A PAGEANVIL WHITE PAPER

The Evergreen Event URL

Why recurring annual events belong on one permanent page, and how to run one

A PageAnvil white paper | Version 1.0 | September 2026 | Published by Comp Edge Solutions

Applies to churches, nonprofits, schools, festivals, conferences, and any organization that runs the same named event year after year.

Executive summary

Most organizations publish their annual event on a new, year-stamped URL every year: `/oktoberfest-2024`, then `/oktoberfest-2025`, then `/oktoberfest-2026`. It feels tidy and it feels like good record keeping. It is neither. It is the single most expensive routine SEO mistake a small organization makes, and it is expensive precisely because it repeats on a schedule.

Search authority attaches to a **URL**, not to the words on it. A year-stamped URL starts from zero every year, competes with its own predecessors, and leaves behind a graveyard of near-identical pages that drags on the whole domain.

The correct pattern is one permanent, unstamped URL (`/oktoberfest`) that is edited each year, paired with dated recap content that carries the history. Run this way, the event page compounds. By year three it typically owns the event name plus the town name, and the organization stops paying an annual restart tax.

This paper explains the mechanic, gives the decision rule for the small number of cases where per-year pages genuinely are correct, and provides the implementation steps, structured data, recovery procedure for existing graveyards, and the annual maintenance runbook.

Table of contents

1. The problem: the annual page graveyard
2. The mechanic: authority attaches to the URL
3. The reframe: pages are standing answers
4. The decision rule: when to split by year

5. The architecture: hub plus recap
6. The three-mode annual cycle
7. Implementation: Squarespace
8. Implementation: other platforms
9. Structured data for recurring events
10. Recovering an existing graveyard
11. The annual flip runbook
12. Measurement: proving it worked
13. Worked example: a church Oktoberfest
14. Anti-pattern quick reference
15. FAQ

---

1. The problem: the annual page graveyard

Here is the pattern almost every organization falls into, and note that every individual step in it is reasonable.

September 2024. Someone builds a page for the fall festival. They name it `/oktoberfest-2024` because that is what it is. The page goes out on Facebook, the local paper picks it up and links to it, the school newsletter links to it, people share it in group texts. It does its job. Four hundred people show up.

September 2025. The 2024 page is obviously stale. Rather than overwrite what feels like a record of last year, someone duplicates it, updates the dates, and publishes `/oktoberfest-2025`. The 2024 page stays up. It is history.

September 2026. Same again.

By year three the site has three pages that are ninety percent identical, and here is what has actually happened underneath:

- Every link the 2024 page earned points at a page nobody should land on now.
- The 2026 page, the only one that matters, is brand new to Google and has earned nothing.
- Google has to choose which of the three answers “first lutheran oktoberfest,” and it frequently chooses wrong, often surfacing a prior year.
- Someone searching in September 2026 clicks a result, lands on the 2024 page, sees a date that already passed, and leaves. That is a lost attendee, and the bounce is a negative quality signal on top of it.
- The site now carries a cluster of thin, near-duplicate pages, which is a quality drag on the domain as a whole rather than just on those pages.

The organization does this every single year, forever, and every year it pays the same startup cost on the same event. Ten years in, the event page is no stronger than it was in year one. That is the whole problem in one sentence: **work that should compound is being reset annually.**

---

2. The mechanic: authority attaches to the URL

This is the technical fact that explains everything else in this paper, and it is the part that is genuinely not widely understood outside professional SEO.

Search engines do not accumulate trust in *content*. They accumulate it in an *address*.

Everything that makes a page rank well over time is bound to its URL:

SIGNAL	WHAT IT IS	BOUND TO
Backlinks	Other sites linking in	The URL
Internal link equity	Your own pages pointing at it	The URL
Historical CTR	How often people click it in results	The URL
Crawl priority	How often Google bothers to revisit	The URL
Age and stability	How long this address has answered this query	The URL
Query association	The set of searches this address is known for	The URL

None of that is bound to the words. This is why you can substantially rewrite the body copy on an established page and hold your rankings, and why moving the identical copy to a new address tanks them.

What each option actually does

Editing in place. The URL keeps every signal in the table above. Google recrawls, sees updated content on an address it already trusts, and updates its index. The trust carries forward. Cost: zero.

Cloning to a new year-stamped URL. The new address has none of those signals. It is a stranger. Meanwhile the old address keeps all of them and points at content you now consider obsolete. You have separated the authority from the page you want people to reach. Cost: the entire compounding history of the event.

Cloning and redirecting. Better, but pointless churn. You are creating a new address, then spending a redirect to send its authority somewhere, then doing it again next year. Just edit the page.

The precedent

This is not a theory or a niche opinion. It is why every publisher that competes on annually-refreshed content works this way. Review sites keep `/best-laptops` and rewrite it every year rather than shipping `/best-laptops-2026`. Tax and finance sites keep `/roth-ira-limits`. Sports sites keep `/nfl-power-rankings`. These are organizations with dedicated SEO teams and direct revenue

exposure to getting it right. They all reached the same answer, and they reached it because the year-stamped alternative loses measurably and repeatedly.

---

3. The reframe: pages are standing answers

The graveyard happens because of a category error. People think of a web page as **a document about a thing that happened**, like a flyer or a program you file away afterward. Under that model, keeping 2024's flyer and making a new one for 2025 is obviously correct. You would not white-out last year's flyer.

Replace that model with this one:

> **A URL is a standing answer to a standing question.**

`/oktoberfest` is not a document about the 2026 festival. It is the permanent answer to the question *"What is First Lutheran's Oktoberfest and when is it?"*

That question gets asked every year. It has the same shape every year. Only the values inside the answer change. A question that recurs deserves an address that persists.

Once you hold that model, the correct behavior becomes obvious in every case:

- Someone searches for the event. They want the *next* one. One address, always current.
- Someone links to the event. They are linking to the idea of it, not to one instance. One address, so the link stays useful for a decade.
- Someone wants to see photos from a past year. That is a **different question**, and different questions get different addresses. See section 5.

The test to apply to any page: *does the question this page answers recur?* If yes, the page is a standing answer and gets a permanent unstamped URL. If the question is genuinely one-time, date it.

---

4. The decision rule: when to split by year

Per-year pages are not always wrong. They are wrong the overwhelming majority of the time for the organizations this paper addresses, but the rule matters more than the default, because it lets you decide correctly on your own.

The rule

> Create per-year pages only when **both** conditions are true:
>
> 1. **Each year's content is substantially different**, not just re-dated. Not a new date and a new photo. A materially different lineup, agenda, speakers, results, or program that someone would want to read on its own.
> 2. **People actually search with the year attached**, in enough volume to be worth a page.
>
> If either is false, use one permanent URL.

The practical test

Ask: **would a real human being type the year into the search box?**

Not "could they." Would they. For nearly every church festival, school fundraiser, community 5K, and nonprofit gala, the honest answer is no. People search "first lutheran oktoberfest" or "oktoberfest springfield ohio." They do not search "first lutheran oktoberfest 2025," because they want the one that is coming, not the one that is gone, and they assume your site will show them the current one.

Examples

EVENT	YEAR IN SEARCH?	CONTENT DIFFERS?	VERDICT
Church Oktoberfest	No	No	**One URL**
School fall carnival	No	No	**One URL**
Annual charity 5K	No	Mostly no	**One URL**
Vacation Bible School	No	Theme changes, format does not	**One URL**
Nonprofit gala	No	No	**One URL**
CES, SXSW, Comic-Con	Yes, heavily	Yes, entirely	Hub plus year archives
The Masters, World Cup	Yes, heavily	Yes, results differ	Hub plus year archives
Academic conference with proceedings	Yes	Yes, distinct papers	Hub plus year archives

Notice what separates the two groups. It is not size or importance. It is whether **each instance is its own subject with its own audience**. CES 2026 has a different exhibitor list, different product launches, and thousands of people researching that specific year's news. Your fall festival has a bratwurst stand and a bounce house, same as last year.

Even in the split case, you still keep the hub

Note that the right column says "hub plus year archives," never "clone and abandon." Even conference sites that legitimately need per-year pages maintain a permanent unstamped hub that always ranks for the bare event name, always points at the next occurrence, and links down to the archives. The archives are supporting cast. The hub is the asset.

So the permanent URL is not an alternative to year pages. It is mandatory in both models. The only question is whether you *additionally* need archives.

---

5. The architecture: hub plus recap

The reason people resist the single-URL model is a real concern hiding inside a wrong solution. The concern is legitimate: *we do not want to erase our history.* The solution they reach for, keeping the old landing page, is the wrong tool for it.

The fix is to notice that you have **two different jobs** that have been jammed into one page, and to give each one the content type it deserves.

	FORWARD-LOOKING	BACKWARD-LOOKING
Answers	"Should I come? When? What is it?"	"What was it like?"
Audience	Prospective attendees, searchers, newcomers	Attendees, members, community, media
Lifespan	Permanent, always current	Fixed in time, never updated
Content type	Standing page	Blog post or gallery
URL	`/oktoberfest`	`/blog/oktoberfest-2026-recap`
Schema	`Event`	`BlogPosting` or `Article`
Count	Exactly one, forever	One per year, accumulating

The URL map

```
/oktoberfest                          <- permanent hub, edited annually
/blog/oktoberfest-2026-recap          <- written the week after, never touched again
/blog/oktoberfest-2025-recap
/blog/oktoberfest-2024-recap
/oktoberfest-2025          -> 301 -> /oktoberfest   (legacy, redirected)
/oktoberfest-2024          -> 301 -> /oktoberfest   (legacy, redirected)
```

The hub links down to the recaps. Every recap links back up to the hub. That internal linking is not decoration. It is how the recaps pass their own accumulated authority up to the page that needs it, which means your archive is actively *strengthening* the hub rather than competing with it.

Why this preserves history better, not worse

The objection to single-URL is "we lose our history." The reality is the opposite. Compare what each model actually preserves:

A stale landing page still says "Join us Saturday, October 3rd! Free admission! Bring the family!" in the present tense, nine months after the fact. It is not an archive. It is a broken page written for an audience that no longer exists, and anyone who lands on it is confused.

A recap post says “Four hundred of you joined us on October 3rd. Here are the photos. Thank you to the sixty volunteers who made it happen.” It was written to be read afterward. It is correct on the day it publishes and it is still correct in 2036. It has the photos, the numbers, the names, the story.

The recap is the real archive. The stale landing page was never an archive. It was just an abandoned page that felt like one.

There is a bonus: recap posts rank well for their own long-tail queries (“first lutheran oktoberfest photos,” “oktoberfest springfield 2026 pictures”) with essentially no competition, and they give you a legitimate freshness signal each year without splitting authority.

---

6. The three-mode annual cycle

A permanent URL raises a fair question: what does the page say in March, when the last event is over and the next one is not scheduled?

It runs in three modes across the year. Same URL throughout.

Mode 1: Recap (roughly the week after, through year end)

The hero flips to gratitude and proof.

- Headline: “Thank you for a great Oktoberfest 2026”
- Attendance number, a strong photo strip, link to the full recap post
- A forward hook: “Oktoberfest 2027 details coming this summer”
- Email capture: “Get the date as soon as it is set”

This mode is doing real work. It is the social proof that converts next year’s first-time visitor, and the email capture converts this year’s traffic spike into next year’s attendance instead of letting it evaporate.

Mode 2: Holding (roughly January through June)

The quiet season. The page still has a job.

- Headline: “Oktoberfest at First Lutheran”
- What it is, in the present tense as an ongoing tradition
- Best photos from last year
- The expectation: “Held annually on the first Saturday in October”
- Email capture, prominent

Critically, the page still ranks all year in this mode. Someone who moves to town in February and searches for local fall events finds you, and you capture them. A year-stamped model has literally nothing to show that person.

Mode 3: Ramp (roughly July through the event)

Full event mode.

- Headline: “Oktoberfest 2026” with date and time
- Schedule, menu, activities, cost, parking, rain plan
- RSVP or volunteer signup
- `Event` structured data live and accurate
- Announcement bar switched on about three weeks out

What this costs you

Here is the part that makes it sustainable. Look at what actually changes:

SECTION	CHANGES ANNUALLY?
Hero headline and date	**Yes**
Event details block (times, schedule, menu)	**Yes**
Photo strip	**Yes**
SEO title and description	**Yes**, one number
Event JSON-LD dates	**Yes**, two fields
What Oktoberfest is	No
Location and parking	No
What about my kids	No
Accessibility info	No
FAQ	Rarely
Contact and directions	No

The bottom two thirds of the page never moves. **The annual flip is a twenty minute edit, not a rebuild.** That is not a happy accident, it is the direct payoff of separating the standing answer from the dated instance.

---

7. Implementation: Squarespace

Squarespace-specific steps. Menu paths shift between versions, so treat these as close guidance rather than exact strings.

7.1 Do not use the Events collection for this

Squarespace’s built-in Events page type is the obvious-looking choice and it is the wrong one for an event you want indexed in perpetuity. Events automatically move into a “Past Events” state once the date passes, and some templates hide past events from view entirely. A page that disappears every November cannot accumulate anything.

Use the Events collection for the recurring internal calendar. Use a regular page for the event you want to rank.

If you want the event to appear in the site calendar as well, add a calendar entry whose description links to the permanent landing page. The landing page stays the destination.

7.2 Build the page

1. **Pages panel** → find the page you are modeling from → hover → `...` → **Duplicate Page**. Duplicating an existing well-built page inherits your section themes, spacing, and type treatment, which saves both time and visual drift.
2. Choose your model page by **structure**, not topic. For an event page you want a page that already handles practical logistics and questions. A "Plan Your Visit" page is usually a near-perfect skeleton. A form-driven page like Contact or Giving is not, because it is built around a conversion form rather than an information hierarchy.
3. **Page Settings → General → URL Slug** → set to `/oktoberfest`. No year. Ever. This is the single most important line in this document.

7.3 SEO panel

Page Settings → SEO

- **SEO Title:** 50 to 60 characters, with the year and the town. `Oktoberfest 2026 | First Lutheran Church, Springfield OH`
- **SEO Description:** 150 to 160 characters, with the date, and written as an invitation rather than a summary.
- **Social Image:** set this explicitly. Squarespace defaults to the first image on the page, which is usually cropped badly for Open Graph and looks careless when the page is shared to Facebook, which for a community event is a primary distribution channel.

7.4 Event structured data

Requires a Business plan or higher, which is also what Code Blocks require.

Page Settings → Advanced → Page Header Code Injection

See section 9 for the full markup and annual maintenance.

7.5 Announcement bar

Turn it on about three weeks out, pointing at the permanent URL. Turn it off the day after. Found under Website Tools or Marketing depending on your version.

7.6 Legacy redirects

Settings → Advanced → URL Mappings (older accounts: **Settings → Developer Tools → URL Mappings**). See section 10.

---

8. Implementation: other platforms

The principle is platform-independent. Only the redirect syntax changes.

WordPress

- Build as a **Page**, not a Post. Posts are date-ordered and carry `BlogPosting` semantics.
- Set the permalink to `/oktoberfest` and never let WordPress append `-2` or a year.
- WordPress will silently create `oktoberfest-2` if a trashed page holds the slug. Empty the trash or edit the old page's slug to free it.
- Structured data: a plugin such as Yoast or Rank Math, or a raw JSON-LD block in the page.
- Redirects: the Redirection plugin, or Rank Math's redirect manager.

```
# .htaccess, if handling redirects at the server
Redirect 301 /oktoberfest-2024 /oktoberfest
Redirect 301 /oktoberfest-2025 /oktoberfest
```

Static sites and hand-built HTML

```
# nginx
location = /oktoberfest-2025 { return 301 /oktoberfest; }
```

Netlify `_redirects`:

```
/oktoberfest-2025    /oktoberfest    301
/oktoberfest-2024    /oktoberfest    301
```

Wix, Weebly, and similar

All have a URL redirect manager under SEO or site settings. The rules in this paper apply unchanged. The only platform-specific caution is the same as Squarespace: avoid any built-in "events" module that auto-archives past entries.

---

9. Structured data for recurring events

Structured data is what produces the Google event rich result: the date badge, the location, the enhanced listing. For a local community event this is often the single highest-leverage technical addition, because it makes your listing visually larger than everything around it.

The markup

Place in the page header code injection.

```
<script type="application/ld+json">
{
  "@context": "https://schema.org",
  "@type": "Event",
  "name": "First Lutheran Oktoberfest 2026",
  "description":
        "Free community Oktoberfest with German food, live polka, kids' games, and a bounce
        house. Everyone welcome.",
  "startDate": "2026-10-03T11:00:00-04:00",
  "endDate": "2026-10-03T16:00:00-04:00",
  "eventStatus": "https://schema.org/EventScheduled",
  "eventAttendanceMode": "https://schema.org/OfflineEventAttendanceMode",
  "location": {
    "@type": "Place",
    "name": "First Lutheran Church",
    "address": {
      "@type": "PostalAddress",
      "streetAddress": "123 Main Street",
      "addressLocality": "Springfield",
      "addressRegion": "OH",
      "postalCode": "45501",
      "addressCountry": "US"
    }
  },
  "image": [
    "https://example.org/s/oktoberfest-1x1.jpg",
    "https://example.org/s/oktoberfest-4x3.jpg",
    "https://example.org/s/oktoberfest-16x9.jpg"
  ],
  "offers": {
    "@type": "Offer",
    "price": "0",
    "priceCurrency": "USD",
    "availability": "https://schema.org/InStock",
    "url": "https://example.org/oktoberfest",
    "validFrom": "2026-07-01T00:00:00-04:00"
  },
  "isAccessibleForFree": true,
  "organizer": {
    "@type": "Organization",
    "name": "First Lutheran Church",
    "url": "https://example.org"
  }
}
</script>
```

Field notes

- **`name`** should carry the year even though the URL does not. The URL is the permanent address; the markup describes the specific upcoming occurrence.
- **`startDate`** / **`endDate`** must include the timezone offset. Note that offset changes with daylight saving time: `-04:00` for a US Eastern event in early October, `-05:00` after the November change. Getting this wrong shifts your displayed time by an hour.

- `image` should ideally supply 1×1, 4×3, and 16×9 crops. Google picks by surface. One image works, three perform better.
- `offers` with `price: "0"` is worth including for free events alongside `isAccessibleForFree`. Google reads the offer block and will surface “Free.”
- `location` is required. An event without a location is not eligible for the rich result at all.
- `performer` is worth adding if you have a named band or speaker.

The three fields you touch annually

```
name        "Oktoberfest 2026"  ->  "Oktoberfest 2027"
startDate   2026-10-03T11:00    ->  2027-10-02T11:00
endDate     2026-10-03T16:00    ->  2027-10-02T16:00
```

Plus `offers.validFrom` if you use it. That is the entire annual schema maintenance.

What happens after the event passes

Once the date is in the past, Google simply stops showing the rich result. There is no penalty and no urgency. You are not doing anything wrong by having a past date sit in the markup for a few months.

That said, best practice is to update as soon as the next date is confirmed. If the gap will be long and you want to be tidy, you can remove the block entirely during holding mode and restore it when the new date is set. Either is acceptable. What is **not** acceptable is marking a past event as `EventScheduled` for a date you know will never occur while actively promoting it, which is a misrepresentation.

On `EventSeries`

Schema.org does offer an `EventSeries` type, and it is conceptually the perfect match for an annual event. In practice, Google’s event rich results are driven by `Event`, and `EventSeries` support is inconsistent across surfaces. The pragmatic recommendation is to mark up the **next occurrence** as a plain `Event` and not to over-engineer this. Revisit if Google’s documentation changes.

Do not put Event schema on the recap posts

Recap posts are `BlogPosting` or `Article`. Putting `Event` markup on a recap creates a second competing event entity for the same event and can confuse which page Google surfaces. One event, one `Event` block, on the hub.

Validate before you walk away

Run the finished page through Google’s Rich Results Test and the Schema Markup Validator. A typo in JSON-LD fails silently. The page looks perfect and the rich result simply never appears, and you will not find out for months.

---

10. Recovering an existing graveyard

If you already have year-stamped pages, this section is the most immediately valuable one in the paper. This is not cleanup. It is a **recovery**.

Every abandoned year page still holds the links and history it earned. Those signals are not gone, they are stranded. A 301 redirect is the instruction that hands them to the page you actually want ranking. You are consolidating years of accumulated authority into a single address rather than leaving it scattered across pages nobody visits.

Procedure

Step 1. Inventory. Find every year-stamped page for the event. Sources:

- The CMS pages list, including unlinked and disabled pages
- Google Search Console → Pages, filtered by the event name
- `site:yourdomain.com oktoberfest` in Google
- Your XML sitemap
- The Wayback Machine if pages were deleted rather than unpublished

Step 2. Pick the survivor. Use the clean unstamped slug: `/oktoberfest`. If that URL does not exist yet, take the strongest existing year page, change its slug to the unstamped version, and let the platform redirect the old slug. Squarespace and WordPress both create that redirect automatically on a slug change, which means you inherit that page's history rather than starting fresh. This is meaningfully better than building a brand new page.

Step 3. Redirect everything else into it.

Squarespace, in Settings → Advanced → URL Mappings:

```
/oktoberfest-2025 -> /oktoberfest 301
/oktoberfest-2024 -> /oktoberfest 301
/oktoberfest-2023 -> /oktoberfest 301
/events/oktoberfest-2025 -> /oktoberfest 301
```

One rule per line. The `301` is required; without it Squarespace defaults to a 302, which is a temporary redirect and passes no authority. **A 302 here does nothing for you.** This is the most common error in this procedure.

Step 4. Unpublish, do not delete. Once redirects are live, unpublish the old pages rather than deleting them. If a redirect turns out to be wrong you still have the content. Do not leave them published, because a published duplicate competes.

Step 5. Fix internal links. Search the site for links to the old URLs and repoint them at the permanent one. Redirects work, but a direct link is cleaner, faster, and passes authority without the small loss a redirect chain can introduce. Check navigation, footers, old blog posts, and any embedded buttons.

Step 6. Update external listings. This is the step everyone skips and it is where much of the value is. Repoint anything you control:

- Google Business Profile posts and event links
- Facebook page and past event links
- Community calendars and chamber of commerce listings
- The local paper's event listing if they will update it
- Eventbrite, Nextdoor, denominational directories
- Email signatures and printed QR codes

Step 7. Resubmit. Submit the permanent URL in Search Console and request indexing. Make sure your sitemap contains the permanent URL and not the old ones.

What to expect

Redirect equity transfer is not instant. Google must recrawl the old URLs to see the 301. Typical timeline:

- **1 to 3 weeks:** old URLs start dropping out of results
- **4 to 8 weeks:** the permanent URL begins showing the consolidated impressions
- **3 to 6 months:** full consolidation, and this is where you see the gain

Do the redirects **well before** the next event, not during the promotion push. Ideally in the off season.

---

11. The annual flip runbook

Print this. Hand it to staff. It should not require the person who built the site.

Immediately after the event (within one week)

- [ ] Write and publish the recap post at `/blog/[event]-[year]-recap`
- [ ] Include photos, attendance number, volunteer thanks
- [ ] Link from the recap up to the permanent event page
- [ ] Switch the event page hero to **Recap mode**
- [ ] Add the photo strip and link down to the recap post
- [ ] Add "Next year's dates coming soon" plus email capture

- [ ] Turn the announcement bar off

Off season (January)

- [ ] Switch the hero to **Holding mode**
- [ ] Confirm the page still reads correctly with no upcoming date
- [ ] Verify the email capture works and the list is going somewhere real
- [ ] Confirm evergreen sections (parking, accessibility, kids) are still accurate

Once next year's date is confirmed

- [ ] Hero H1 to the new year and date
- [ ] Event details block: date, times, schedule, menu, cost, rain plan
- [ ] Page Settings → SEO: title and description updated to the new year
- [ ] Page Settings → Social Image if the artwork changed
- [ ] Page header JSON-LD: `name`, `startDate`, `endDate`, `offers.validFrom`
- [ ] Run the Rich Results Test and confirm it still validates
- [ ] Swap in current photos, archive the old strip to the recap post
- [ ] RSVP or volunteer signup links pointed at the current form
- [ ] **Confirm the URL slug is unchanged.** No year. Ever.
- [ ] Email the list from last year's capture

Three weeks out

- [ ]

Announcement bar on, pointing at the permanent URL

- [ ] Google Business Profile post linking to the permanent URL
- [ ] Facebook event created, with the permanent URL in the description
- [ ] Community calendar submissions using the permanent URL
- [ ] Request indexing in Search Console

Never

- [ ] ~~Create a new page for this year~~
- [ ] ~~Add the year to the URL slug~~
- [ ] ~~Delete last year's recap post~~
- [ ] ~~Leave a past date in `EventScheduled` while actively promoting~~

12. Measurement: proving it worked

The whole argument for this pattern is that it compounds, so measure the compounding. Track in Google Search Console, filtered to the permanent URL.

The primary metric: year over year at the same point in the cycle

Compare the six weeks before the event each year. Under the correct pattern this should **step up every year**. Under the graveyard pattern it is flat or noisy, because you restart annually.

METRIC	WHERE	WHAT GOOD LOOKS LIKE
Impressions, 6 weeks pre-event	Search Console, URL filter	Higher every year
Clicks, same window	Search Console	Higher every year
Average position, bare event name	Search Console, query filter	Climbing toward 1
Distinct ranking queries	Search Console	Set widens each year
Referring domains to the URL	Any backlink tool	Accumulates, never resets
Rich result impressions	Search Console → Enhancements → Events	Present during ramp

Secondary signals

- **Query set widening** is the clearest evidence the pattern is working. Year one you rank for your event name. Year three you also rank for "fall festival near me," "german food festival springfield," "free family events october." That breadth is only possible on an address with accumulated history.
- **Direct and returning traffic in the off season.** A permanent URL gets traffic in March. A year-stamped one does not exist in March.
- **Email list growth** from holding mode, which is the mechanism that converts search traffic into attendance.

After a graveyard recovery

Watch the old URLs' impressions fall and the permanent URL's rise. The sum should recover to at least the prior total within about eight weeks, then exceed it, because consolidated authority on one page outranks the same authority split across four.

---

13. Worked example: a church Oktoberfest

Concrete application, for reference.

URL structure

```
/oktoberfest                        permanent hub
/blog/oktoberfest-2026-recap        published the week after
```

Page structure

Built by duplicating the existing Plan Your Visit page, whose section skeleton already matches what an event page needs:

PLAN YOUR VISIT SECTION	BECOMES
Service Times	Date, time, rain plan
What to Expect	What the day looks like
What About My Kids?	Kids' activities and bounce house
Location & Parking	Unchanged, already correct
FAQs for Visitors	Event FAQ
Let Us Know You're Coming	RSVP, volunteer signup, online signup link

This is a better base than a Prayer or Giving page, because those are conversion-funnel pages built around a single form. An event page is an information hierarchy that resolves to a form at the end, which is exactly the Plan Your Visit shape.

FAQ section

The FAQ is not filler. It is the primary Generative Engine Optimization surface, because AI search assistants extract direct question-and-answer pairs. Write the headings as the literal questions people ask, and answer in the first sentence before elaborating.

- Is Oktoberfest free?
- Do I need to be a member to come?
- Where do I park?
- Is there food for kids? What if my child has allergies?
- Is it accessible for wheelchairs and walkers?
- What happens if it rains?
- Is there alcohol? (a congregation with a clear position on alcohol should answer this plainly rather than let people guess)
- Can I volunteer?

Annual cost

Twenty minutes in July to flip the page, plus one recap post in October. That is the entire ongoing maintenance, and the page gets stronger every year.

---

14. Anti-pattern quick reference

ANTI-PATTERN	WHY IT HURTS	DO INSTEAD
`/event-2026` slug	Zero authority every year	`/event`, year in H1 and title only
Keeping old year pages published	Duplicates compete; stale pages rank	301 into the permanent URL
Deleting old year pages outright	Throws away the equity a 301 would transfer	Redirect first, then unpublish
302 instead of 301	Passes no authority	Always 301 for permanent moves
Blog post as the landing page	Date-sorted, gets buried, wrong schema	Regular page
Platform “Events” module	Auto-archives, may hide after the date	Regular page; calendar entry links to it
Cloning forward each year	Creates the graveyard	Edit in place
Year in the URL “for the archive”	The archive belongs in recaps	Recap post carries the history
Leaving the page stale post-event	Broken experience, bad quality signal	Flip to recap mode within a week
Event schema on recap posts	Competing event entities	`BlogPosting` on recaps

ANTI-PATTERN	WHY IT HURTS	DO INSTEAD
Not updating external listings	Authority keeps flowing to dead URLs	Repoint everything you control
Default social image	Bad crops on the channel that matters most	Set it explicitly

15. FAQ

Will Google think the page is stale if it says "2026" in March?

No. Google evaluates freshness by content change and crawl signals, not by whether a number in your headline matches the calendar. And under the three-mode cycle the page is not sitting there saying 2026 in March anyway. It is in holding mode. The year only appears during recap and ramp.

Our 2024 page has a lot of Facebook and news links. Do we lose those?

That is precisely the case where this matters most, and no, you do not lose them, as long as you 301. The redirect passes that authority to the permanent URL. Left alone, those links point at a dead end forever. This is the single best argument for doing the recovery in section 10.

Should the recap post have Event schema?

No. `BlogPosting` or `Article`. See section 9.

What if the event's name changes, say Oktoberfest becomes Fall Fest?

Then the standing question changed, and a new URL is defensible. Create `/fall-fest`, 301 `/oktoberfest` into it, and update the H1 and all internal links. Do this once, for a real rename, not annually.

What if we skip a year?

The page persists in holding mode. Add one line stating the event is not being held this year and when it returns. You keep the authority. A year-stamped model would have you delete or abandon a page, and you would restart from zero when the event comes back.

We have two events that would both want the same slug.

They do not. `/oktoberfest` and `/fall-festival` are different slugs. If two events genuinely share a name, disambiguate by the thing that actually differs and will keep differing, such as location or audience, never by year.

Does this apply to weekly things like sermons?

Different pattern. Individual sermons are genuinely one-time content with distinct subjects, so they get dated or slugged individually and live under a permanent hub like `/messages`. The rule from section 4 gives you this automatically: each sermon's question does not recur, so it is not a standing answer. The hub is.

How long before we see the benefit?

The redirect recovery shows up in 4 to 8 weeks. The compounding benefit of the permanent URL is structural rather than immediate: you see it as the absence of an annual reset. The clearest read is the year-over-year comparison in section 12 at the second and third year.

Is any of this risky?

Editing in place is not risky. The redirect consolidation carries the normal, small risk of any redirect work, which is why section 10 says to do it in the off season and to unpublish rather than delete. Nothing in this paper requires touching anything outside the event pages.

Summary

1. Search authority lives on a **URL**, not on content.
2. A recurring event asks a recurring question, so it deserves a **permanent unstamped URL**.
3. History belongs in **dated recap posts**, which archive better than an abandoned landing page ever did.
4. Split into per-year pages only when each year is its own subject **and** people search with the year. For most organizations, never. And even then you keep the hub.
5. Existing year-stamped pages should be **301'd into the permanent URL**. That is a recovery of stranded authority, not housekeeping.
6. The annual maintenance is a **twenty minute edit** and a recap post.

The organizations that get this right stop paying an annual restart tax and start compounding. The gap between the two approaches is small in year one and enormous by year five.

About PageAnvil

PageAnvil, from Comp Edge Solutions, turns long-form content into search-optimized landing pages. The standing-page model in this paper is the one PageAnvil is built around: one permanent address per subject, refreshed rather than replaced, with the history carried in dated supporting content. Learn more at pageanvil.com.

LayerLytix, the Comp Edge Solutions analytics dashboard, reports the year-over-year search comparisons described in section 12, so the compounding this paper promises can be watched rather than assumed. Learn more at layerlytix.com.

Comp Edge Solutions builds PageAnvil and LayerLytix and works directly with churches, nonprofits, and small organizations on search and content strategy. Learn more at compedgesolutions.us.

A PageAnvil white paper, published by Comp Edge Solutions. First prepared for a church client engagement and generalized for reuse. Version 1.0, September 2026.