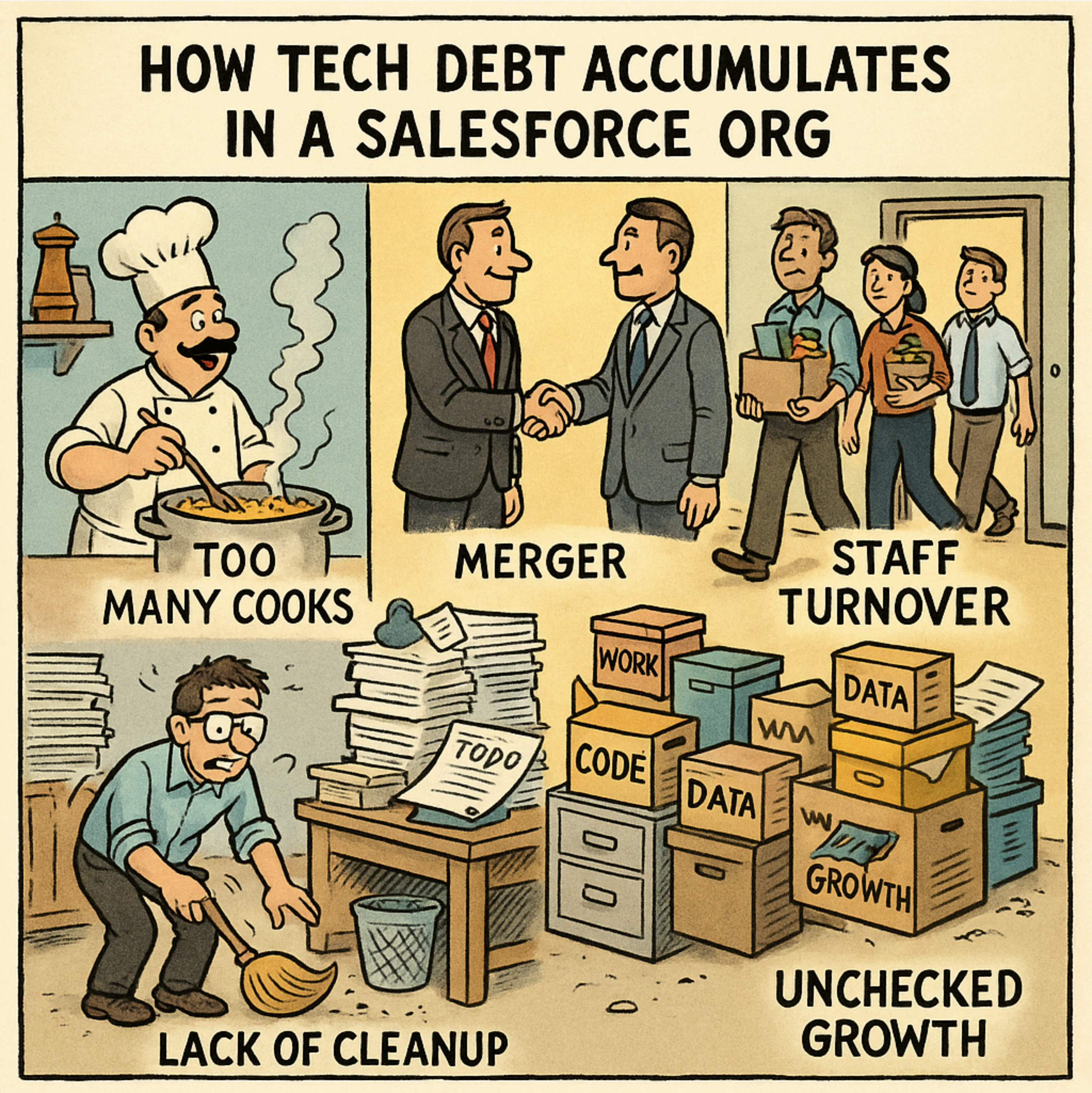
10 Types of Tech Debt in an old Salesforce Org



Salesforce orgs can slowly accumulate a lot of technical debt. How does it build up without anyone realizing?

Too Many Cooks in the Kitchen
Salesforce admins and developers might add fields, workflows, or validation rules without taking a step back to think about the long-term impact. At the moment, it makes sense: “Let’s add a field for this,” or “A new record type here will save us time.” But a few years later, it became excessive (500 custom fields on the Account object, 40 page layouts and 15 Lightning pages) and no one is keeping track.

Mergers and Acquisitions
When companies acquire others, it often means inheriting a new Salesforce org with a different set of processes and configurations. All that is merged into one org without cleaning up the old data, fields, and automations. That’s how you end up with things like duplicate workflows and unnecessary custom fields. The result is a Frankenstein of an org that works but not as efficiently as it could.

Staff Turnover
People leave. As employees (especially the ones who designed the org) leave, their knowledge of why certain things were done also leaves. New staff step in, not knowing why 60 validation rules exist, and start adding their own. The old ones aren't cleaned up. Over time, we end up with something way more complex than it needs to be. Fields go unreferenced, workflows pile up, and no one knows what’s actually being used anymore.

Lack of Cleanup
It’s out of sight and out of mind. No is assigned to the big task of spring cleaning the org. Over the years, features and technology evolve. Legacy components (Visualforce pages or Apex classes) often remain. At one point they were critical, but now they just hang around, unused, unoptimized, and taking up space.

Unchecked Growth
As business needs change, Salesforce orgs expand. And when features grow, they often expand in all directions. Before you notice, you're dealing with things like overstuffed page layouts, excessive custom components, and fields with no meaningful data. It happens gradually but eventually, it becomes conspicuous.

In the end, tech debt accumulates in orgs because of constant change: adding new features, merging companies, or staff turnover. Without ongoing maintenance, it can get bogged down with things that are no longer necessary. But with the right strategy, this mountain of tech debt can be tackled one field, record type, and workflow at a time.

The next 10 pages show common types of tech debt with suggestions on how to start assessing them.

Compiled by Fernando Fenandez on 5/9/25

Complex Object Structures

Objects with excessive page layouts, record types, and custom fields (e.g., Account, Case, Opportunity, Lead, Knowledge).

Example: Account with 30 page layouts, 25 Lightning record pages, 500 custom fields, and 30 relationship fields.

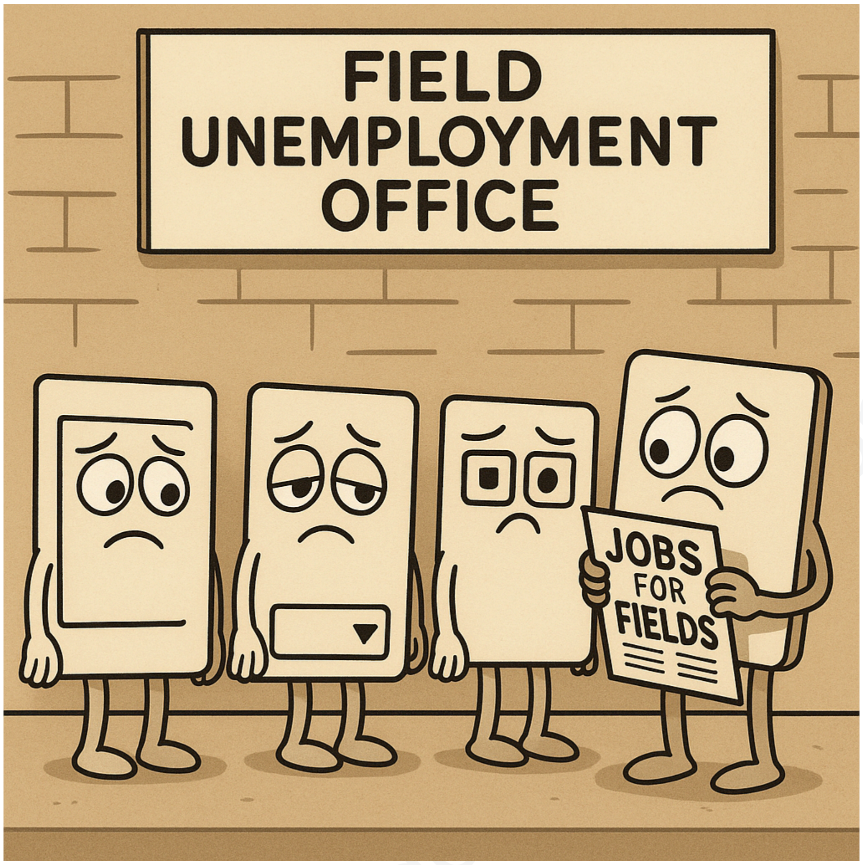
How to assess:

- Install Org Check package https://salesforcelabs.github.io/OrgCheck/
- Run Setup / Optimizer / Open Optimizer
- Install SF Explorer browser extension https://info.sf-explorer.com/
- Run query(*): objects with large number of fields:
  ```
  SELECT COUNT(DeveloperName), TableEnumOrId FROM CustomField GROUP BY
  TableEnumOrId ORDER BY COUNT(DeveloperName) DESC
  ```

(*) query can be done via Salesforce Inspector Reloaded, Workbench, Developer Console or SF CLI

Compiled by Fernando Fenandez on 5/9/25

Unpopulated or Inconsistent Data Fields



Large portions of custom fields in objects remain unpopulated across records, leading to clutter and inefficiency.

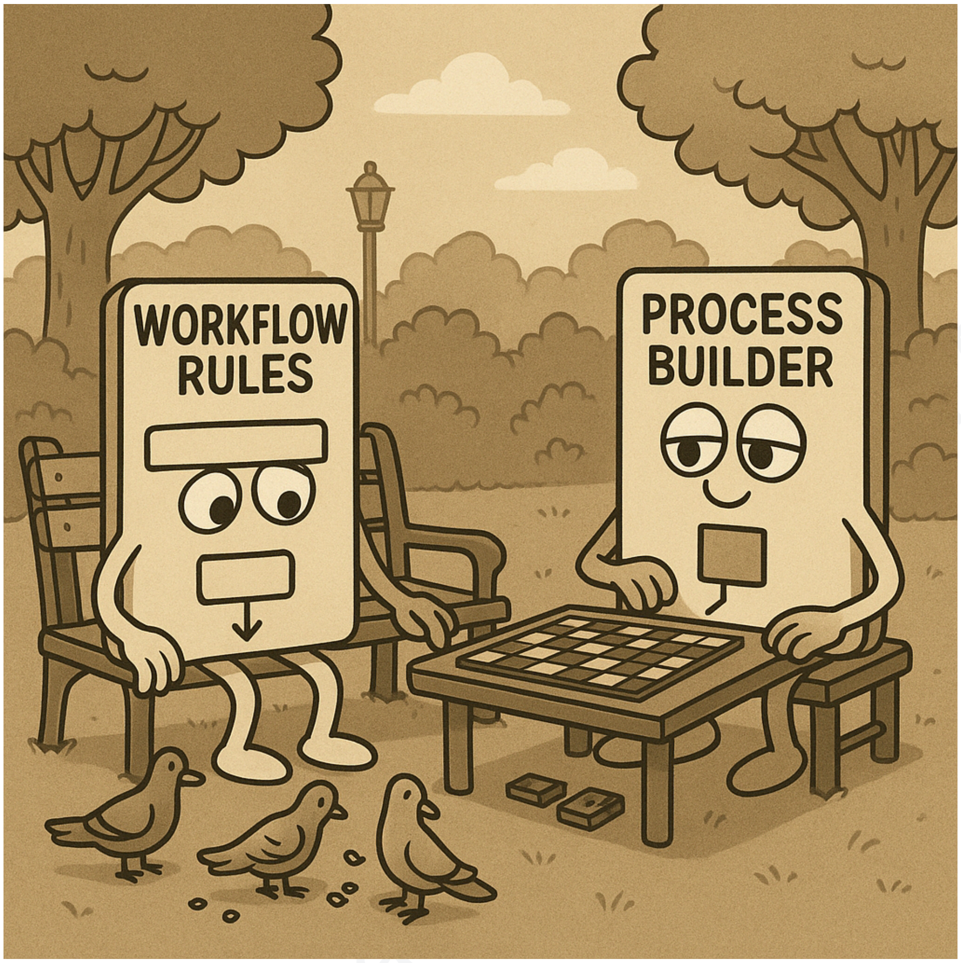
Example: Only 100 fields on the Account object have been populated, while 400 fields are blank in more than 25% of records.

How to assess:

- Field Trip, Field Footprint or similar packages
- Install Cuneiform Field & Data Management: https://appexchange.salesforce.com/appxListingDetail?listingId=a0N4V00000IPSsZUAX
- Install FieldSpy by Sonar: https://appexchange.salesforce.com/appxListingDetail?listingId=a0N4V00000GuIM2UAN
- Hubbl.com diagnostics
- Run query via Workbench, Developer Console or SF CLI - count different values in all records for a given field:
  ```
  SELECT FieldName__c, COUNT(Id) FROM Account WHERE CreatedDate >= LAST_YEAR
  GROUP BY FieldName__c LIMIT 10
  ```
- Run an Impact Analysis using https://happysoup.io/
- Before deleting fields that appear unused, check this list at https://www.salesforceben.com/8-reasons-why-you-shouldnt-delete-an-empty-field/

Compiled by Fernando Fenandez on 5/9/25

Process Builder and Workflow Rules still in use



These 2 features are supposed to be retired and need to be migrated to Flow Builder before they are no longer supported in 2026.

Workflow rules and process builder across several objects, some of which are inactive, outdated, lack descriptions, or are redundant.

Example: 40 workflow rules remain and need to be migrated to Flow Builder.

How to assess:

- Install Org Check package https://salesforcelabs.github.io/OrgCheck/
- Run Setup / Optimizer / Open Optimizer

Compiled by Fernandez 5/9/25

Outdated or Stale Code

Unreferenced or outdated code, such as Visualforce pages, components, and unutilized Apex classes.

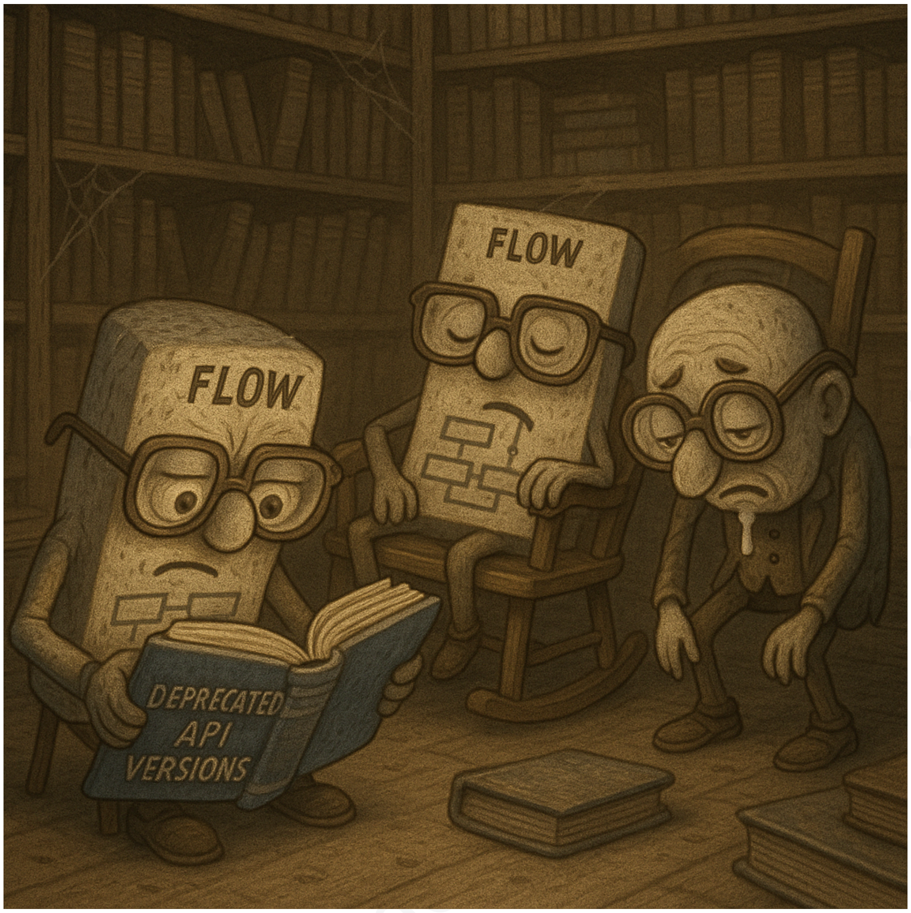
Example: 120 Visualforce pages, 50 Visualforce components, 100 Apex classes with less than 75% test coverage, most unreferenced and created several years ago.

How to assess:

- Install Org Check package https://salesforcelabs.github.io/OrgCheck/
- Run Setup / System Overview (shows number of Apex classes, Visualforce pages and total code)
- Run query: check whether an apex/page is referenced by any other item in the org: `SELECT MetadataComponentId, MetadataComponentName, MetadataComponentType, RefMetadataComponentName, RefMetadataComponentType FROM MetadataComponentDependency WHERE RefMetadataComponentId = '01p0… …AAK'`
- Run an Impact Analysis using https://happysoup.io/

Compiled by Fernando Fenandez on 5/9/25

Outdated Flows



Flows with outdated versions, no active versions, or lack of documentation, and flows that have not been updated in over a number of years.

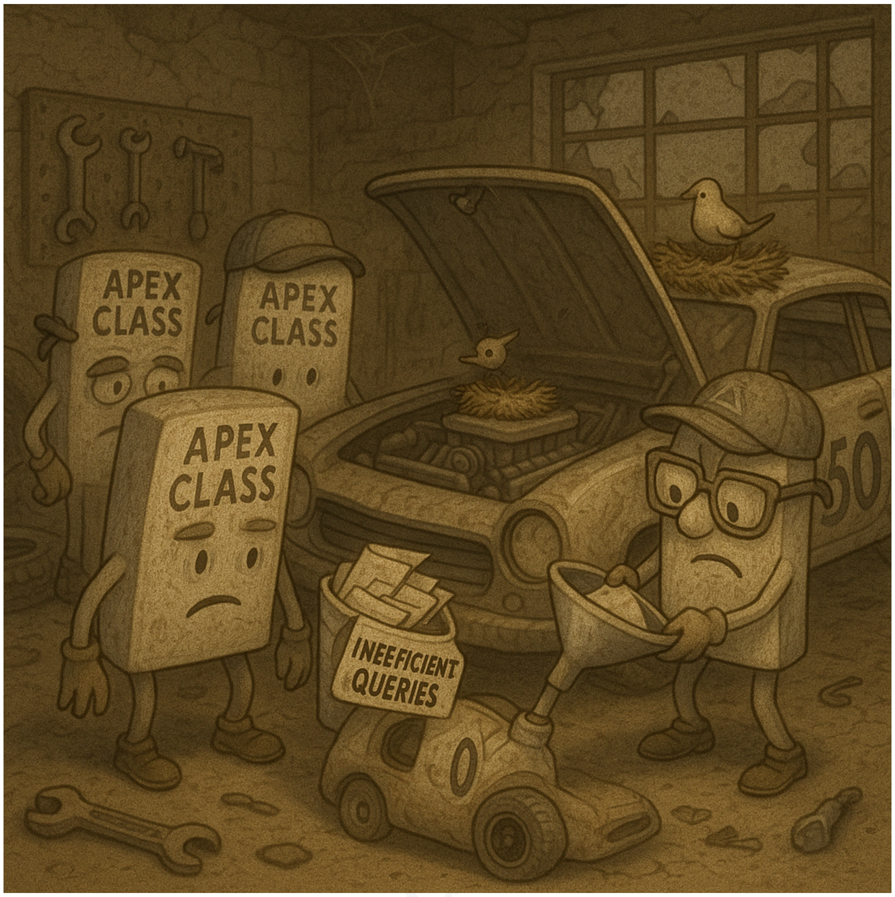
Example: 170 total flows, most of which are on deprecated API versions or have no active version published.

How to assess:

- Install Org Check package https://salesforcelabs.github.io/OrgCheck/
- Run query: check whether an apex/page is referenced by any other item in the org: `SELECT ApiVersion,IsActive,IsOutOfDate,Label,VersionNumber FROM FlowDefinitionView LIMIT 200`

Compiled by Fernando Fenandez on 5/9/25

Outdated/Inefficient Apex Code



Apex code with dead methods, inefficient queries, and SOQL/DML anti-patterns that hinder performance.

Example: 272 Apex classes with unused methods and 108 classes implementing SOQL/DML anti-patterns.

How to assess:

- Go to Setup / Scale Center / Performance Analysis / New Analysis, then go to Setup / Scale Center / Scale Insights / ApexGuru Insights
- Run Setup / System Overview (shows number of Apex classes and total code)

Compiled by Fernando Fenandez on 5/9/25

Excessive Number of Profiles and Permission Sets

A large number of profiles and permission sets, some with zero assigned users or outdated configurations.

Example: 100 profiles and 400 permission sets, many with zero assigned users.

How to assess:

- Install Org Check package https://salesforcelabs.github.io/OrgCheck/
- Run query: check for profiles with no users: `SELECT Id, Name, ( SELECT Id FROM Users ) FROM Profile`
- Run query: check for permission sets with no users: `SELECT Id,Label FROM PermissionSet WHERE Id NOT IN ( SELECT PermissionSetId FROM PermissionSetAssignment )`

Compiled by Fernando Fenandez on 5/9/25

External Access Misconfiguration

Incorrect sharing settings that allow public access to sensitive data.

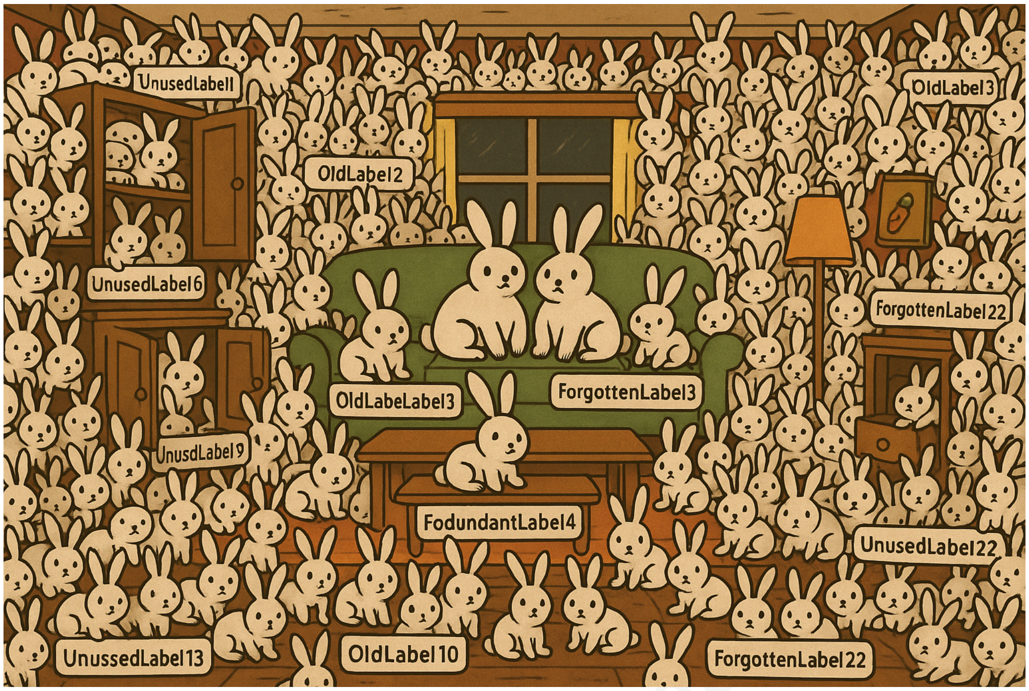
Example: 200 objects have their Default External Access set to Public, potentially exposing data so that external/portal users can view and even edit records by default. Guest user profile has edit access to a number of other objects.

How to assess:

- Run Org Health Check via App Launcher or via Setup / Security / Health Check

Compiled by Fernando Fenandez on 5/9/25

Untracked or Unused Custom Labels



Custom labels that are no longer used in the org, adding unnecessary clutter.

Example: org with 2000 custom labels, with half of them having no references.

How to assess:

- Install Org Check package https://salesforcelabs.github.io/OrgCheck/

Compiled by Fernando Fenandez on 5/9/25

Overloaded Storage

Compiled b… on 5/9/25

Org storage near, at or exceeding the capacity, driven by excessive record data and files.

Example: 245% of storage is used, primarily by large volumes of EmailMessage, Task, and Case records.

How to assess:

- Go to Setup / Data / Storage Usage then check Storage Type, Current Data Storage Usage and Current File Storage Usage
- Run query: check the trend of email message totals per month: `SELECT CALENDAR_YEAR( MessageDate ), CALENDAR_MONTH( MessageDate ), COUNT(Id) FROM EmailMessage WHERE MessageDate >= LAST_YEAR GROUP BY CALENDAR_YEAR( MessageDate ), CALENDAR_MONTH( MessageDate ) ORDER BY CALENDAR_YEAR( MessageDate ), CALENDAR_MONTH( MessageDate )`