Detecting Breaking Changes in Parametric Chair Releases

Irène Valsoret
Department of Software Maintenance, Institut de la Rive Sèche
Nantes, France
ivalsoret@riverseche.example

Mikel Ordeiza
Laboratory for Parametric Design, Urdal Technical Institute
Bilbao, Spain

Aminah Jelvani
Department of Software Maintenance, Institut de la Rive Sèche
Nantes, France

capstract:2611.00023v1 [cs.SE] 12 Nov 2026

Abstract

We study backward compatibility in eight families of parametric chair packages. For each of 48 adjacent release pairs we replay six downstream operations on independent scale prints, including fitting a cover, loading a trolley and nesting a second chair. Twenty-four releases break at least one previously successful operation. All retain the ability to seat the standard occupant. A release check derived from client contact surfaces correctly classifies 42 pairs, compared with 36 for a global geometric difference and 29 for review of matched drawings. The most frequent consumer to lose compatibility is a previous version of the chair itself, held fixed in the upper position of a stack. Across all subsets containing five of the six clients, an average of three breaking releases remain undetected. The results distinguish compatibility with an installed inventory from conformance to a producer's seating specification.

CCS Concepts

• Software and its engineering → Software testing and debugging.

Keywords

backward compatibility, regression testing, parametric design, client contracts

Reference Format:
Irène Valsoret, Mikel Ordeiza, and Aminah Jelvani. 2026. Detecting Breaking Changes in Parametric Chair Releases. Preprint, November 12, 2026.

1 Introduction

A maintenance release replaced four square feet with four round ones. The chair still accepted a seated figure, fitted under the specified table and could be lifted by its back. A school that had bought thirty chairs could no longer use its floor trolley. The trolley had thirty sets of square holes. Its purchasing specification gave the name of the chair and said nothing about feet.

This is a familiar problem in release engineering. A producer can test every operation it offers and still overlook an operation performed entirely by a client [10, 16]. What makes the chair interesting as a software artifact is that we can obtain a fairly complete set of clients and physically try them. Client programs leave traces in source repositories, but a release study seldom gets to execute old binaries against the exact state that preceded an upgrade. A slipcover can be put back on the shelf for six months and called again with the same arguments.

We built a small registry of parametric chairs whose source files compile to solid geometry. Its clients include upholstered goods, transport equipment and a second copy of the previous chair. The registry maintains a predicate for each operation a client is allowed to perform. It publishes a major version when substituting the new chair in any previously successful operation makes the operation fail. A chair with a low back can therefore accept a person of the specified size and still require a major version increment on account of its relationship with its own raincoat.

Our question concerns the information a maintainer needs before making that increment. Surface change is cheap to compute. A producer can also distribute a declaration of its points of contact. In Section 4 we compare these two release checks with a decision made from renderings alone. The reference outcome comes from independently built fixtures used on scale models (Section 5). A negative result licenses a patch within this registry only. In particular we did not sit on the models.

2 Contracts at the package boundary

Information hiding permits a producer to change features on its side of an interface [16]. Behavioral subtyping places obligations on the replacement when used through that interface [10]. The difficult step here is deciding whether the apex of a back post lies on the producer's side. Its only function as far as a drawing for assembly is concerned may be to terminate the post. A laundry that supplies chair covers can reasonably have a different declaration for the same point. Work on client-observed interfaces makes this distinction for software built against partially documented components [18, 22].

We use a compatibility relation on ordered pairs of releases. It takes a frozen client inventory as a third argument. The registry has no notion of a generally compatible chair. The unit of analysis is an upgrade from one published source archive to the next within a family. A client added to the registry may change a fu-

ture decision about precisely the same geometric edit. The experiment keeps the inventory fixed across both ends of a pair, as in differential testing [11]. It also keeps the client’s configuration fixed. Letting the trolley acquire larger holes would answer a question about coordinated migration.

Mutation studies usually have to decide whether an undetected change is equivalent to the parent [7, 14]. Here equivalence depends on the list of physical operations. We can count a mirror-image change as equivalent for the bench gauge and breaking for the upholstery pattern. The producer supplies neither the expected result for the new solid nor a proposed label. This removes one circularity in earlier archive studies, in which a maintainer’s release tag served as ground truth for a predictor fitted to the maintainer’s changelog [2, 15].

There is also an empirical literature on adaptation by downstream maintainers [1, 19]. A client can often accommodate a breaking change at little expense. The laundry could replace its cloth bag with an elastic one. This observation concerns cost. For a registry that promises installation without client modifications, the amount of elastic available to a downstream maintainer has no bearing on the truth of the promise.

3 Registry and release corpus

Each archive contains one source solid, a named origin and a build manifest specifying units. A build takes the chair through the same sequence of operations used to produce its sale drawing and exports a closed triangle mesh. The shape families were drafted by eight model makers working independently from the same one-page size envelope. All eight started with four legs. The frame could be continuous and two legs could share a runner, provided that the maker could identify the four contacts with a level floor. All models have a nominal seat height of 45 mm at the printing scale of one tenth.

A maker then prepared six maintenance releases in order. We supplied cards giving the reason for a change, such as reducing material, improving the handling grip or simplifying an upholstery seam. Cards did not prescribe a change in dimensions. Makers received the same six reasons in separately randomized orders. They completed all six releases before they saw any client equipment. We retained changes to the build instructions even when they produced indistinguishable solids. The resulting 56 archives gave 48 adjacent pairs. Counting pairwise differences between all releases of a chair would have multiplied the sample size while silently changing the client’s starting point.

Six clients were commissioned separately for each family. The maker of a trolley had the original chair and an outline of the task, but no access to its source archive. We requested a fixed fixture for transport, a fitted cover, a clearance gauge for a desk, an upper chair for a stack, a rigid carrying cradle and a seated articulation figure. These six downstream packages were made once per family. The upper chair was a duplicate of the founding release. In every stack trial the candidate occupied the lower position. A chair presented upside down by itself does not exercise this dependency [4].

We deliberately kept the desk as a fixed outer envelope. During development we had allowed a user to draw the desk forward to accommodate thicker front legs. The test then depended on how willingly the person operating the desk would compensate for a software update. With the stop fixed, an upgrade has a reproducible outcome. A participant who would have moved the desk at home can still interpret a failed clearance test. The other hardware stops and their tolerances were set before the release candidates were printed.

For the figure, the sole operation is lowering it onto the seat with the articulated knees in their recorded position. We used a figure whose arms terminate above the seat. Tests involving armrests could otherwise detect contact with either the left or right hand, which wander several millimeters between placements. This restriction precludes any general claim about a person being comfortable. It yields a repeatable downstream client with two feet that can be returned to known coordinates [17].

4 Release checks

Write P_{ict} for a successful execution of client c on family i at release t. A release check emits a single bit for the six clients together. The physical oracle in Eq. (1) permits a client that already failed to remain broken. This matters late in a family history, when an old client can lose compatibility and regain it after a further change.

$$B_{it} = \max_{c=1}^{6} \{ P_{ic,t-1} (1 - P_{ict}) \}. \tag{1}$$

In particular, the first release to which the trolley can be attached again receives no penalty for the trolley. If the same change tears the cover, the release breaks on the cover. Comparing each candidate only with the founding release would continue to flag all the intervening releases on the trolley, including a release that changed only the inside wall of the backrest.

4.1 Render review

A maintainer examined matched three-quarter views and scale drawings, with one opportunity to rotate each model. Reviewers knew the six operations by name but could neither open a client nor consult its maker. Each pair was independently assigned to one of four reviewers outside its family. We took the reviewer’s first decision. The reviewer could mark any part of the drawing in support of it. Those annotations were withheld from the authors until scoring was complete. Prior work with drawing reviews found that allowing a reviewer to name a probable failure after inspecting the fixture increases the apparent specificity of a review [13].

4.2 Surface and contact checks

The global baseline registers the meshes at the source origin and scores the 99th percentile of bidirectional nearest-surface distances. Points are drawn in proportion to triangle area, with an equal sample from each mesh. We used 20,000 samples in each direction and fixed the pseudorandom stream by the ordered archive identifiers. The threshold of 0.8 mm on the model was set on two development chairs with separate clients. At full scale the same displacement would be 8 mm. A screw-length adjustment

hidden inside a closed solid has no surface distance. The baseline can still detect a decorative rail if the rail occupies enough area.

The contact check instead reads six declarations, one per client. A declaration lists regions, directions of approach and permissible displacements. Each predicate accepts an old and a new envelope and predicts loss of clearance along the approach direction. The cover includes a perimeter constraint around the widest cross section encountered during insertion. The stack registers the three bearing regions on the underside of the fixed upper chair against their counterparts on the candidate. The predicted bits are combined by the same rule as the observations in Eq. (1). This code operates only on archived geometry. It has no run-time access to a printed chair.

We gave the authors of the declarations one afternoon with the original fixtures and the same two development chairs. No one had yet printed release 1 of the eight evaluation families. A loose thread, the coefficient of friction of the transport tray and the preferred fold of the cover were omitted from the representation. Declarations describe interactions at a resolution useful to maintainers and lose information in the process [21]. All score thresholds and the rules for an empty contact region were frozen on the development set. Moving the percentile until the surface check accepted the founding chairs would have used both versions of an identical solid as its only negative example.

5 Physical replay

The operator received randomly ordered pairs of chair and fixture identifiers. Version order was concealed, including whether the item under test was the older member of a pair. Two prints of each solid were obtained on separate build plates. For each print the operator made three attempts per client, restoring the fixture to its stop between attempts. A print passed on at least two successes. We required both prints to pass. A physical execution in Eq. (1) therefore aggregates six trials and gives each plate equal veto power. The 56 solids generated 336 archive-client labels from 2,016 placements. An archive appearing on both sides of adjacent pairs keeps its original label.

Procedural details follow the replay studies of miniature assemblies [3, 6]. We retensioned the spring in the cradle before the day’s randomization and used witness marks to reset the desk stop. The cover passed if its lower hem reached all four marks without an operator pulling the hem sideways. We photographed the hem with the chair upright. A corner lifted by a projecting back post is quite difficult to see with the model in the hand. For the stack the upper chair had to settle against all three bearings under its own weight. An operator could steady the lower chair at its feet, which belong to the lower client’s environment in this operation.

A major-release recommendation can be composed of several correct client predictions or one correct prediction about the wrong client. We score release decisions first. For diagnosis we also record each client that newly fails. A reader buying only the transport package may find the latter measure more useful. All rates below describe this finite corpus. The model makers chose their edits, so binomial intervals obtained by treating the 48 releases as independently sampled commercial chairs would convey a precision the design does not support.

6 Results

Of the 48 ordered pairs, 24 preserve every previously successful operation. The other 24 cause 31 individual losses. Eighteen releases lose one client, five lose two and one loses three. All 48 accept the seated figure when its predecessor does. During the experiment a maintainer considering only the operation printed on the front of the package would have issued 48 nonmajor releases. The families contributed between two and four breaking changes apiece. Table 1 reports counts with the same physical outcome on both sides of each comparison.

Table 1: Release recommendations relative to independent fixture replay. Rows describe 48 decisions each.

Check	True major	False major	Missed major	True compatible
Render review	14	9	10	15
Global surface	18	6	6	18
Client contacts	21	3	3	21

Client contacts yield 42 correct decisions. The surface check yields 36. The paired results are narrower than these two margins might suggest. Nine pairs change from incorrect under surface checking to correct under contact checking and three change the other way. Both get 33 right and three wrong. At the level of the family, the eight differences in the number correct are $2, 1, 1, 0, 2, 0, 1, -1$ in archive order. Removing any one family leaves an advantage of four to seven correct decisions on the remaining 42 transitions. The small number of families makes extrapolation to another collection of chairs quite uncertain.

The types of loss in Figure 1 explain why using a seated figure as an end-to-end test misses so many failures. Every failure requires a second artifact. The most demanding such artifact by count is another chair. A stack can fail on a barely visible change in the slope of a bearing surface while both chairs retain the same outer dimensions.

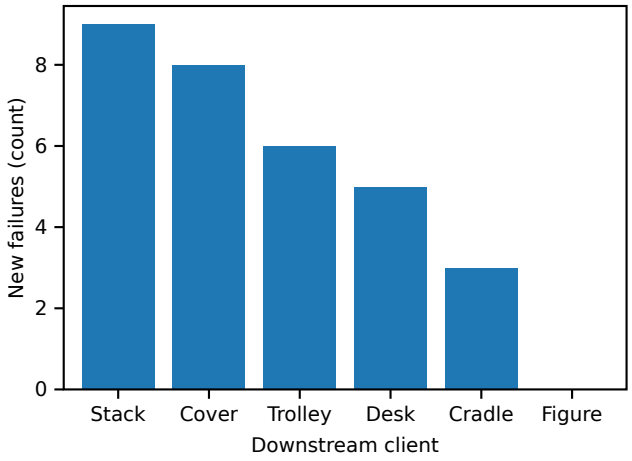


Figure 1: New physical failures across 48 upgrades. A release can contribute to several bars.

Table 2 gives the primary finding from examination of all six incorrect contact decisions. Each row corresponds to a different transition. Reviewers classified a decision after viewing its six replays in order. If two clients could account for a false alarm, they recorded the one with the larger modeled violation. This choice affects attribution of an error to a client and leaves the release counts in Table 1 intact.

Table 2: Source of each incorrect contact decision. The magnitude is measured on the scale model.

Decision	Client	Replay finding
Miss	Cover	A raised seam caught on a back post
Miss	Cradle	The frame twisted under the spring
Miss	Stack	A fourth point touched before the bearings
False alarm	Cover	The hem stretched over a 0.6 mm projection
False alarm	Trolley	A foot rotated inside the sleeve
False alarm	Desk	A rounded edge passed under the stop

The false alarm on the trolley is the converse of the incident that motivated this work. The replacement foot had a flattened heel. It could turn slightly during insertion into its square sleeve. The declaration uses the orientation in which the chair leaves the build system. Rotation increased clearance in replay. A constraint language with bounded rotation would address this case but would also change the test for a desk, whose operator is forbidden to rotate the candidate. We have left the predicate as frozen for the eight families.

6.1 Client discovery

For each prefix of a fixed enrollment order we recalculated the physical decision using only the clients in the prefix. Figure 2 shows how many of the 24 actual major releases would have been recorded as such. The order follows acquisition dates and was determined before replay. It is not an optimum test order. The jump after installing the laundry package comprises six releases that were acceptable to all previously enrolled clients. The eight failures of that package in Figure 1 include two already identified elsewhere.

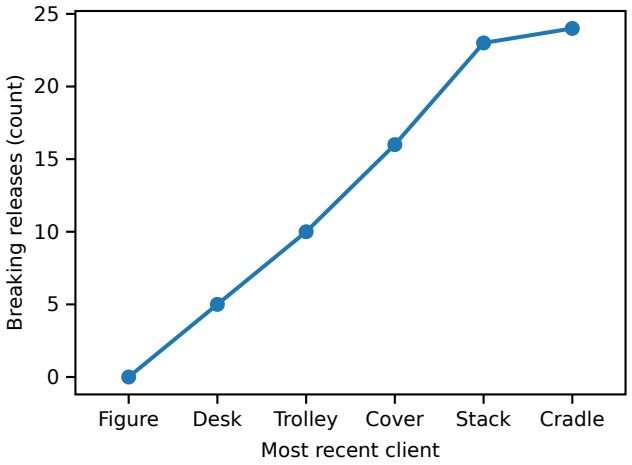


Figure 2: Breaking releases recognized when clients enter in acquisition order. Each point adds the named client to the preceding inventory.

We investigated how much the ordering itself drove this curve. For a subset of size k drawn uniformly from the six clients, a transition which loses m clients escapes observation with probability given by Eq. (2). We enumerate the subsets without replacement. The calculation uses only the multiplicities above, so it remains valid if two clients always fail together.

$$q(k, m) = \binom{6 - m}{k} \Big/ \binom{6}{k}. \tag{2}$$

Figure 3 uses Eq. (2) to average coverage over all subsets. Buying a single client in that experiment would detect $31/6$ breaking releases on average. Buying five still loses three releases on average, because the omitted client can be the only consumer of a changed surface. A completely cataloged set of operations for the producer cannot replace the missing consumer by running the producer more times.

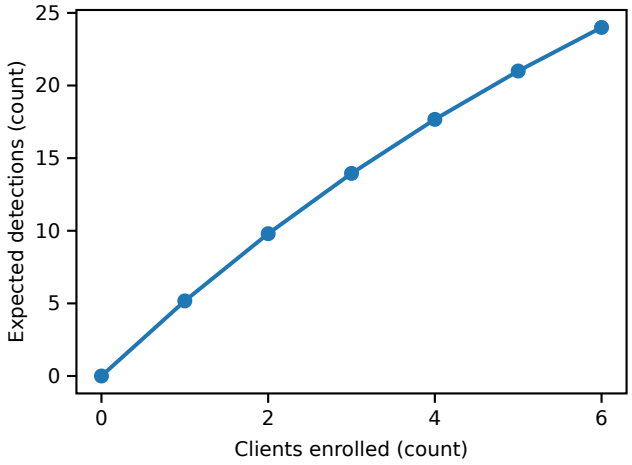


Figure 3: Expected physical detections over equally weighted client subsets. End points use the empty and full inventories.

7 Threats to validity

We test loss of an established operation. We have no oracle for a consumer which has never worked, even if a maintainer specifically intended to make it work in the new release. Such a claim would require a test for newly offered functionality. The asymmetric treatment of gains has the incidental advantage that machinists can repair a previously broken trolley interface without classifying the repair as another break.

Scale affects cloth and gravity differently. A full-size piece of fabric does not acquire the bending stiffness of a scale cover simply by being cut ten times larger [8]. The cover result is about a garment of the stated construction. Extrapolation to a production textile calls for new replay data. A solid chair at this scale is also less susceptible to joint failure than its assembled counterpart. We did not introduce an artificial joint failure rate into the client labels to compensate.

A producer might already know about a prominent downstream trolley. Our reviewers lacked detailed information about the fixtures, so the comparison measures the value of supplying that information along with an executable check. It cannot separate better documentation from better geometry. A blind review of the actual contact declarations, with the same time allowance but without an interpreter, would be a way to make that distinction. Earlier studies of interface summaries suggest that a reviewer can use a drawing of a boundary even when no executable specification is available [9, 20].

The enrollment exercise also limits the interpretation of the word major. A seventh client could require a larger change than all six considered here. One of the model makers subsequently supplied a clothespin clipped to a chair post to hold a place card. We could find no operation in the six declarations that bounds the force exerted by that clothespin. Giving it standing would require the registry maintainer to accept an additional consumer and obtain its old configuration. Revising the name of the release after a wedding at which the place cards had already fallen off would not repair that dependency.

8 Conclusion

A maintainer with the full set of archived fixtures can answer the compatibility question for each of these releases by replay. The practical difficulty is collecting the fixtures while the old chair is still available. By the time a purchaser reports that a chair no longer fits its cover, there may be no record of which of the two used to fit inside the other at which revision. The date and direction of the last successful insertion belong in the client manifest.

A Repeatability and correspondence

We labeled the initial solid zero and the maintenance releases one through six. Archive numbers increase only within families. Sorting globally by print date would put some successors before their predecessors because the plates were randomized. Assignment to physical test was keyed on the family, archive, print and client. Repeating a placement generated a new row under that key. A correction of a mistyped mark retained the previous row and an operator signature, following the procedure of Eiranova [5]. Only the last signed observation of an attempt was eligible for the majority count.

Coordinate correspondence across two versions is by the client origin. A common alternative is to center each candidate on its own bounding box. Lengthening the front of a chair with this rule would move its back in the opposite direction on the comparator, even if the back were fixed in the source file. The desk has a stop at the rear and the transporter indexes the rear pair of feet first. Both would then acquire an apparent change in clearance. Keeping the shop origin in the archive removes this ambiguity and exposes the separate problem of a maker editing the origin. Such an edit can itself be breaking for fixtures that approach under numerical control [12].

The seating client has no fitted sleeve and a relatively generous articulation envelope. To check that its series of successes was recorded correctly, a second operator repeated the founding and final placements for all eight families from the still images and reset marks. All sixteen checks agreed. This provides little information about false negatives near its tolerance since none of these seats was close to its upper limit. A useful future control would issue a chair deliberately too high for the figure. It would belong in a separate calibration series, since including it in the 48 upgrades would give the producer a change targeted specifically at a known client.

B Accounting for intersections

For $k = 5$, the 18 transitions with a single failing client each have a one-in-six chance of being missed. No transition with two failing clients can escape a five-client subset. Subtracting the expected three misses from 24 gives the penultimate ordinate of Figure 3. At $k = 2$ there are fifteen subsets. A transition with three failures escapes in three of them, whereas one with one failure escapes in ten. This difference accounts for the modest

curvature in the plot without assuming any statistical independence between the trolley and the desk.

On a registry that sells client suites separately, an additional observable would be the number of copies of each downstream item. One cover made in a thousand sizes for the same chair still supplies only one recorded loss here. Conversely a single preserved trolley for a discontinued chair family supplies a full loss. A sales-weighted estimate would need both quantities at the time of release. None of the archives records the number of times a single trolley has been lent to someone else.

Acknowledgments

We thank the fixture makers for preserving the founding upholstery patterns and the original trolley sleeves.

References

[1] Néla Avadour and Rui Petraves. 2023. Labor budgets for downstream refitting. *Transactions on Design Archives* (2023).

[2] Élise Carlenot. 2020. When a release tag is an outcome label. *Journal of Release Studies* (2020).

[3] Inael Dorescu. 2020. Repeated placement of compliant textile gauges. *Transactions on Design Archives* (2020).

[4] Piera Edranel. 2017. Upper and lower members in a nesting fixture. *Review of Assembly Metrology* (2017).

[5] Metka Eiranova. 2018. Audit records for corrected bench observations. *Review of Assembly Metrology* (2018).

[6] Maret Isselund. 2022. Blinding specimen order in reversible assemblies. *Review of Assembly Metrology* (2022).

[7] Yue Jia and Mark Harman. 2011. An Analysis and Survey of the Development of Mutation Testing. *IEEE Transactions on Software Engineering* (2011).

[8] Aimée Jorellet. 2015. Fabric stiffness in miniature fitted goods. *Journal of Model Manufacture* (2015).

[9] Sandor Lemayre. 2022. Diagrams of consumer boundaries in code review. *Journal of Release Studies* (2022).

[10] Barbara H. Liskov and Jeannette M. Wing. 1994. A Behavioral Notion of Subtyping. *ACM Transactions on Programming Languages and Systems* (1994).

[11] William M. McKeeman. 1998. Differential Testing for Software. *Digital Technical Journal* (1998).

[12] Abel Morlensen. 2024. Origins as dependencies in part archives. *Transactions on Design Archives* (2024).

[13] Cezia Narelli. 2020. Inspection drawings and post hoc fault localization. *Review of Assembly Metrology* (2020).

[14] A. Jefferson Offutt and Jie Pan. 1997. Automatically Detecting Equivalent Mutants and Infeasible Paths. *Software Testing, Verification and Reliability* (1997).

[15] Talha Oruseki. 2022. Declared and observed costs of a dependency update. In *Proceedings of the Velin Software Colloquium.*.

[16] David L. Parnas. 1972. On the Criteria To Be Used in Decomposing Systems into Modules. *Communications of the ACM* (1972).

[17] Sabine Pellervaux. 2016. Joint envelopes in scale seating trials. *Journal of Model Manufacture* (2016).

[18] Enno Riemhout. 2019. Observing consumer contracts at an artifact boundary. *Journal of Release Studies* (2019).

[19] Odda Tenviksen. 2018. *Repair work in small client packages.* North Strand Monographs.

[20] Ida Tolsenius. 2024. Review time for geometric interface declarations. In *Proceedings of the Velin Software Colloquium.*.

[21] Orsa Veirantes. 2021. Selecting interfaces for a clearance declaration. *Journal of Model Manufacture* (2021).

[22] Ilma Veskinen and Clovis Navret. 2021. Interfaces in translated mechanical packages. *Transactions on Design Archives* (2021).