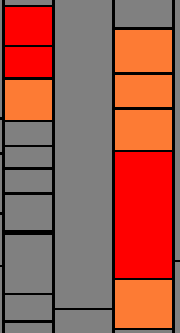
Central Repo for Test-Coverage



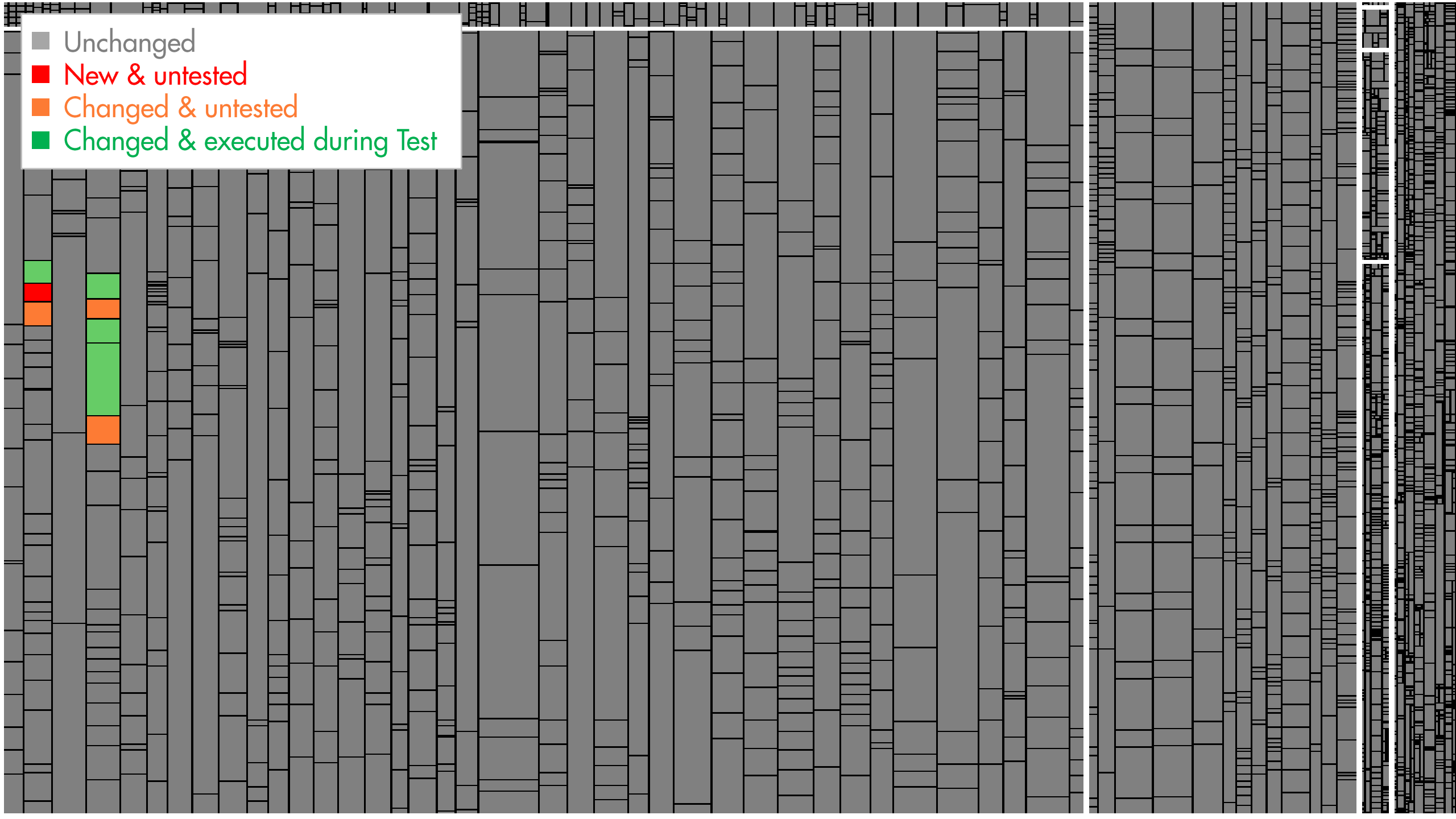
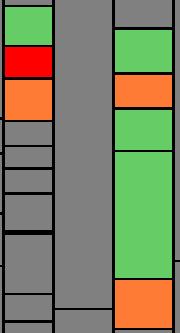
Hotfix Test

- On demand, little time
- Only the changes of the hotfix are being tested

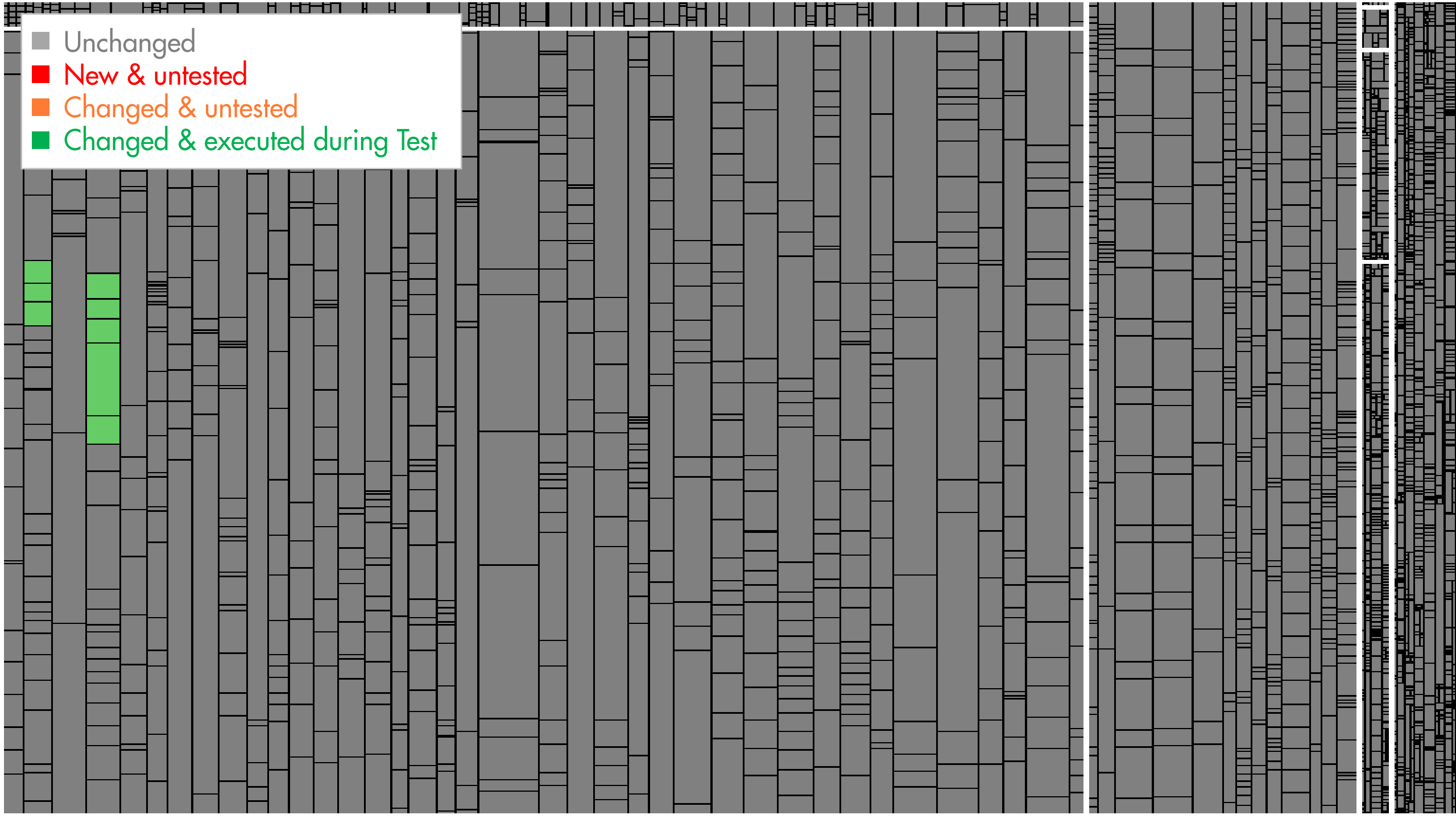
■ New
■ Changed



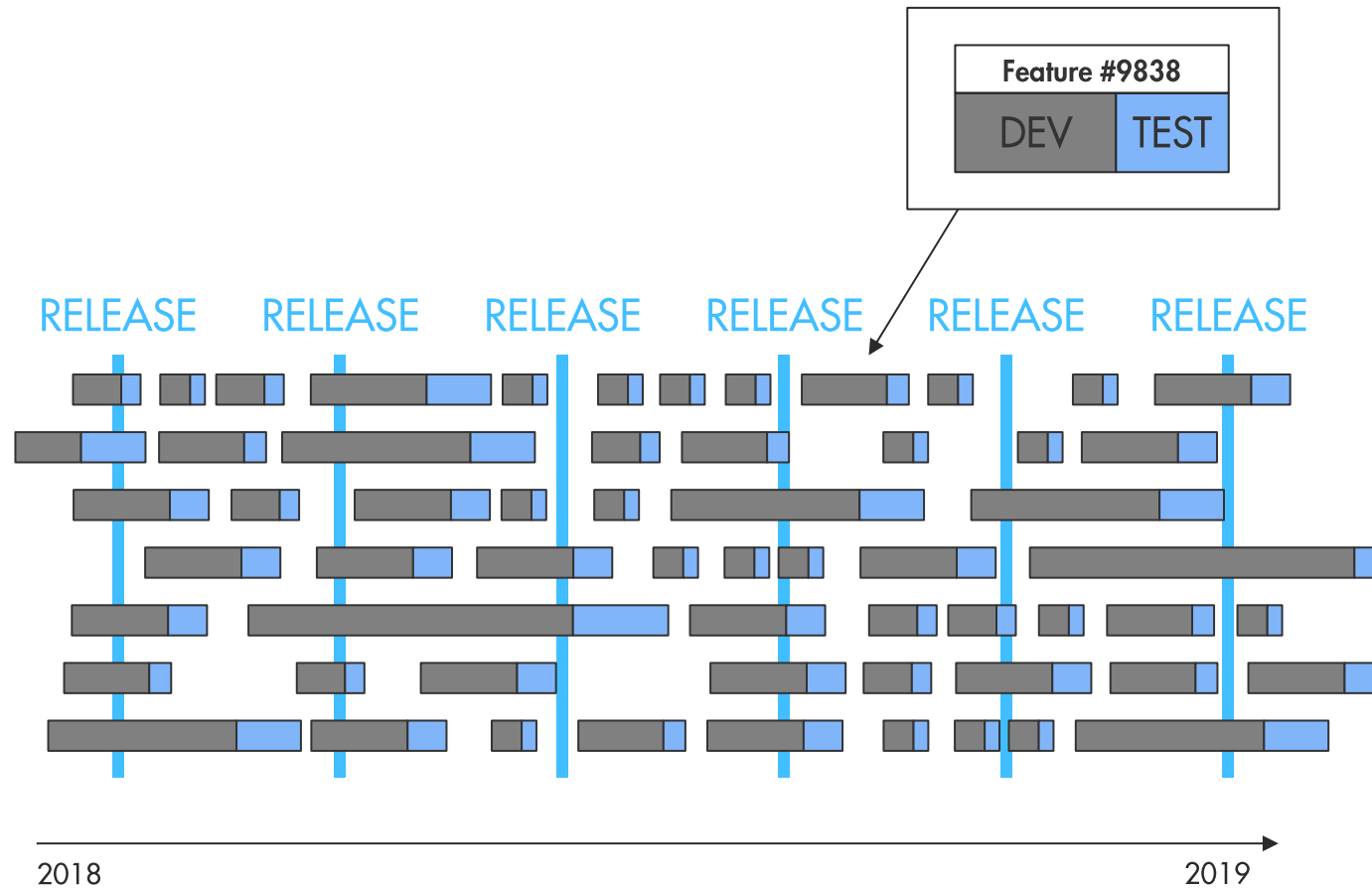
- Unchanged
- New & untested
- Changed & untested
- Changed & executed during Test

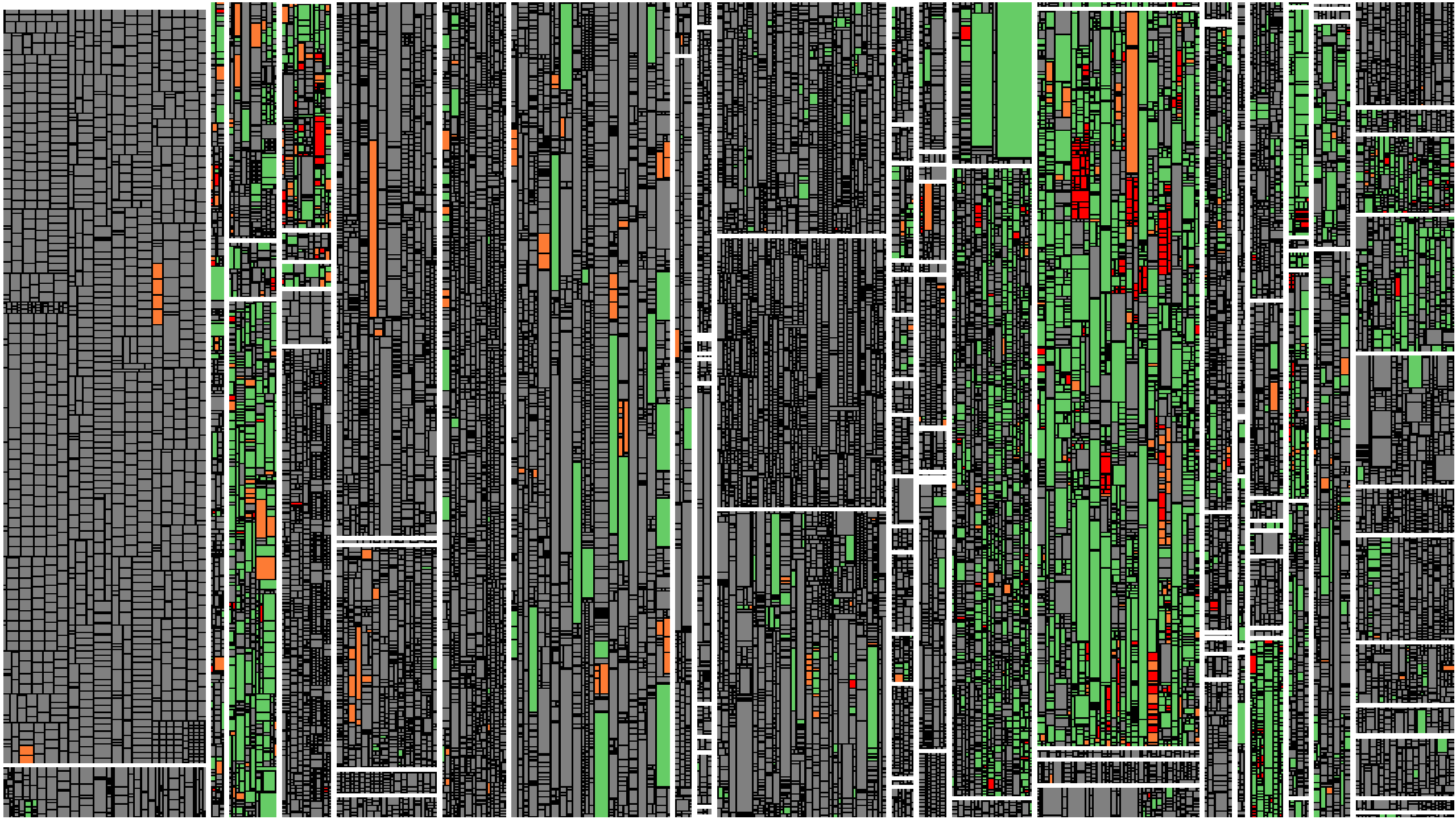


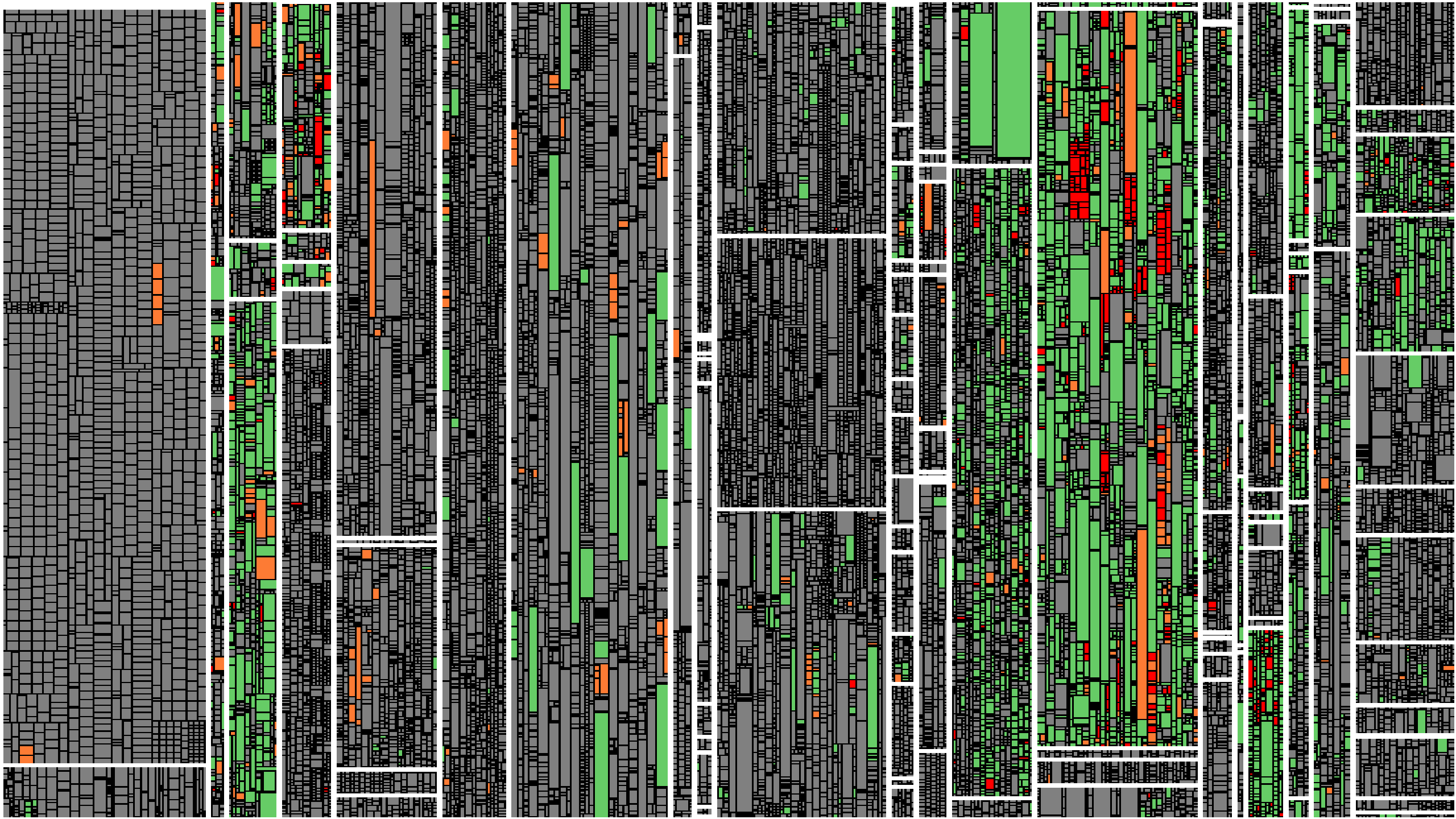
- Unchanged
- New & untested
- Changed & untested
- Changed & executed during Test

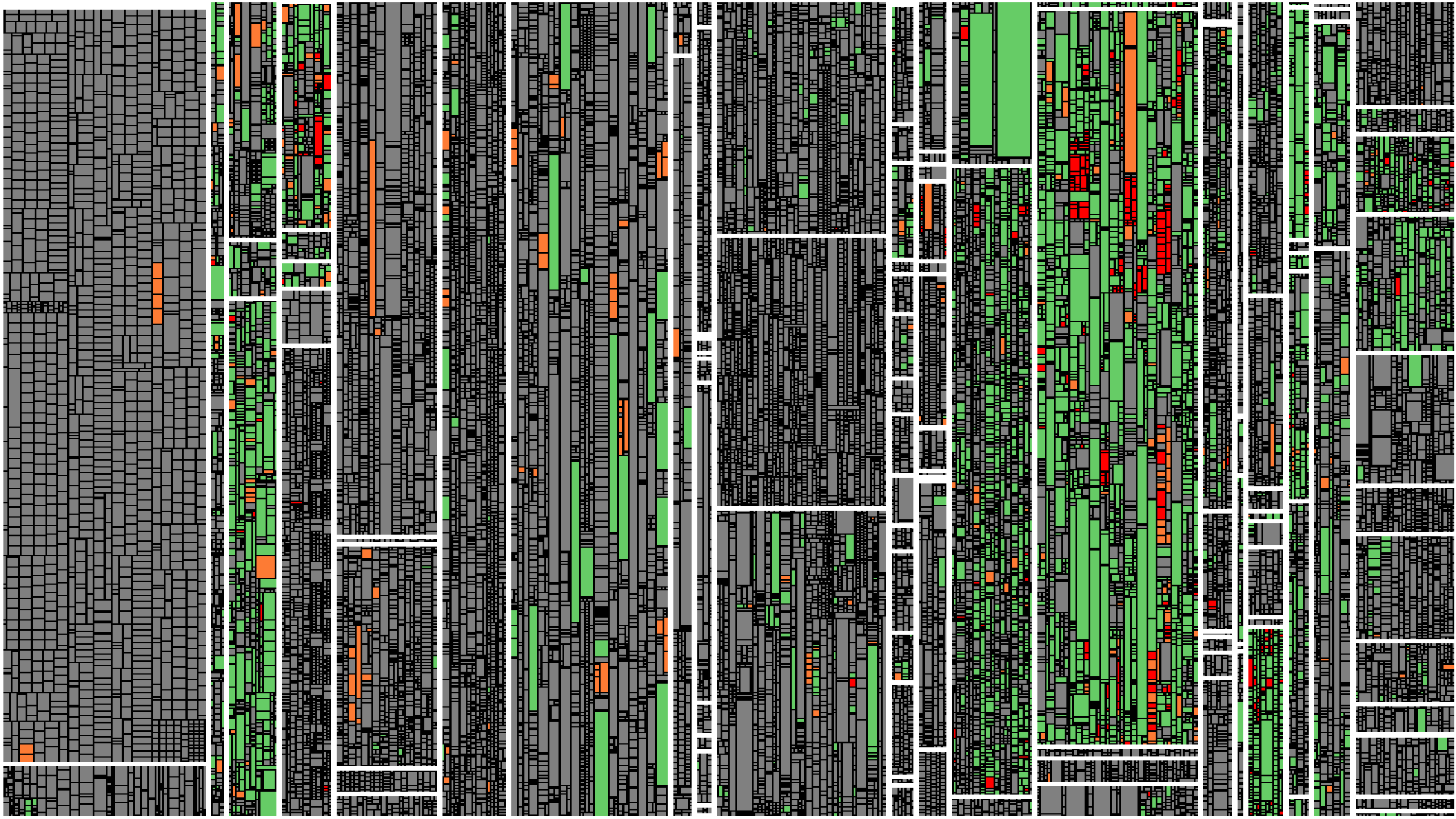


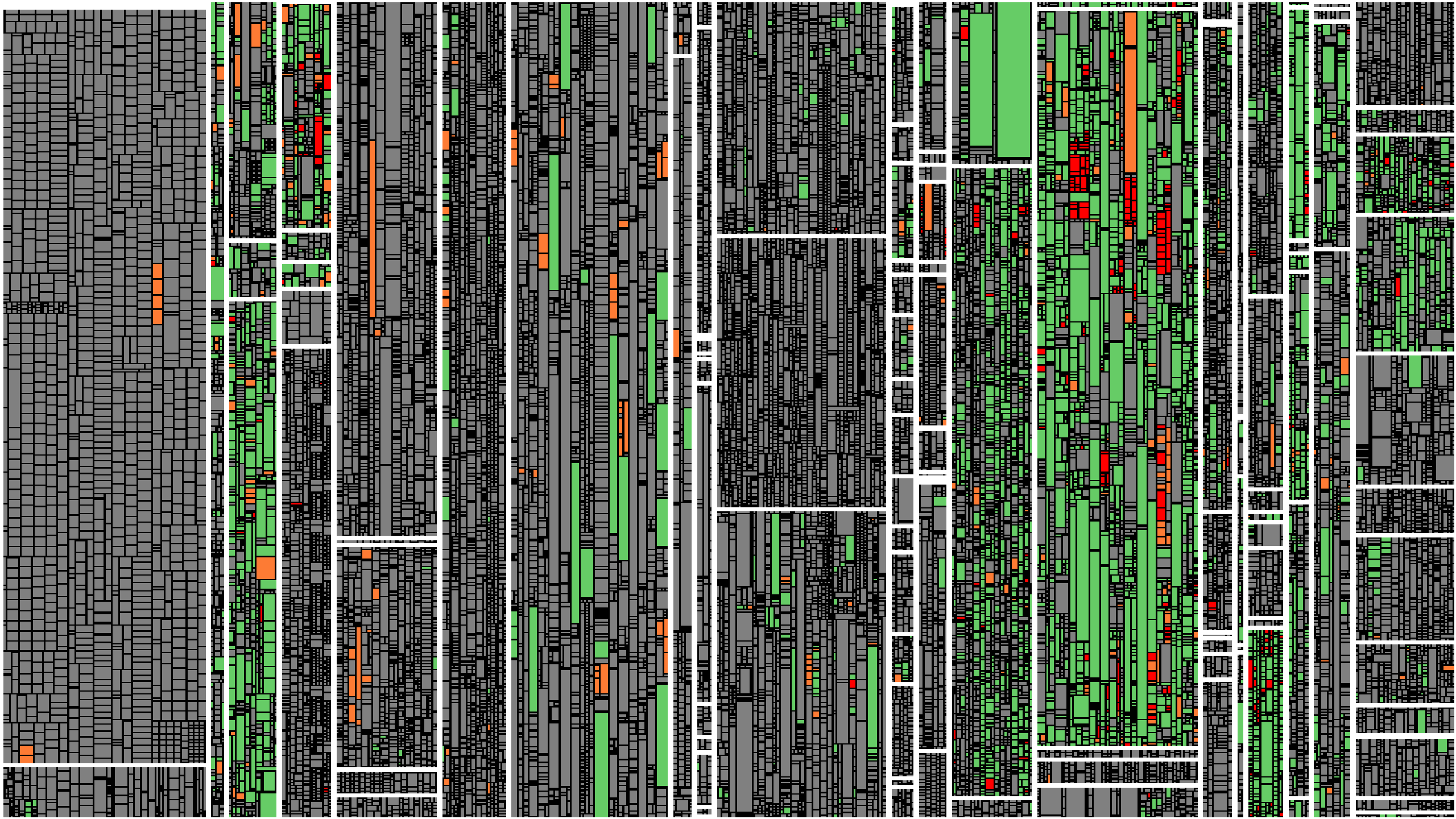
Continuous Testing

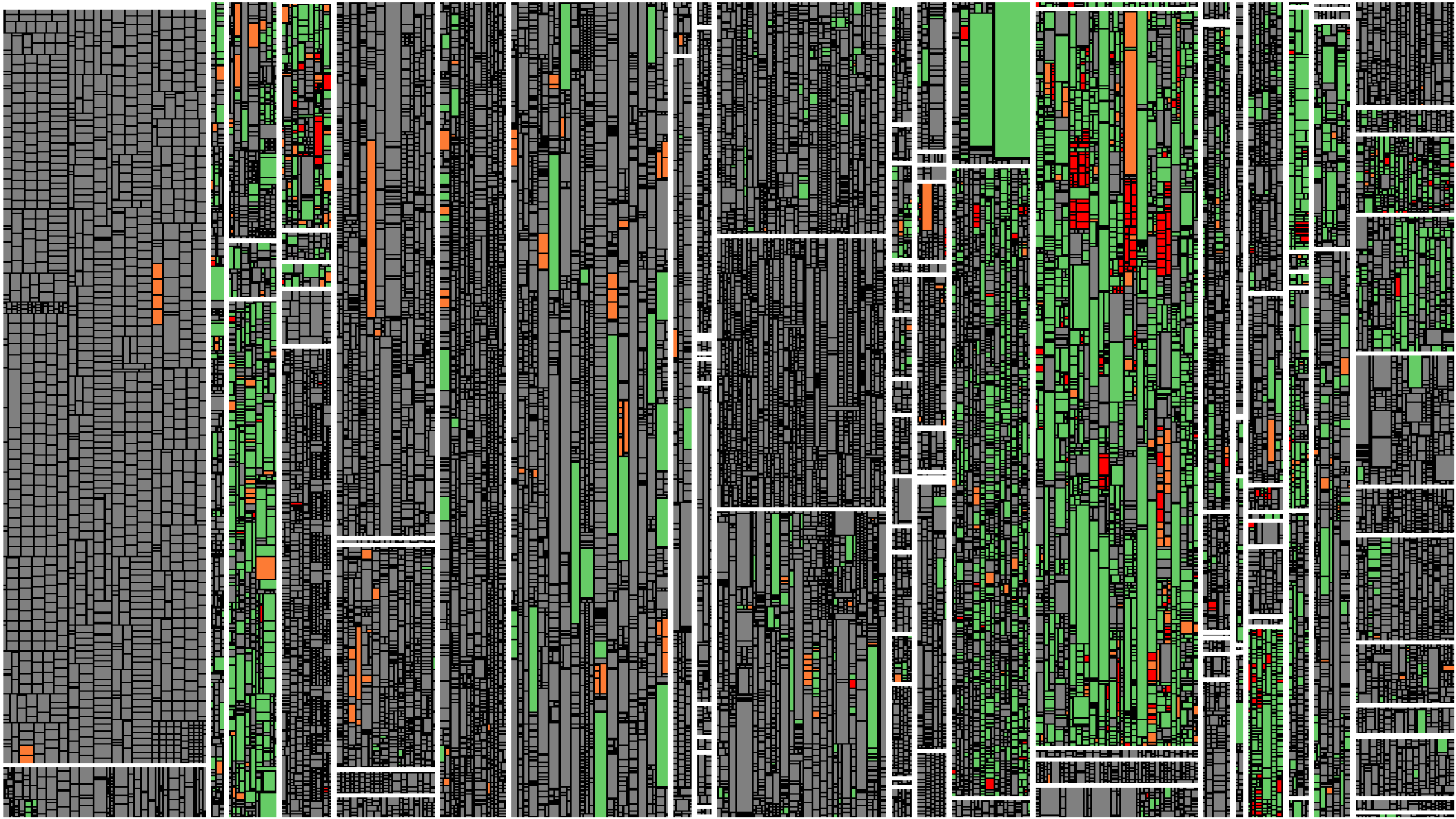


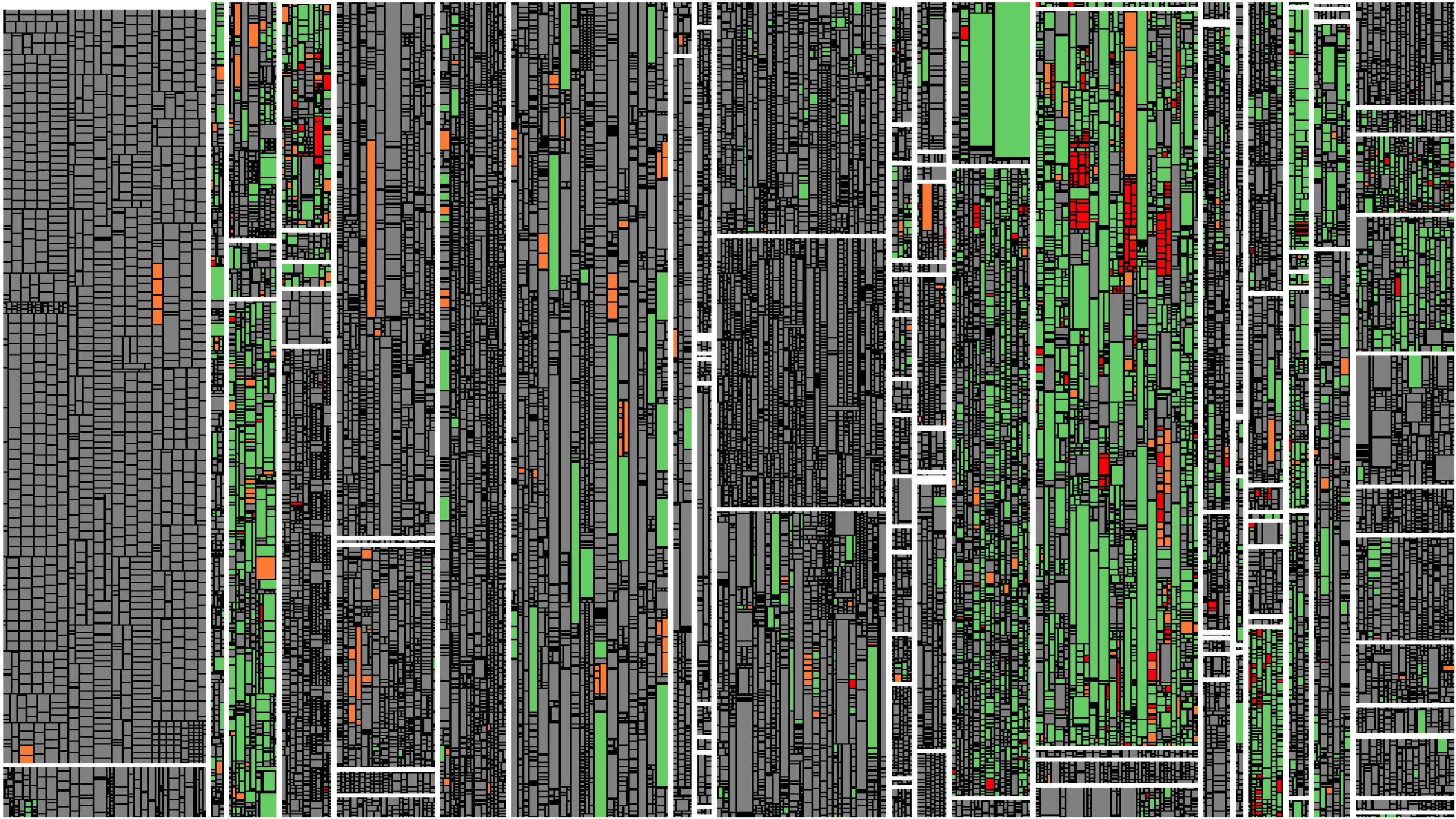


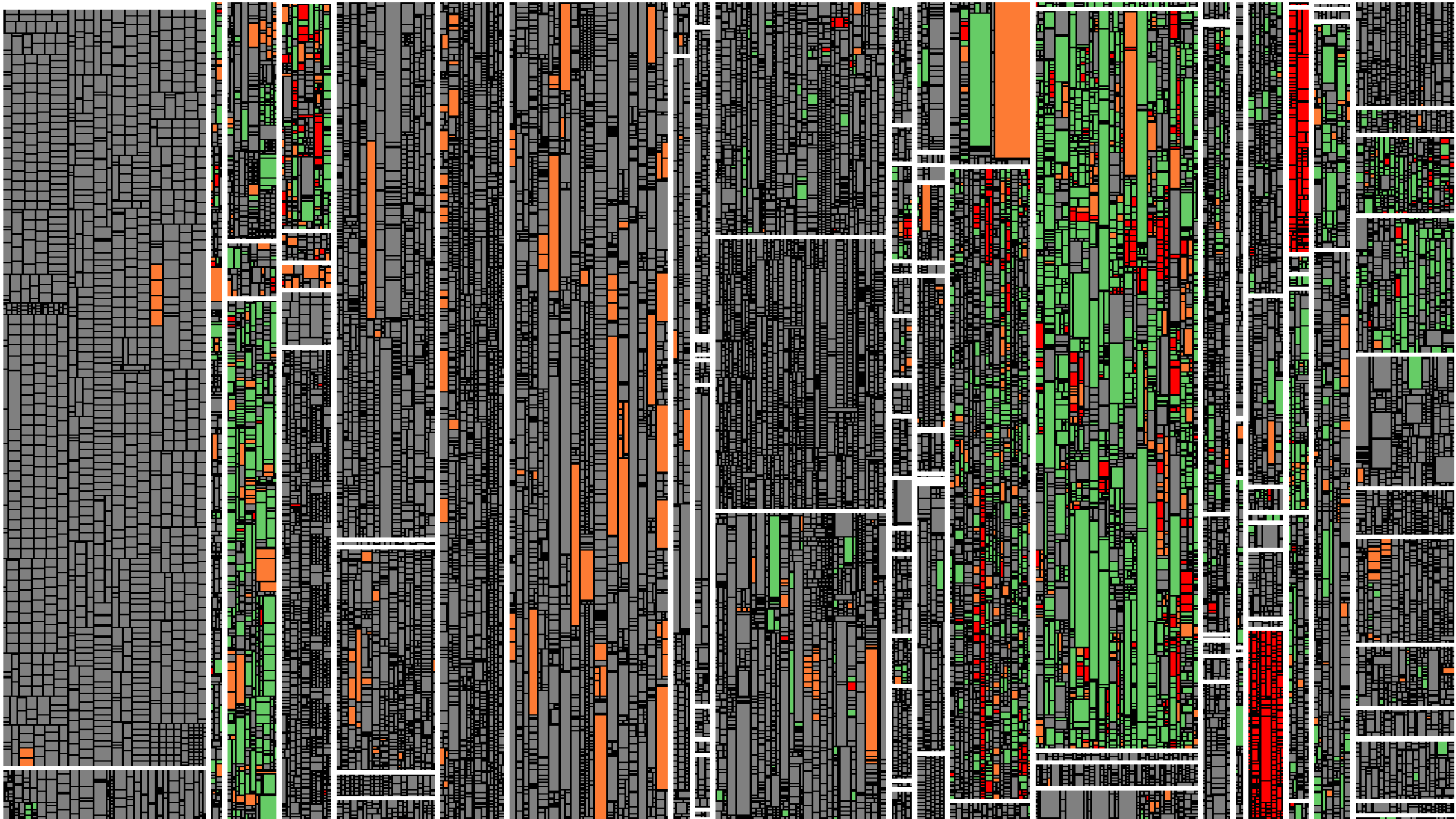


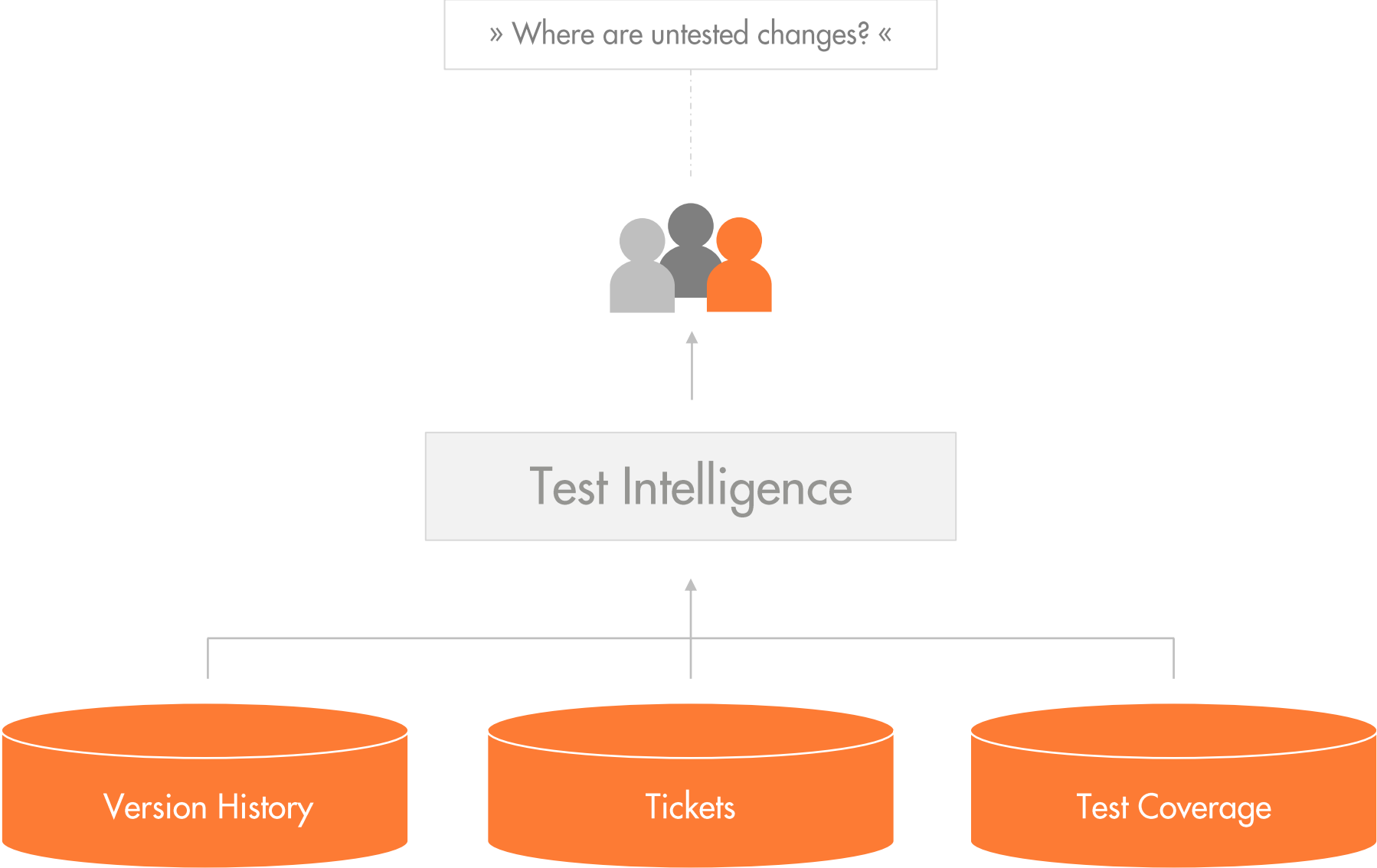


































Issue # ▼	Subject	Done		Test Gap
↗ TS-10549	Undo/Redo for web-based architecture editor	Done		0% 
↗ TS-10784	Fix long method finding in TaintAnalysisRunner	Done		0% 
↗ TS-10923	Implement metric 'Nesting Depth' for Simulink	Done		29% 
↗ TS-11364	External findings are not registered during first upload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

Issue # ▾	Subject		Test Gap
TS-14421	Get rid of TestGapSynchronizer block	Done 	0% 
TS-14733	Remove Dataflow blocks	Done 	22% 

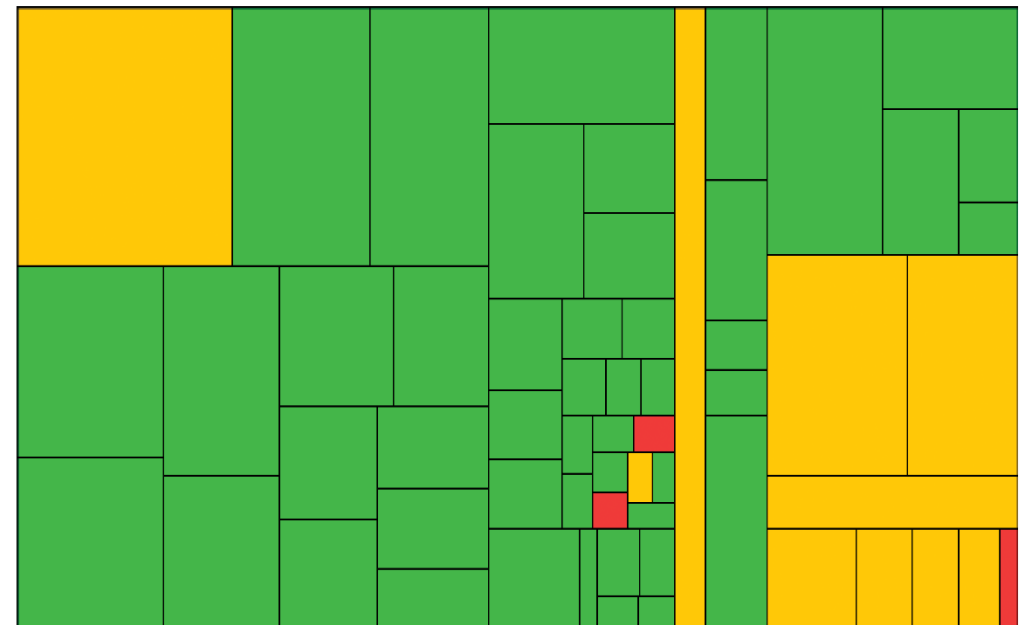
Done Issue TS-14733 - Remove Dataflow blocks

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		
		https://git.cqse.eu/cqse/teamscale/3621		

Aug 15 2018 12:37–Now | Test Gap: 22%



**How can we quantify
the benefits?**

How many changes are untested?

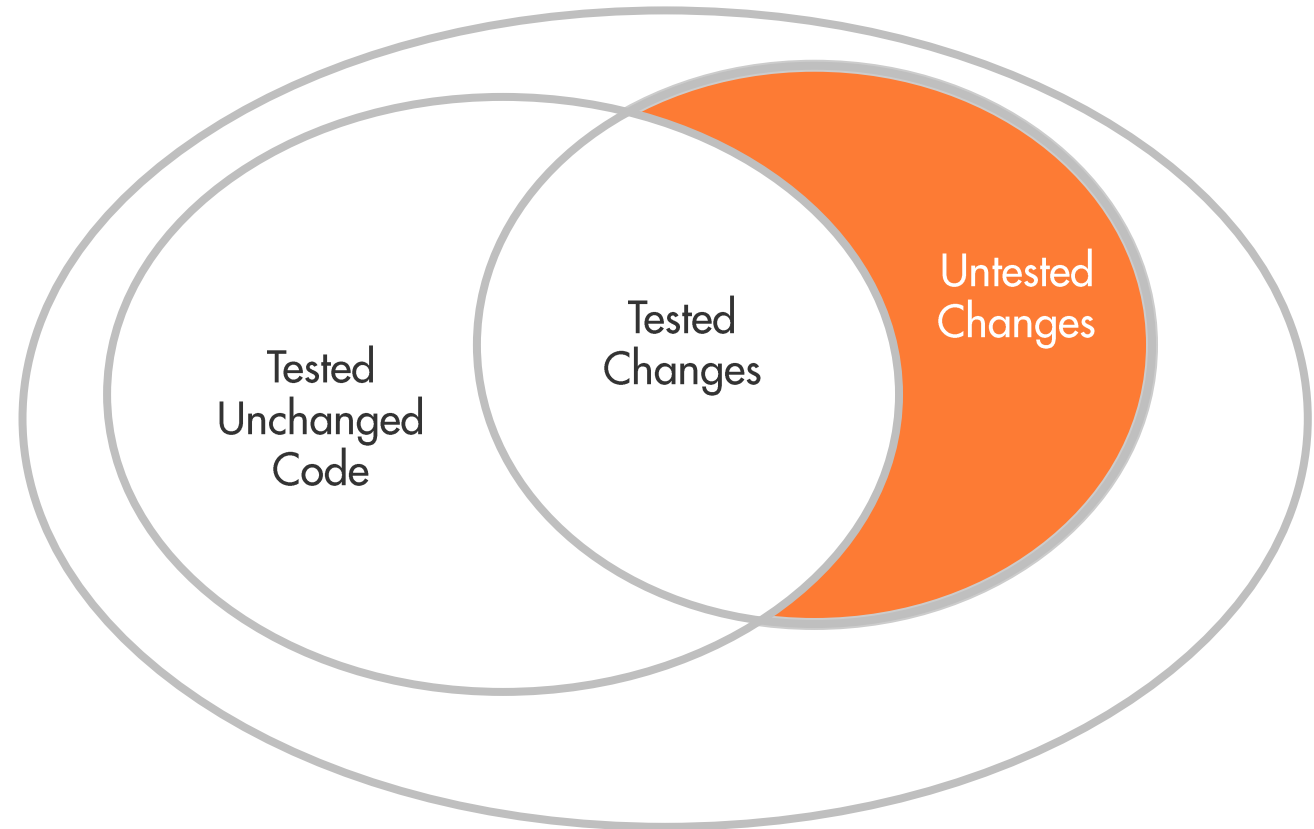
Study: C# System @MunichRe

Release A:

15% Code new/changed,
>50% untested

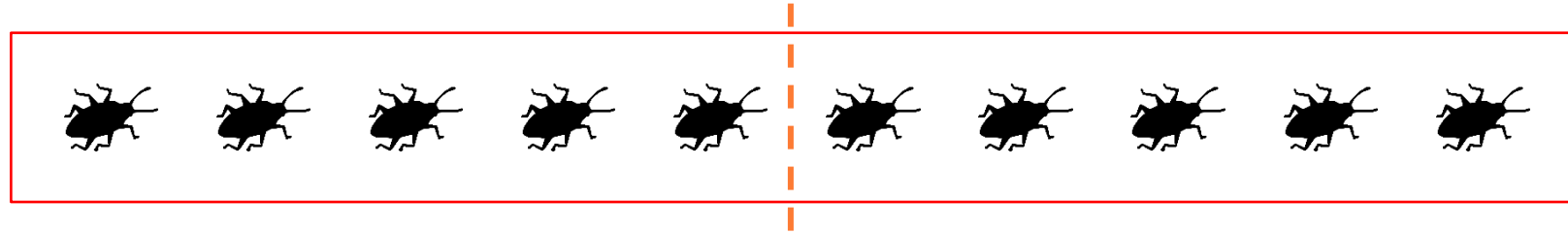
Release B:

15% Code new/changed,
>60% untested

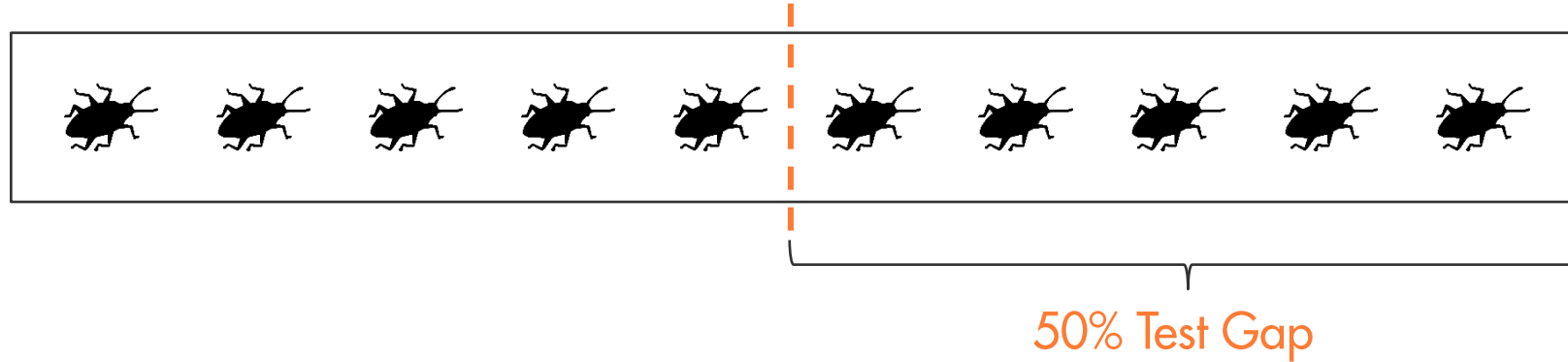


Probability of production bugs is 5x higher for untested changes!

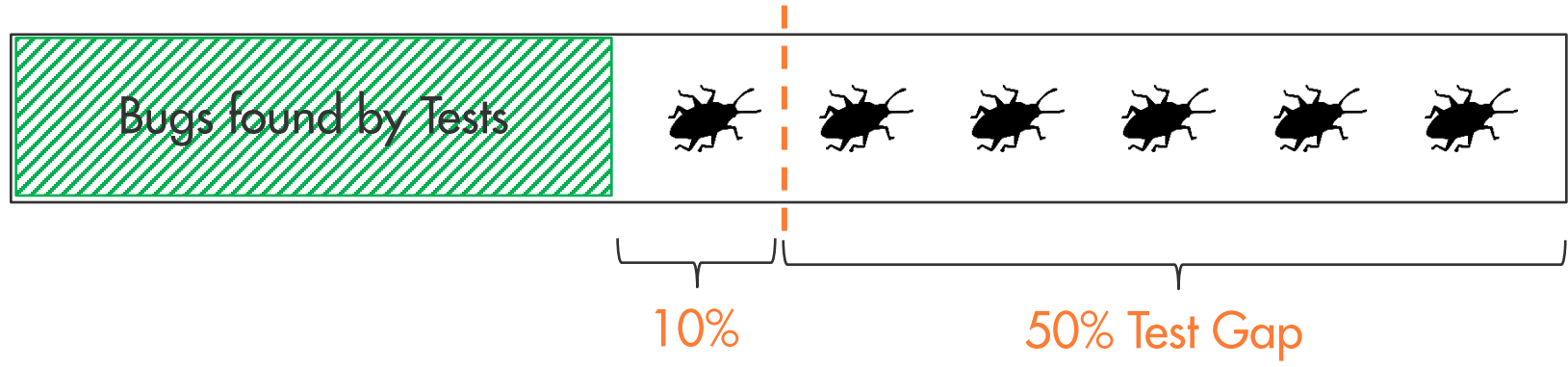
$$\% \text{Remaining Bugs} = \% \text{Tested} * \text{Test Ineffectiveness} + \% \text{Test Gap}$$



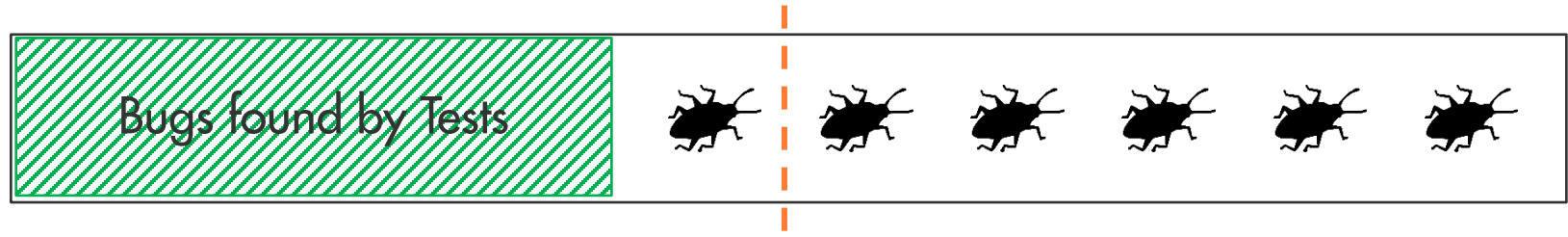
$$\% \text{Remaining Bugs} = \% \text{Tested} * \text{Test Ineffectiveness} + \% \text{Test Gap}$$



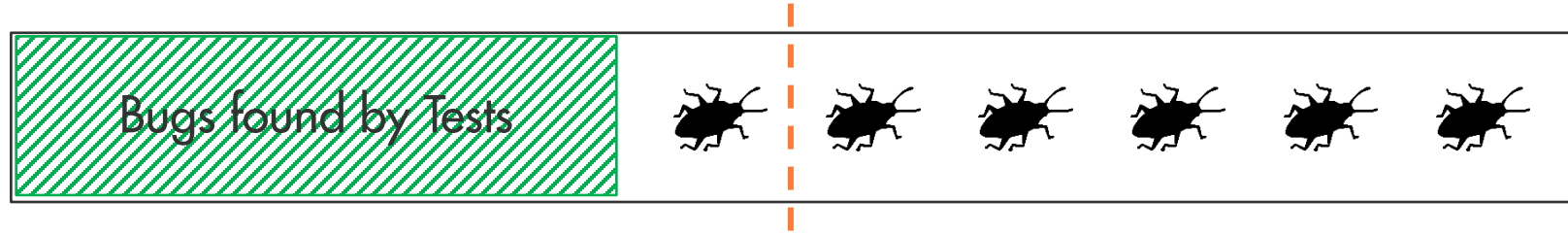
$$\% \text{Remaining Bugs} = 10\% + 50\%$$



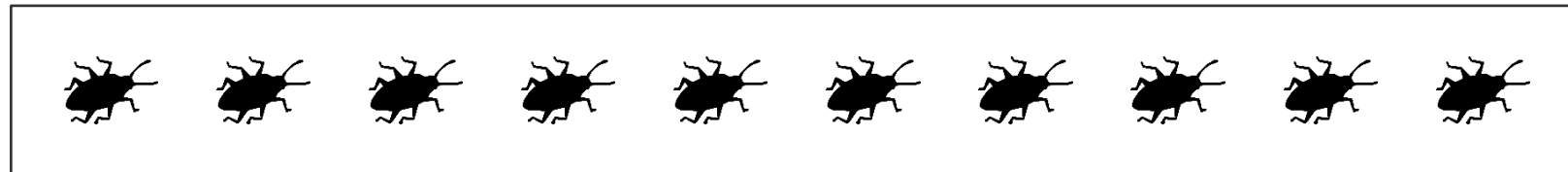
%Remaining Bugs = 60%



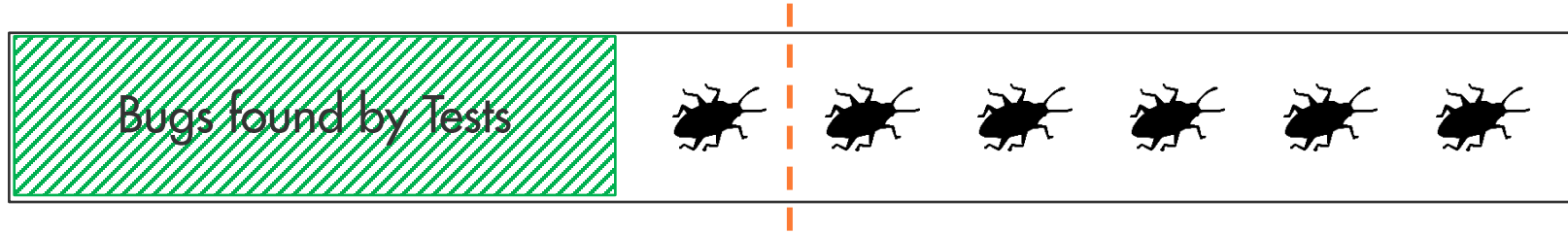
%Remaining Bugs = 60%



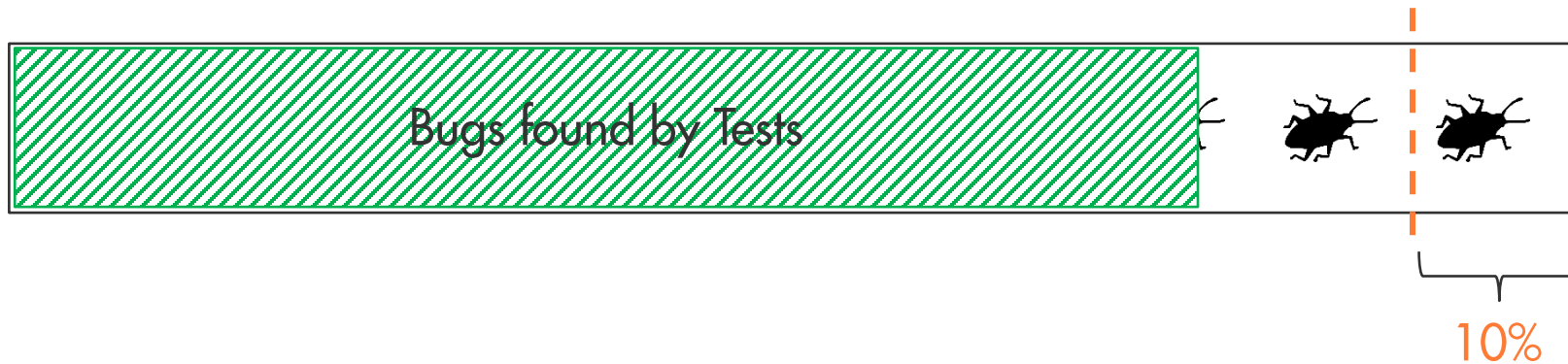
%Remaining Bugs = %Tested * Test Ineffectiveness + %Test Gap



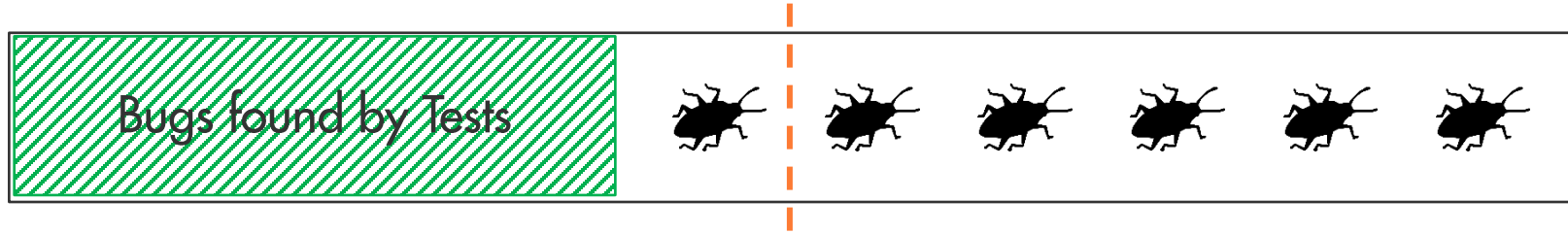
$$\% \text{Remaining Bugs} = 60\%$$



$$\% \text{Remaining Bugs} = 90\% * 20\% + 10\%$$



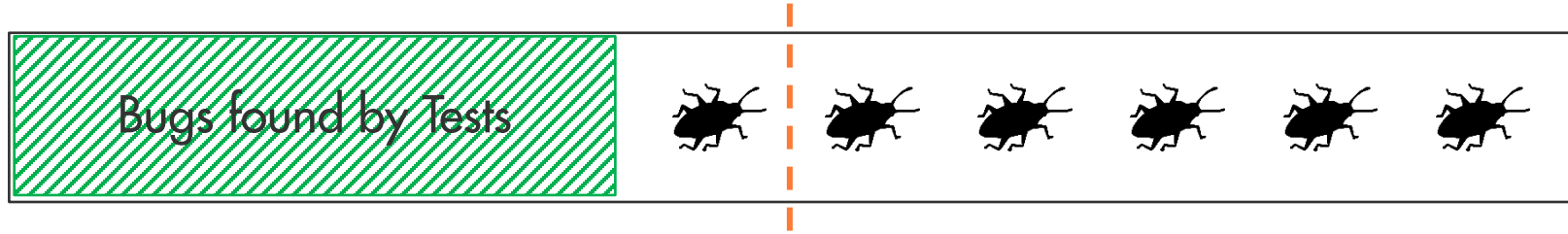
$\% \text{Remaining Bugs} = 60\%$



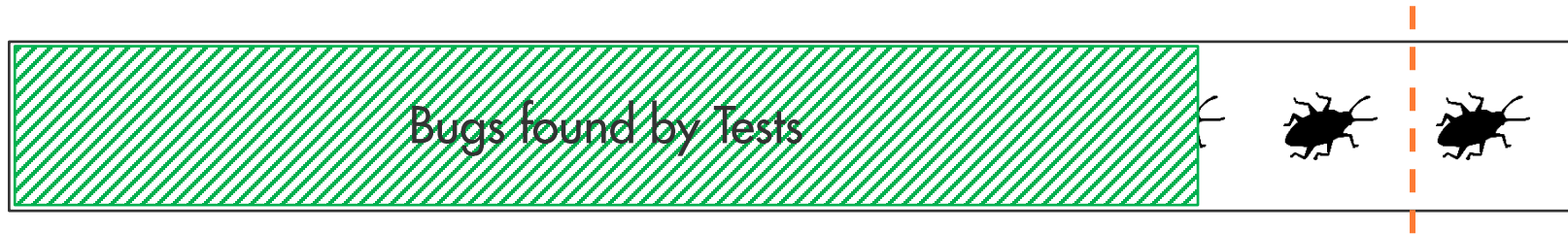
$\% \text{Remaining Bugs} = 18\% + 10\%$



%Remaining Bugs = 60%



%Remaining Bugs = 28%



Reduced Production Defects = **50%**

Test Gap analysis reduced production defects in applications of MunichRe by 1/2

» Where are untested changes? «

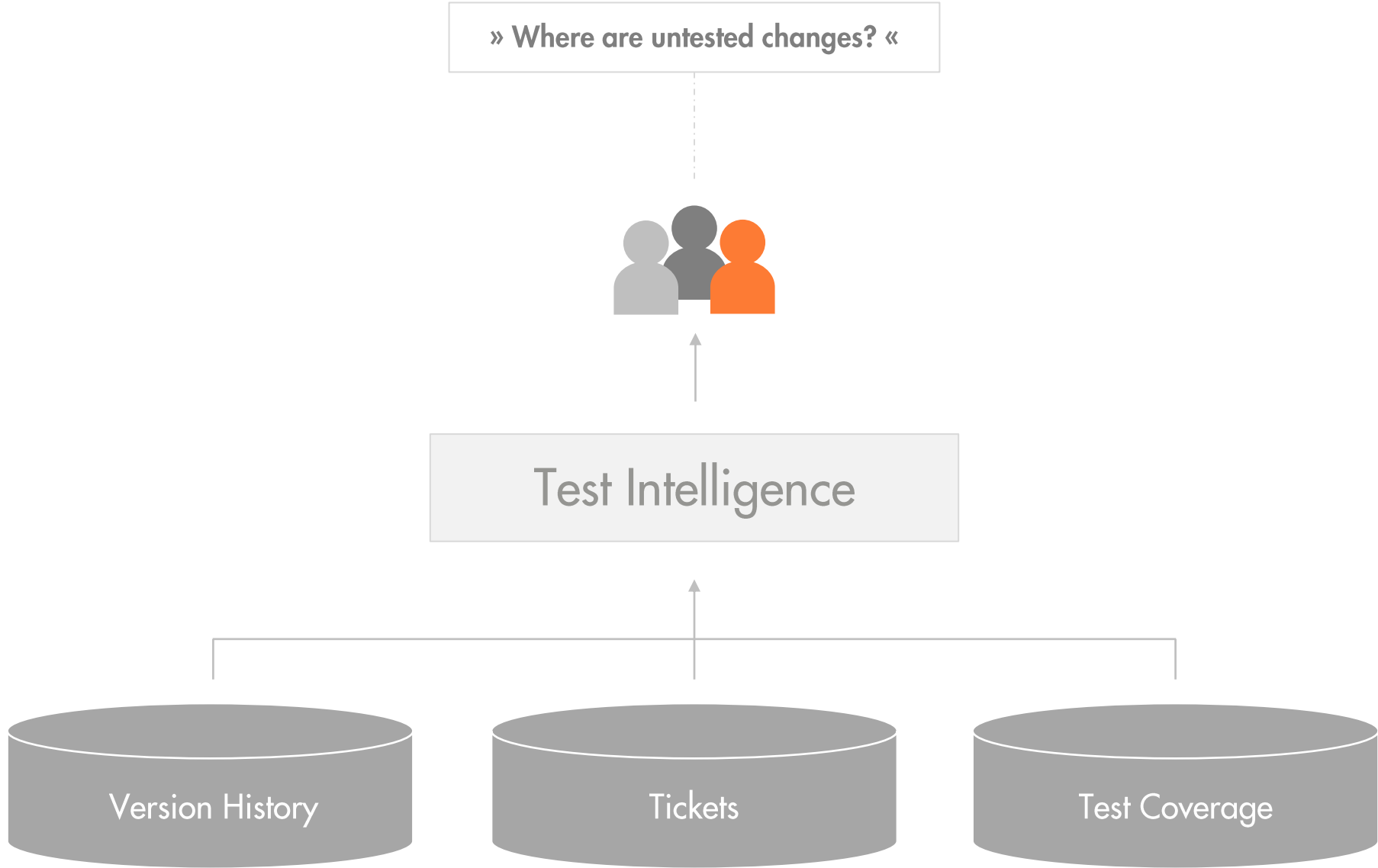


Test Intelligence

Version History

Tickets

Test Coverage





Test Gap Analysis

Reveal Untested Changes in Source Code



September, 25
10:30-12:00 CEST
tmscl.me/tga-249-odin



Fast Feedback From Long-Running Tests

Test Selection for Ever-Growing Test Suites



October, 23
10:30-12:00 CEST
tmscl.me/ts-2410-odin





<https://meetings-eu1.hubspot.com/elmar-juergens>

Contact – Looking Forward to Discussions 😊



Dr. Elmar Jürgens · juergens@cqse.eu · +49 179 675 3863



[Link to slides](#)

CQSE GmbH
Centa-Hafenbrädl-Str 59
81249 München
www.cqse.eu

CQSE