**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

# Your RTO Is a Hypothesis: Drill-Based Verification of Recovery Objectives in Multi-Cloud

Narayana Rajasekhar Vemburaj
*Technical Account Manager Amazon Web Services, Inc*

**ARTICLE INFO**

Received: 03 Jan 2022

Revised: 20 Mar 2022

Accepted: 29 Mar 2022

**ABSTRACT**

While all regulated businesses have recovery goals, few can demonstrate them. A business-continuity document is a theory about how a complex, drifting, multi-cloud estate performs on its worst day, and most businesses run a tabletop exercise against the architecture as it was when the document was created, at least once a year. This article does not treat RTO and RPO as objectives but rather as claims that can be measured and it outlines a discipline for certifying RTO and RPO – Recovery Certification – that I use for certification and which is based on a drill. Recovery objectives can only be validated by some evidence of it: by fielding full-path drills that fail on actual workloads, by providing data-loss and recovery-time distributions and confidence bounds rather than single best-case figures, and by providing expiry rules by which a drill result goes out of current currency as the estate drifts. Those partial drills are only at the estate level on a dependency graph, and the untested seams between clouds are where multi-cloud recovery fails. Or the distance between the recovery a document has announced and the recovery a drill a ledger has used remains hidden until the worst day of the year at which point only the latter matters.

**Keywords:** Disaster recovery, RTO, RPO, recovery objectives, resilience verification, multi-cloud, drills, business continuity, site reliability engineering.

## I. FIVE A.M. ON A SATURDAY

This template, modified in MS WordThe business-continuity document said four hours. I had read it the week before the drill, as you read a document you will be testing. RTO 4 hours, RPO fifteen minutes, reviewed annually, last validated fourteen months earlier. The validation on record was a table top: 11 people were in a conference room and walked through the failover runbook, agreed it was going to work and signed the page. At 05:00 on a Saturday we ran it for real. The scene is a composite of drills I've run and drills I've sat through in enterprise DR planning at a big cloud provider, ops in the streaming industry, and regulated financial services, with identifying details redacted. The failure sequence is accurate, as each element has occurred at least to my personal experience. Everyone was concerned with the database restore and it was fine. We played 92 minutes until we got the game back and we were in the plan comfortably. One by one the parts that nobody had been drilling came in. There was no workload identities associated with the IAM of three services since the last drill, and the federation config had not been mirrored by the pipeline, but had been mirrored with the services that existed when the pipeline was written. It can take 40 minutes to diagnose and an hour to hand patch. The five-minute runbook budget for the DNS hit a record that was supposed to be 60, but instead ended up at 3,600 seconds after a migration two quarters ago. When traffic finally got underway, the application tier again took 40 minutes to come out of the brownout, warming connection pools and caches against an existing database that was not responsible for this traffic, and a licensing callback in one vendor component persisted in trying to reach an endpoint that was only in the dead space. Estimate of the time it takes to recover from the cut to the estate inside its error budget is 11 hours and 20 minutes. According to the document, four. Nobody had lied. The meeting had been the test, and the signature had been the proof of a four-hour number that now had no meaning, for an architecture that is no longer. A declared RTO is a theory of what a drifting estate does when it's at its worst. The hypothesis was finally supported by the data on drill day. Fig. 1 lays out the timeline of that morning against the declared objective.

*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

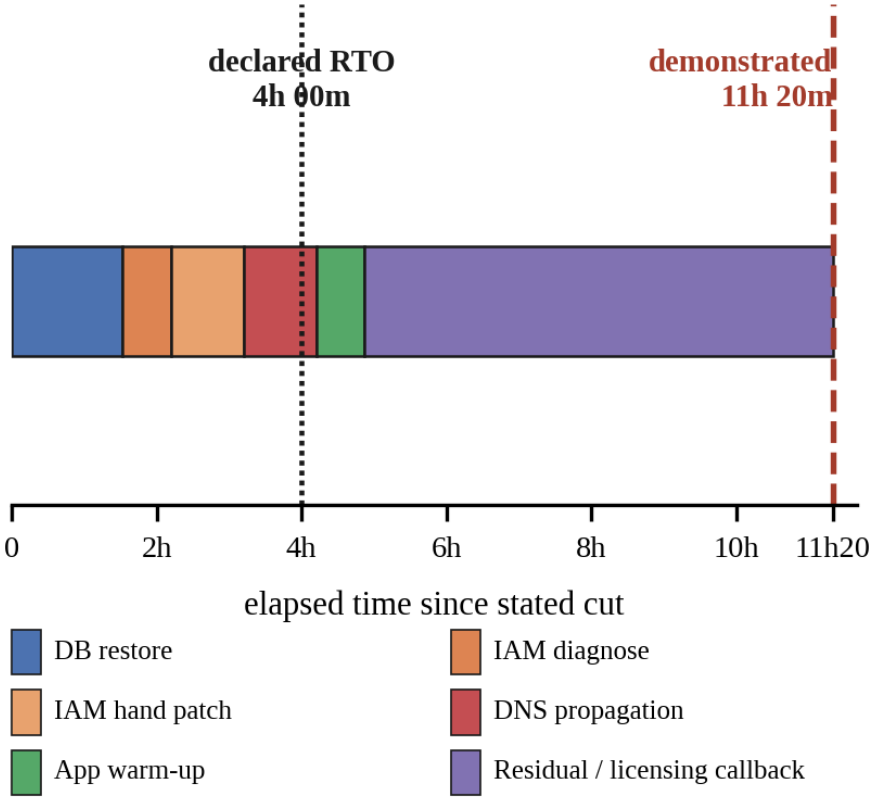



**Fig. 1. Timeline of the opening drill. Six untested seams, none individually large, extend recovery from a declared four hours to a demonstrated 11 h 20 min.**

## II. Declared Versus Demonstrated

I can make all the regulated enterprises, that I have been involved in or supported, say what their targets are to recover on request—after all the control regime requires them too. Financial regulators in the United States have documented and tested capability to maintain continuity with the business [5] and the EU's Digital Operational Resilience Act requires financial entities to conduct tests of their ICT continuity plans for extreme scenarios [6]. This is another matter, and as I've come to find out over time, the two answers are quieter than they say, one PR at a time.

DR literature is short of being candid: there is a rift here. There is a healthy body of work on recovery architectures for cloud and multi-cloud; from backup-and-restore to pilot light and warm standby to active-active, the providers do their share of documentation [4] and academic research examines RTO and RPO trade-offs between on-premises and cloud deployment [7]. It also notes, rightly, that evaluation of DR is unstandardized, that is, there is no agreed-upon way to declare whether a given goal is true or not. Frameworks explain you how to construct recovery. Not much will tell you how to check it.

This is the verification layer contributed in this article. I'm calling it recovery certification, and there are 4 rules. A full-path drill that fails over real workloads is the only type of drill that can certify a recovery claim. The certified value is not a single best-case number, but rather a distribution of values over drills with a given percentage of confidence. Each drill result is expired after a certain procedure (change volume in the recovery track) and will no longer count. Only along the dependency graph, partial results make up to claims at the estate level, thus, the untested seams between clouds are exactly that—seams in the certificate.

This is the declaration of recovery as stated in the document. The drill ledger is where you will see demonstrated recovery. The discipline is the machine that keeps the second current moving and only the second should be permitted to meet an auditor, a regulator or you. Table I states the discipline's four rules in full.



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

**Table I. The Four Rules Of Recovery Certification**

| RULE | REQUIREMENT |
|---|---|
| 1 | Only a full-path drill against real workloads may certify a recovery claim. |
| 2 | The certified value is a distribution over drills at a stated confidence, not a single best-case figure. |
| 3 | Every drill result expires under a wall-clock cap and a per-path change budget. |
| 4 | An estate-level claim requires every vertex and every cross-cloud seam in scope to be covered by an active drill. |

## III. What Counts as a Drill

Not every exercise that gets called a DR test exercises the claim, so the taxonomy comes first.

The plan is explained to the humans with a tabletop. I execute them and I appreciate them; they expose issues of ownership, out of date contact lists, and runbooks that no longer make sense. However, a tabletop is a training for a meeting, not the actual recovery. It confirms that people accept the plan would work - the opposite of what the opening scene assumed. A tabletop cannot certify a claim, but can expire a claim (if someone notices the topology moved).

A component restore will restore just one vertex: restore this database from backup, rebuild this cluster from source, replay this topic into a new cluster. These are inexpensive and can be performed frequently and they find and detect real rot, not only the backup that's been removed, but the row count that passed while the contents were garbage. As in my previous article in this series, it is important to verify and make an exercised, verified restore a hard precondition before deleting persistent state [CROSS-REF: companion on the deletion gate]. A restore that doesn't have verified contents is something of a hypothesis in the past.

The only drill in the taxonomy that does exercise an RTO claim end-to-end is the full-path drill and there are four requirements for a drill to count towards the claim. Content-based data restore with the actual recovery mechanism, to the actual recovery site. There is a traffic cutover through the actual DNS/Global load-balancer change (the TTLs actually in the zone, not simulation through hosts-file). Then there's dependency warm-up to steady state, with the clock ticking until the estate serves within its error budget [8] – the difference between the “the process started” and “the service recovered” is the forty minutes of brownout the scene paid for. And failback, which is a second failover without that adrenaline, and this is where the data that is written during the failover is reconciled with the region that has come back. Table II summarizes what each exercise in the taxonomy can and cannot certify.

**Table Ii. Drill Taxonomy And Certification Power**

| DRILL TYPE | EXERCISES | RTO? | EXPIRES ? |
|---|---|---|---|
| Tabletop | Plan comprehension, ownership, contact lists | No | Yes, if drift noticed |
| Component restore | One vertex: backup / restore correctness | No, vertex only | Yes, for that vertex |
| Full-path drill | Restore, cutover, warm-up, failback, end to end | Yes | Yes, for the path drilled |



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

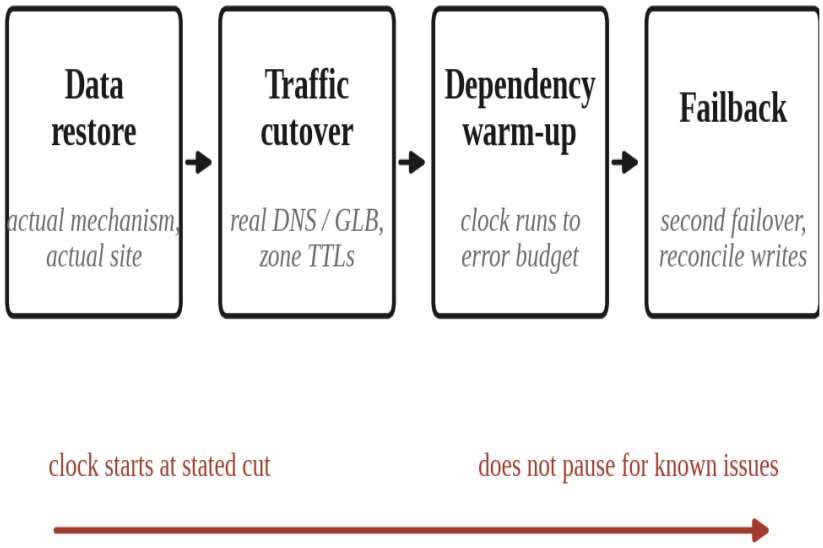



**Fig. 2. The four phases a drill must cross to certify an end-to-end RTO claim, run under the two clock rules.**

Two clock rules, both of which teams are not willing to follow. The clock does NOT start when the team gathers, it begins at the stated cut. And the clock does not pause for known issues. Your score on a drill that was stopped twice to correct the IAM mirror should be recorded as being "a drill in which the clock was stopped twice to correct the IAM mirror."

**Table III. The Two Clock Rules**

| RULE | STATEMENT |
|---|---|
| Start | The clock begins at the stated cut, not when the team gathers. |
| Pause | The clock does not pause for known issues; stoppages are recorded, not excluded. |

## IV. Recovery Time Is a Distribution

The most common failure mode for DR reports is arithmetic: a report of a point where the claim requires a distribution with a sample size attached

The "RTO four hours" evidence that is usually provided is the last drill, where the duration was 3 hours 40 minutes. That's one draw from a random variable, and typically a good one as drills take place at 5am on a Saturday when the A-team is awake and the runbook is open. The real events are sampled from a worse distribution. The honest minimum is to treat each full-path drill as a measured recovery time T_i, retain all of them, and report the empirical distribution: the median gives you an idea of what a good day looks like, the worst drill you have seen gives you an idea of what you actually have to do to recover, and the range gives you an idea of the amount of engineering versus dumb luck. Fig. 3 shows the shape such a sample takes once enough drills accumulate.



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

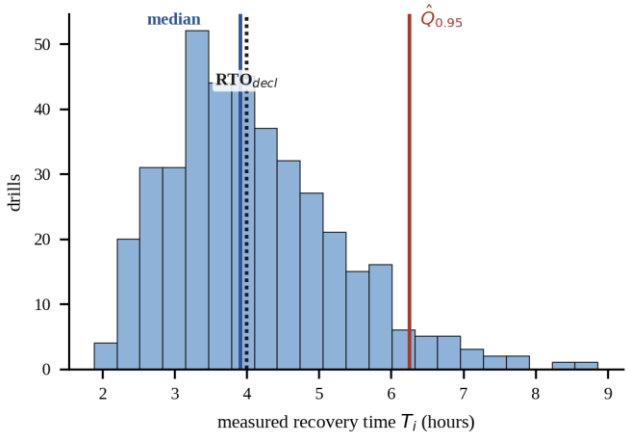



**Fig. 3. Simulated recovery-time draws from a drill program: the median flatters, the observed 95th percentile warns, and both sit either side of the declared RTO.**

Certification is then a statement of a probability. “RTO met” should imply that Pr(T ≤ RTO_decl) ≥ p_target at a given confidence level, where T is the number that would result from a real cut, RTO_decl is the number from the document, and p_target is the fraction of days that the business is claiming to survive at said confidence level within it. The practices which follow are all a pass-or-fail exercise on the declared bound; the calibration companion has embodied in the finite-sample machinery to certify on the one-sided 95 percent Wilson lower bound of the observed pass rate, so print n next to every claim [CROSS-REF: companion on finite-sample certification of calibration].

The numbers are chilling, and should be vocalized at the time of certification. So, if all your drills have been under the target, then, by the standard rule of three, [1] the upper limit on your true RTO miss probability is still about 1/4. If every drill you've ever run has been under the objective, (i.e. 3 years of perfect execution), you still cannot rule out missing the RTO more than once in five. If you wish to certify for that level of miss rate at that confidence you must perform about 59 consecutive clean full-path drills under the exact binomial bound, which is defined later in the one-sided Wilson gate, and will become invalid well before you reach that point because of the expiry rules set out in the next section. Table IV lists the exact bound behind that arithmetic.

**Table IV. Sample Size Vs. Certified Lower Bound, All Drills Passing**

| N | LW(N,N) | IMPLIED MISS-RATE BOUND |
|---|---|---|
| 3 | 0.53 | 0.47 |
| 12 | 0.82 | 0.18 |
| 36 | 0.93 | 0.07 |
| 59 | 0.96 | 0.04 |

One-sided 95% Wilson lower bound, z = 1.645 (Eq. 5), for an unbroken streak of n clean full-path drills. n = 59 is where the bound first clears p_target = 0.95.

I don't read that arithmetic as a counsel of despair. I read it as being a design constraint that has three consequences. Claims should be coarse and honest, "demonstrated at the median, unexercised at the tail. The only lever which would make the bound better is drill frequency, which is the business case for automation of the drill until it is boring. And the extent of a full-strength claim should be limited: a monthly credit-critical path is good for over an entire estate drilled yearly. RPO is less ceremonious when submitted, since the measure is already in place. In the streaming-DR article in this series, I said that the honest RPO for an asynchronously replicated system is the replication lag at the time of the cut: RPO(t_cut) = t_cut-t_replicated (t_cut) [CROSS-REF: companion on multi-region streaming failover]. Measuring lag takes place constantly, in production, while the drill measures the cut mechanics every second of every day. A declared 15-minute RPO should be investigated in reference to the lag distribution at Tuesday



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

peak, not at 05:00 on a drill Saturday when the pipeline is coasting. Any team that plots that distribution the first time learns something; several teams have worked with found out their declared RPO was violated for a couple of minutes during each and every nightly batch window, which is exactly when the actual regional failure is statistically most likely to arrive.

## V. Drill Results Expire

A certificate from a drill certifies the estate drilled. The business you are running today isn't the same business. The concept of this series has been used twice before. If the input to the blast-radius guard is stale, and that is the only one, then the guard will not act on the input. The live-state change gate attaches an approval to a snapshot of the live system, and will not pass the approval over time [CROSS-REF: companion on change approval against live state]. A drill result is like that: it is evidence of some system state that silently fails whenever there is drift, and is honest only as long as the drift is small.

So, every entry in the drill ledger carries a validity condition, and I write it the same fail-closed way. A drill $D$ of recovery path $P$, run at time $t_D$, is valid at time $t$ only while

$$\mathrm{Valid}(D,t) \;=\; (t - t_D \le \tau_{max}) \wedge (\Delta C_P(t_D, t) \le \kappa_P) \quad (1)$$

where $\tau_{max}$ is a wall-clock cap, $\Delta C_P(t_D, t)$ counts the changes landed on any component or seam of $P$ since the drill (deploys, infrastructure changes, IAM and DNS changes, topology moves, provider migrations), and $\kappa_P$ is the change budget the path is allowed to absorb before the evidence stops counting. Both thresholds are declared per path, in review, the way the guard's bounds are. Both conjuncts fail closed. If you cannot compute the change volume on the path, because some of the path changes outside your change-tracking, the result is expired; unmeasured drift is not zero drift. The recoverability-tiered companion applies a wall-clock freshness bound of exactly this shape to restore proofs; the second conjunct, a per-path change budget that expires evidence on drift before the wall clock gets a vote, is what this article adds. The opening scene is the argument for the second conjunct. Fourteen months of wall clock had passed, but the killing drift was three services and one zone migration, all of it visible in change records nobody was required to reconcile against the DR claim. A two-year-old drill cannot vouch for today's topology, and neither can a two-month-old drill on a path that ships daily. Expiry is also what keeps the discipline operable. An expired result does not demand a full re-drill of everything; it tells you which path expired and which changes expired it, which scopes the next drill to the drift.

## VI. The Seams Are the Estate

You can't full-path drill everything at certification frequency, so partial results have to be created, and that's where multi-cloud DR falls by the wayside. Model the estate's recovery as a dependency DAG $G$: identity comes up before the services that authenticate against it, the database before the applications, service discovery before traffic. For maximal parallelism, and at the instant it was priced, the price of every handoff between components is zero, estate recovery time is the critical path, Fig. 4 shows the shape of the problem: vertices sit inside a single cloud, seams cross between them.

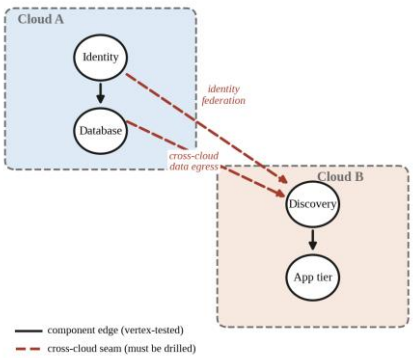



**Fig. 4. A two-cloud recovery graph. Component drills certify the vertices; only a drill that crosses the dashed seams certifies identity federation and cross-cloud egress.**



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

$$T_{estate} = \max_{\pi \in \Pi(G)} \sum_{v \in \pi} T_v \qquad (2)$$

where $\Pi(G)$ is the set of dependency chains in $G$ and $T_v$ the recovery time of component $v$: the longest chain of dependent recoveries [12], which no amount of parallel heroics elsewhere shortens. The vertices of that graph are certified by component drills. The expensive edges are the cross-cloud seams: identity-federation between cloud providers, DNS cutover across cloud zones/registrars, data egress from one cloud to another, replication promotion. The edges — the transition between one recovered thing and the next — are only certified by drill operations that traverse them; with multi-cloud, the expensive edges are the cross-cloud seams: identity-federation between cloud providers, DNS cutover across cloud zones and registrars, data egress from one cloud to another, and replication promotion. All of those seams were in the opening scene, and none of them had a vertex to call home, hence component testing could not have been able to find all of them. A second composite, of same shape, but of different seam. A team I worked with had successfully certified their database restore for 40TB with quarterly component drills using a 3TB sample restore of 40 minutes that resulted in a linear time of 9 hours for the full restore of the estate, which fit well within a 12-hour RTO. The first full-path drill was the real restore and the real restore ran at about 1/5 of the extrapolated rate because of cross-cloud egress throttling, per-object request overhead on millions of small objects, and a rehydration tier that the sample wasn't large enough to access. The actual full restore number was measured in days while the number was honest and calculated with full restore. The component drill was on the money with its component. The statement that it was used to support was on an edge that the drill never crossed. So the composition rule is strict, and it is the fourth rule of the discipline: an estate-level claim is certified only if all vertices and all edges of the recovery graph within the claimed scope are included in an active drill (including the seams). It isn't sufficient to cover only the critical path, since the critical path is calculated from measured times (and a big unmeasured seam on a chain that was thought of as non-critical is its critical path that moves it). The critical path does not lessen what is required in the certificate, it just tells you the drill to do next. Composed per-component estimates are planning tools, they are not certificates, and adding per-component p95s doesn't lead to the p95 of the sum, unless you wanted to prove it in a post mortem.

## VII. The Claim, Written Down

Recovery certification fits on a page, and a page is what an audit or a design review can actually argue with, so here it is.

A declared RTO is the claim

$$\Pr(T \le \mathrm{RTO}_{decl}) \ge p_{target} \qquad (3)$$

where $T$ is the recovery time the estate would produce on a real cut, $\mathrm{RTO}_{decl}$ is the number in the document, and $p_{target}$ is the fraction of bad days the business claims to survive inside the objective. Almost no continuity document states $p_{target}$, which means the claim gets read as certainty, and certainty is the one target no finite drill program can certify.

Each full-path drill $i$, run under the two clock rules, yields a measured recovery time $T_i$ and one Bernoulli trial of the claim,

$$X_i \in \{0,1\}, \qquad X_i = 1 \Leftrightarrow T_i \le \mathrm{RTO}_{decl} \qquad (4)$$

Collect $k = X_1 + \cdots + X_n$ passes across the $n$ drills still valid under the expiry rule, and certification is the same finite-sample gate the calibration companion built, reused whole: compute the one-sided 95 percent Wilson lower bound $L_W(k, n)$ on the true pass probability, the score interval at $z = 1.645$ [11] , and certify only while

$$L_W(k, n) \ge p_{target} \qquad (5)$$

The bound is the certificate, with one honesty clause attached: $n$ drills certify the drill-day distribution, and drill day flatters, so the certificate is an optimistic bound on the real-cut claim, which is one more reason to read it coarsely. The distribution's job is diagnosis. With $\hat{F}_n$ the empirical distribution of the measured $T_i$, the ledger also prints the observed tail, Fig. 5 plots that bound as the clean streak grows.



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

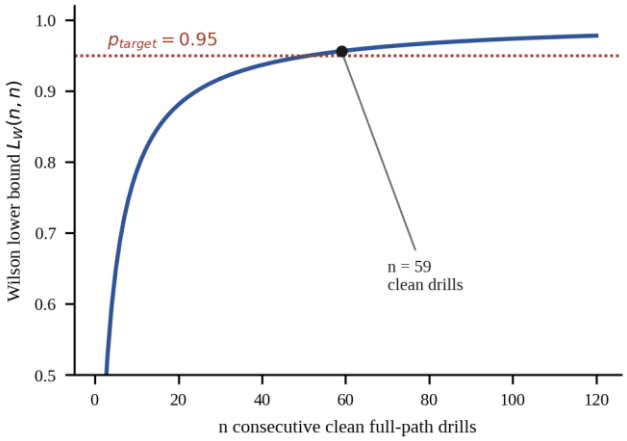



**Fig. 5. One-sided 95% Wilson lower bound for an unbroken streak of n clean full-path drills. The bound clears p_target = 0.95 only at n = 59.**

$$\widehat{Q}_{0.95} = \min\{x: \hat{F}_n(x) \geq 0.95\} \qquad (6)$$

next to the objective and next to the median. At drill-program sample sizes $\widehat{Q}_{0.95}$ is close to the worst drill on record, and that is fine. A worst-on-record above the declared objective is a warning years before the Wilson bound can say anything.

RPO needs less machinery because the gauge already exists. The data exposure at failover is the replication backlog at the instant of the cut,

$$RPO_{exp} = L(t_{cut}) \qquad (7)$$

with the lag $L(t)$ and its drift $dL/dt$ exactly as the streaming-failover companion defines them. A drill samples $L$ once, at a chosen hour. Production writes down the whole distribution every second, so the RPO claim certifies against Tuesday-peak lag, not drill-Saturday lag.

Composition, then. For dependent recoveries in series, the component times lower-bound the estate,

$$T_{estate} \geq \sum_{v \in \pi^*} T_v + \sum_{(u,v) \in \pi^*} S_{uv} \qquad (8)$$

where $\pi^*$ is the critical path from the earlier equation, $T_v$ the recovery time of component $v$, and $S_{uv} \geq 0$ the seam time of the handoff from $u$ to $v$: the federation, the cutover, the egress. The earlier critical-path equation priced these handoffs at zero; the seam terms are what that simplification costs. Component drills measure the $T_v$ and say nothing about any $S_{uv}$, and the opening scene is what an unmeasured seam term looks like on the day it finally gets measured. So the estate certificate is a conjunction over the whole graph in scope,

$$Cert_{estate}(t) \Leftrightarrow (\forall\, v \in G: Valid(D_v, t)) \wedge (\forall\, (u, v) \in G: Valid(D_{uv}, t)) \qquad (9)$$

every vertex $v$ and every seam $(u, v)$ covered by its own drill, $D_v$ or $D_{uv}$, currently valid under the expiry rule. The conjunction runs over $G$ and not over $\pi^*$ for the reason the composition rule gave: until every seam in scope has a measured $S_{uv}$, the true critical path is not computable, and an incomputable critical path is no license to certify the one you can compute. An unmeasured seam is not a zero seam; the same way unmeasured drift is not zero drift. One expired vertex or one never-crossed seam and the estate claim decomposes back into component claims, which is the honest state it was in all along.

And the asymmetry, which the ledger should state out loud. A drill with $T_i > RTO_{decl}$ does its work immediately: read as the certainty most documents imply, the claim is dead on one counterexample, and read with an honest $p_{target}$, the zero drags $L_W(k, n)$ down at once, usually below any target worth declaring at the sample sizes drills allow. Passing drills work slowly. A spotless record of n still concedes a miss rate near $3/n$ by the rule-of-three arithmetic above, and expiry keeps restarting the clock. Falsification costs one Saturday; certification costs years of them. Any process that treats those two as symmetric, one signature each way, has the economics backwards.



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

[AUTHOR-DATA: ledger values for the author's most-drilled path — n valid drills, k passes, $L_W(k, n)$, median and $\widehat{Q}_{0.95}$ of the measured $T_i$ — printed against the declared objective.]

## VIII. What to Show the Auditor

Today the audit artifact of a DR program is typically an architecture diagram of the second region with arrows and the signed tabletop minutes. Both are declarations. There is no evidence, and no regulator with a testing mandate [5, 6] deserves that.

What I give away is the drill ledger. Per claim: recovery time and data loss distributions (distributions are printed first, followed by n and the confidence bound for the distribution) and nothing more; the recovery time recorded under the two clock rules, with the two sets of clock times under the rules; the validity state of each contributing drill and the expiry rule change count that drives the validity state; and the exceptions.

Exceptions are most important. We have a declared only, never exercised in our ledger, which is a vendor system and we can't drill without the vendor for failover, and we requested twice and haven't yet received it. I would prefer to demonstrate an auditor that line rather than decorate it. A ledger with gaps is an evidence ledger, a ledger with no gaps is a tabletop that has been formatted better.

## IX. Related Work

A technique systematised in this article has been used by famous ancestors. Since the mid-2000s, Google has been deliberately failing real systems in their DiRT program, finding the failures, and reporting that the value is in the failures found [2]. Chaos engineering was a discipline of continuous experiments in production [3] that was a generalized fault injection technique. Both address the adjacent problem: fault tolerance of the system? Here, the question is more limited, colder: Is the number in the compliance document accurate, at what confidence, and when? A perturbation is a chaos experiment, a measure is a certification drill, with a certified objective and exercised evidence, which expire. The multi-cloud DR literature provides architectures and discusses the RTO/RPO trade-offs of these architectures [4, 7] and admits that it does lack a standard way to evaluate. I don't have another architecture to present. This is a discipline that is compatible with all recovery architectures in the ladder; the discipline doesn't care how you recover, it only refuses to believe until you do.

## X. Where the Discipline Breaks

Full-path drills are risk of production. The failure-over of real workloads can result in the rehearsing of the incident, and I've seen a drill do that. The mitigation is the same for any risky change: Start with the smallest scope, perform the drill, build the drill up as it becomes boring, but if the team has no drill incident history, then they are probably not doing anything of consequence. Plan it politically before you plan it technically. Disasters are not orderly! A real regional event is typically a gray degradation, with the region impacted but not dead [9] as the provider's own control planes are struggling right when your failover needs them; the public record of large regional events includes a record of failover tooling and consoles degrading with the corresponding impacted workloads [10]. A clean cut will be certified by a clean cut drill. It doesn't constrain any aspect of the gray day, and the ledger should summarise which scenario was exercised. There is no clean solution to the tension between frequency and expiry. Rule-of-three arithmetic desires dozens of suitable drills, expiry makes them expire as the estate ships. In cases of high velocity paths, the honest certified claim may forever be worse than the stated goal, and the proper answer is to declare it in the ledger and continue shipping. The discipline as expressed is based on sound recovery inputs. But ransomware and malicious corruption violate that premise: The backups could be compromised; the replication might have faithfully replicated the damage; and the restore point you need might be the last clean one and not the one that was last recently. That threat model changes what a drill has to demonstrate, and in this series, that's exactly what the isolated backup planes and clean-point condition for the last uncorrupted restore [CROSS-REF: companion on the recoverability-tiered defense] do. Only the boundary will be mentioned here, all the above is proof that you can retrieve what an honest failure can offer. That article is the problem of what's left behind by a hostile.

## XI. Monday Morning

Recover your business-continuity documentation and determine your RTO for your most critical service. Then write the answers to three questions.



*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

**Journal of Information Systems Engineering and Management**
2022, 7(1)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

When was the last time a true workload went through the complete recovery sequence from data restore to traffic cutover, warm-up and failback — under the clock and with the clock under drill rules? The claim is not proven but is stated if the answer is tabletop. Second, is there one number you are encouraged to print, or is there a distribution you are encouraged to print with an n? Thirdly, what has happened to that recovery path since evidence was gathered, and would that recovery path hold up to the expiry test above?

Then take one of them, your full-path or one, and scope it small enough that you will actually run it; then drill that path this quarter. Take it seriously, record it in a ledger, with an expiration date, and have the reference to your first race be embarrassing – ours was against a declared four, and took 11 hours and 20 minutes. The document was not damaged by the drill—this was the drill working. Your RTO is a ‘supposition'. The only question is which Saturday did you pick up—or which day of your year.

## References

[1] J. A. Hanley and A. Lippman-Hand, "If nothing goes wrong, is everything all right? Interpreting zero numerators," JAMA, vol. 249, no. 13, pp. 1743–1745, Apr. 1983, doi: 10.1001/jama.1983.03330370053031.
[2] K. Krishnan, "Weathering the unexpected," ACM Queue, vol. 10, no. 9, p. 30, Sep. 2012. [Online]. Available: https://queue.acm.org/detail.cfm?id=2371516
[3] A. Basiri, N. Behnam, R. de Rooij, L. Hochstein, L. Kosewski, J. Reynolds, and C. Rosenthal, "Chaos engineering," IEEE Software, vol. 33, no. 3, pp. 35–41, May/Jun. 2016, doi: 10.1109/MS.2016.60.
[4] Amazon Web Services, "Disaster recovery options in the cloud," in Disaster Recovery of Workloads on AWS: Recovery in the Cloud, AWS Whitepaper. [Online]. Available: https://docs.aws.amazon.com/whitepapers/latest/disaster-recovery-workloads-on-aws/disaster-recovery-options-in-the-cloud.html
[5] Federal Financial Institutions Examination Council (FFIEC), "Business Continuity Management," FFIEC IT Examination Handbook, Nov. 14, 2019. [Online]. Available: https://ithandbook.ffiec.gov/it-booklets/business-continuity-management
[6] Regulation (EU) 2022/2554 of the European Parliament and of the Council of 14 December 2022 on digital operational resilience for the financial sector (DORA), Official Journal of the European Union, L 333, Dec. 27, 2022, pp. 1–79.
[7] B. Beyer, C. Jones, J. Petoff, and N. R. Murphy, Eds., Site Reliability Engineering: How Google Runs Production Systems. Sebastopol, CA, USA: O'Reilly Media, 2016.
[8] P. Huang, C. Guo, L. Zhou, J. R. Lorch, Y. Dang, M. Chintalapati, and R. Yao, "Gray failure: The Achilles' heel of cloud-scale systems," in Proc. 16th Workshop on Hot Topics in Operating Systems (HotOS '17), Whistler, BC, Canada, May 2017, pp. 150–155, doi: 10.1145/3102980.3103005.
[9] Amazon Web Services, "Summary of the AWS Service Event in the Northern Virginia (US-EAST-1) Region," Dec. 10, 2021. [Online]. Available: https://aws.amazon.com/message/12721
[10] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," Journal of the American Statistical Association, vol. 22, no. 158, pp. 209–212, Jun. 1927, doi: 10.1080/01621459.1927.10502953.
[11] J. E. Kelley, Jr. and M. R. Walker, "Critical-path planning and scheduling," in Papers Presented at the Eastern Joint IRE-AIEE-ACM Computer Conference (IRE-AIEE-ACM '59), Boston, MA, USA, Dec. 1959, pp. 160–173, doi: 10.1145/1460299.1460318.

*Copyright © 2022 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*