Stonetusker Insights · Technical note on high-traffic platforms and release engineering

A Long Peak-Season Freeze Moves Risk into January: Check Your Release Safety in 30 Minutes

Black Friday falls on 27 November 2026. Many teams answer with a change freeze from November into January. This note shows how to check in 30 minutes whether your releases are small and reversible enough to keep shipping through peak.

For: Ecommerce, travel, gaming and streaming teams, and any business with a seasonal peak.

Skip if: Your business has no seasonal peak.

By Subeesh Sivanandan, Founder, Stonetusker Systems · Updated 29 October 2026 · About 4 minutes to read · PDF version

The short version

- Thanksgiving in the United States is 26 November 2026, so Black Friday is 27 November and Cyber Monday is 30 November.[1]
- Adobe expects 275.1 billion dollars of United States online sales in November and December, with Cyber Monday the largest day at 15.1 billion dollars.[2]
- A long freeze stops deployments while development continues. The first release in January then becomes the largest and least rehearsed release of the year.

- Small, reversible releases keep the risk of each change low. Three commands and four questions show whether yours are ready.

27 November 2026
Black Friday; Cyber Monday follows on 30 November[1]

275.1 billion dollars
Adobe forecast for United States online sales, November and December 2026[2]

15.1 billion dollars
Adobe forecast for Cyber Monday, the largest single day[2]

A note to forward to your team

Peak season release check

Black Friday is 27 November and Cyber Monday is 30 November. Before we agree a freeze window, I would like a few numbers.

Please run the checks in this note by [date] and reply with:
1. Changes merged during last year's freeze
2. Changes in the first release in January
3. Measured rollback time of [revenue service], in minutes
4. Irreversible schema changes planned before January

Owner: [name]
Note: https://stonetusker.com/insights/peak-season-release-readiness.html

Why it matters now

The busiest days of the year are close. Adobe expects the five days from Thanksgiving to Cyber Monday to bring 47.5 billion dollars of online sales, 17.3% of the season. Cyber

Monday is forecast at 15.1 billion dollars and Black Friday at 12.9 billion dollars.[2] Travel, gaming and streaming businesses have their own peaks, such as fare sales, game launches and live events, and the same reasoning applies.

A freeze from mid-November into January feels safe. It stops deployments and leaves development running, so finished work piles up and the first release after the freeze carries weeks of change at once. Research from the DevOps Research and Assessment programme shows that speed and stability go together.[3] Small batches reach feedback sooner, and when one fails, the cause is easier to find.[4]

The goal is a release that is small, observable and quick to reverse. With that in place, a short freeze around the busiest days is a business choice backed by data, and the weeks around it stay productive.

Check your release safety in 30 minutes

Step 1. Look at the rollout history

Start with the service that carries revenue at peak, such as checkout, booking or playback. On Kubernetes:

```
kubectl rollout history deployment/checkout --namespace shop
```

Step 2. Time a rollback

In a rehearsal environment, time a rollback from the moment you decide to the moment traffic is healthy again.

```
kubectl rollout undo deployment/checkout --namespace shop
```

Step 3. Count last year's freeze backlog

Inside a clone of the main repository, count the changes merged during last year's freeze.

```
git log --merges --oneline --since=2025-11-15 --until=2026-01-10 |
wc -l
```

If your team squashes pull requests into single commits, run the same command without `--merges`.

Step 4. Answer four questions in writing

- How long does a rollback of the revenue service take, measured rather than estimated?
- Which share of last month's changes reached customers behind a feature flag?
- Which schema changes planned before January are irreversible?
- Is there an error budget for the revenue path, and does it decide when releases stop?

What the result means

What you find	What it means	Typical effort
Rollback measured in minutes, flags on risky changes, only reversible migrations in flight	The team can keep shipping small changes through most of the season	A short freeze around the busiest days only
Rollback never timed, or slow	A bad release on a peak day stays live longer than it should	One rehearsal, then fix the slowest step first
An irreversible schema change planned before January	A failed release cannot simply roll back	Finish it now, postpone it, or split it into phases
A high merge count during last year's freeze	The first January release carried weeks of change at once	Plan small releases for early January and ship behind flags

An error budget turns the decision to stop into a rule. In the example policy from the Google site reliability engineering workbook, a service that exceeded its error budget for the preceding four-week window halts changes other than the most urgent (P0) issues and security fixes until it is back within its objective.[9]

The preparation, step by step

1. **Measure the rollback.** Time a rollback of each revenue service in a rehearsal environment, and fix the slowest step first.
2. **Put risky changes behind flags.** Wrap every change planned for November and December that alters customer-facing behaviour in a feature flag with a named owner.
3. **Finish irreversible migrations early.** Complete or postpone any irreversible schema change, and split the rest into expand, migrate and contract phases.
4. **Automate canary analysis.** Route a small share of traffic to each new version and abort automatically when error rate or latency moves beyond agreed limits.
5. **Agree an error budget policy.** Write down when releases stop and restart based on reliability data, and agree a short freeze only for the busiest days.
6. **Rehearse an incident.** Run a game day on the revenue path with the on-call team before Thanksgiving, including a rollback under load.

The tools are common. Argo Rollouts aborts a rollout and returns traffic to the stable version when analysis fails, and Flagger rolls back when failed checks reach a threshold.[5,6] Feature flags separate releasing a feature from deploying the code.[7] Expand and contract splits a schema change into three phases, so the application can roll back at every step.[8]

What is still open

- The spending figures are forecasts. Adobe measures United States online sales for November and December, and other forecasters use different scopes and periods.
- A short freeze around the busiest days can still be the right decision, for example for staffing or contractual reasons.

- These practices depend on reliable service level indicators, so teams that lack them should start there.

Questions

When is Black Friday in 2026?

Black Friday is 27 November 2026, the day after Thanksgiving in the United States. Cyber Monday is 30 November 2026.

Should we freeze changes over the peak season?

A short freeze around the busiest days can be sensible. A long freeze collects weeks of change into one large release in January, so small, reversible releases are the safer default.

How do feature flags help during peak?

They separate deploying code from releasing behaviour, so a risky feature can be switched off by changing a flag, with no new deployment.

How do we change a database schema safely during peak?

Use expand and contract: add the new structure, move readers and writers, and remove the old structure only when nothing depends on it.

Want a second opinion?

Run the check yourself first. If you would like a second opinion, reply with two counts only: the changes merged during last year's freeze and the changes in the first release in January. No repository names, service names, configuration or command output are needed. Within five working days you receive a short written review with a suggested freeze window, the practices to put in place first and an order of work. It is free and confidential, and a call is optional.

Ask for a second opinion

The button opens an email to subeesh@stonetusker.com with the subject and the numbers to fill in. You can also use the contact page, or reply to the email or LinkedIn message that brought you here.

Next page for your situation

- 20-Minute Delivery Checkup: five numbers that show where your delivery waits the rest of the year.
- Checklist for ultra-scale teams: eight plain questions and the dates that matter.

Sources

1. United States Office of Personnel Management: Federal holidays
2. Adobe: United States holiday shopping season forecast (28 September 2026)
3. DORA: Software delivery performance metrics
4. DORA: Working in small batches
5. Argo Rollouts: Analysis and progressive delivery
6. Flagger: How it works
7. Pete Hodgson on martinfowler.com: Feature Toggles
8. Danilo Sato on martinfowler.com: Parallel Change
9. Google: Site Reliability Engineering workbook, example error budget policy

About the author

Subeesh Sivanandan is the Founder and CEO of Stonetusker Systems. He has worked for more than 27 years in release engineering, DevOps, platform engineering & embedded Linux. His career includes serving as the Principal Engineer & Manager of Release Engineering at Stryker, alongside earlier roles at Nokia, VeriSign, and IP Infusion. LinkedIn profile.

Revision history (recent 3 updates)

Version 2.1, 29 October 2026. Every fact on this page is checked against a minimum of two sources each month.

Version	Date	Change
1.1	16 October 2026	Rewritten as an article for quick readers: introduction, summary of all major changes, what it means and first steps; one figure; commands folded for platform leads; print option and direct email offer
2.0	25 October 2026	Simpler layout and shorter text; the review offer now asks for a few counts only
2.1	29 October 2026	Who-this-is-for box, note to forward moved up, email offer with the numbers listed, next pages; countdown removed