Elements
.cloud

Change Intelligence Research Series

The impact of building the wrong thing in Salesforce

Xavery Lisinski, Jodi Hrbek & Vanessa Grant

Executive summary

The challenge of building the right thing

In this report, we uncover the hidden waste of unused customizations. Our research offers an unparalleled view into the real-world levels of adoption and the challenges of building valuable solutions in Salesforce.

Power is nothing without control

Salesforce delivers a powerful Customer360 solution out of the box, with specific variants to support different industries. But what makes Salesforce so powerful is the ability to customize the platform for any business need.

But how much of this customization is never used and is, therefore, wasted effort? Every Salesforce professional knows that there is technical debt in their Salesforce implementation. But, how much is there, and where do they stand vs. other organizations?

This report lifts the lid on adoption.

Real data

The analysis needed to be current, representative and meaningful. Elements.cloud syncs over 50,000 Orgs and over 1.3 billion metadata items monthly, so the report is based on anonymized data from 100 Production Orgs that were statistically significant. They were not selected because they were outliers. They had sufficient volume and complexity with at least 2 Salesforce Clouds, and the metadata that was analyzed was created recently over the course of 12 months in 2023.

50% never gets used

Over 51% of custom objects built on the platform never get used by the business. Nearly 40% of objects do not see any new records being created or updated beyond the first 30 days. Staggeringly, only 23% of custom objects were still being updated after 90 days. 43% of custom fields for standard objects and over 40% on custom objects have never been populated.

This represents a huge cost to the business but also contributes to the accumulation of technical debt and complexity. The impact is the quality of user experience drops and business agility plummets. But it is also the lost opportunity cost. That development effort could have been directed at projects that make the users more productive. Addressing the root cause of that waste can help organizations unlock the ROI for Salesforce and accelerate time to value.

Business analysis and product management are critical

Building successful solutions takes more than just development and deployment. It requires a product management mindset when prioritizing changes. And, then it needs rigorous business analysis to consistently capture, document, and act on true business needs.

The report quantifies the business and human cost of not putting business analysis, and product management, at the forefront of Salesforce management.

A Change Intelligence Platform enables collaboration around a single source of all change documentation and Org metadata insights. It is critical to enable rigorous business analysis and strong product management practices.

Table of contents

Authors

Xavery Lisinski

Xavery is spearheading the development of the leading Change Intelligence Platform for Salesforce. Under his leadership, the product team at Elements.cloud has increased release frequency by 800%, introducing many innovative solutions that notably improve productivity by as much as x541. He has led the incorporation of AI into every area of Elements.cloud operations. Xavery shares insights on product management and development on his blog and is the author of the "Total Story Visualization" analysis technique.

Jodi Hrbek

Jodi Hrbek is the author of the Amazon bestseller Rock Your Role as a Salesforce Admin: Create Value, Calm the Chaos, and Supercharge Your Salesforce Career and the creator of the audio course, Listen Up, Salesforce Admins! She has spent the last two decades implementing and optimizing Salesforce for startups and enterprise organizations. She's played pivotal roles in leading Salesforce implementations for two fast-paced startups within the Blackstone portfolio and has held leadership positions overseeing the Salesforce platform at several top enterprise companies. Jodi's on a mission to empower Salesforce professionals to ask better questions so they can solve the right problem with the right and "right-sized" solution.

Vanessa Grant

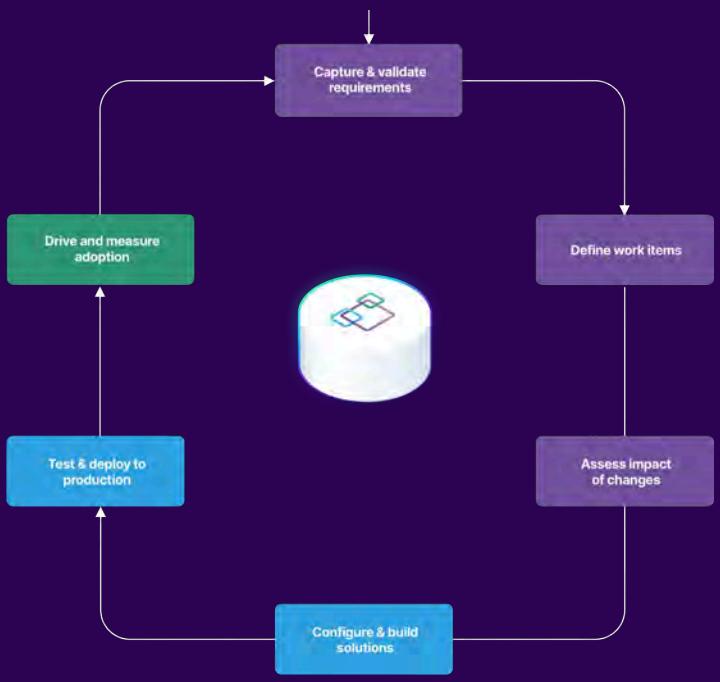
Vanessa Grant is the Salesforce Product Owner at FinTech company Mosaic and the co-host of the Salesforce Career Show podcast. She has built her career around solving business problems with technical solutions and process improvements. Besides being a certified Scrum Master and 10x Salesforce certified, she had the honor of also contributing to the creation of the Salesforce Business Analyst Certification and currently leads the Salesforce Business Analyst Group Virtual Trailblazer Community Group. As a mentor and speaker, she's committed to improving outcomes for all Salesforce projects by promoting the importance of quality business analysis, devops, and design.

Introduction

Elements.cloud is the Change Intelligence Platform for Salesforce. We believe in helping our customers understand and manage their implementations better, so that the teams accelerate time to value.

Change Intelligence is the organizational ability to understand the business and system configuration of your systems and then apply that intelligence to decide how best to implement change. As the market leader in the Change Intelligence category, we believe we must help educate the market and empower Admins, Business Analysts, and Architects to take control of their implementations and accelerate time to value.

Change Intelligence Implementation Cycle



50,000+

Salesforce Org scans every month

1,317,859,139

metadata components analyzed monthly

We sync and analyze over 50,000 Salesforce Orgs and analyze over 1.3 billion metadata components every month, so we are uniquely positioned to contribute to the discussions on best practices and trends in our ecosystem. These Orgs vary greatly in size, scope and complexity. Through the Change Intelligence Research Series, we aim to shed light on different aspects of Salesforce implementation and configuration trends. This data helps the community understand how their efforts rank across the ecosystem and drive changes in the implementation lifecycle.

The report is based on aggregating data across 100 anonymized Salesforce Production Orgs. We had no visibility into the organization name but wanted a statistically significant data set, so therefore the Orgs needed sufficient volume and complexity. They all have at least 2 Salesforce Clouds, with the majority having 3 to 4 implemented. There were not outliers in terms of configuration, customization or usage. Also, we excluded Managed Packages from the analysis because we wanted to focus on the customizations that involved manual effort to create.

Talk to us

Purpose of the report

Salesforce teams spend a lot of time and effort developing custom solutions to help transform and accelerate business operations. But how successful are those endeavors? And how strong is the ROI for organizations that innovate on the Salesforce platform?

This report sheds light on the real levels of adoption of custom capability built on the platform. To reveal the enormity of hidden cost and waste spent building the wrong things. This report aims to quantify the cost of insufficient business analysis and product management in the Salesforce implementation lifecycle.

Custom object adoption

Custom objects with no records

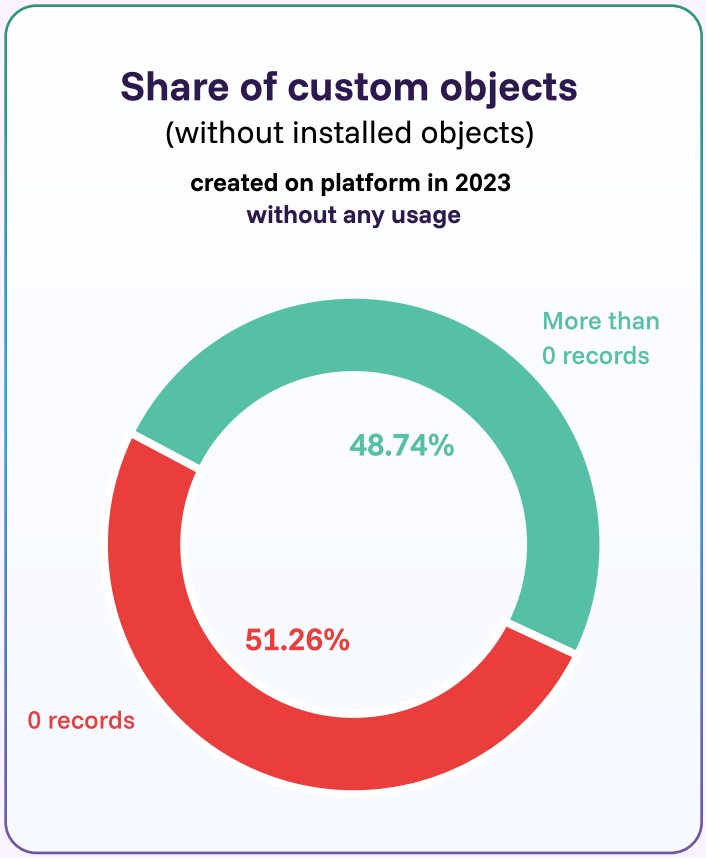
Among custom objects created in 2023, 22% were created by Admins (users with admin permissions) and the remaining 78% were objects installed through Managed Packages. A significant development effort is spent by teams to build custom functionality so we excluded Managed Package objects from the analysis. In this report we did not look at how many Managed Packages were installed and never used.

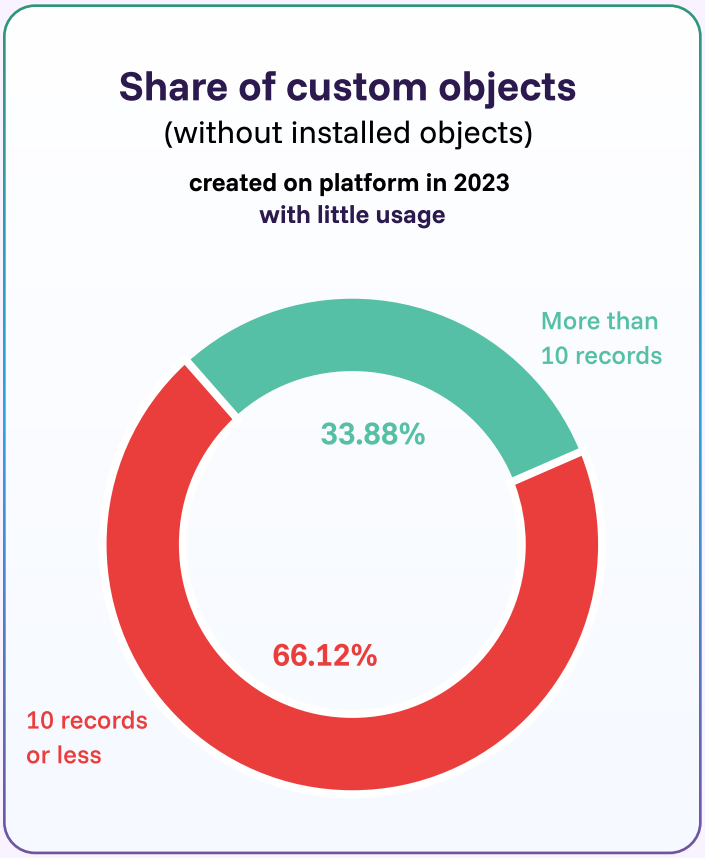
Of the custom objects built by Admins and deployed to Production, 51% were never used within a year of being created. They had 0 records.

More than 10 records

We also compared custom objects built by Admins for those that, after a year, had more than 10 records vs. objects with less than 10 records. We were interested in custom objects that get some initial adoption, where a few records are created, before the retention drops to zero.

Over 66% of custom objects ended up with 10 or fewer records within a year of being created in Production.



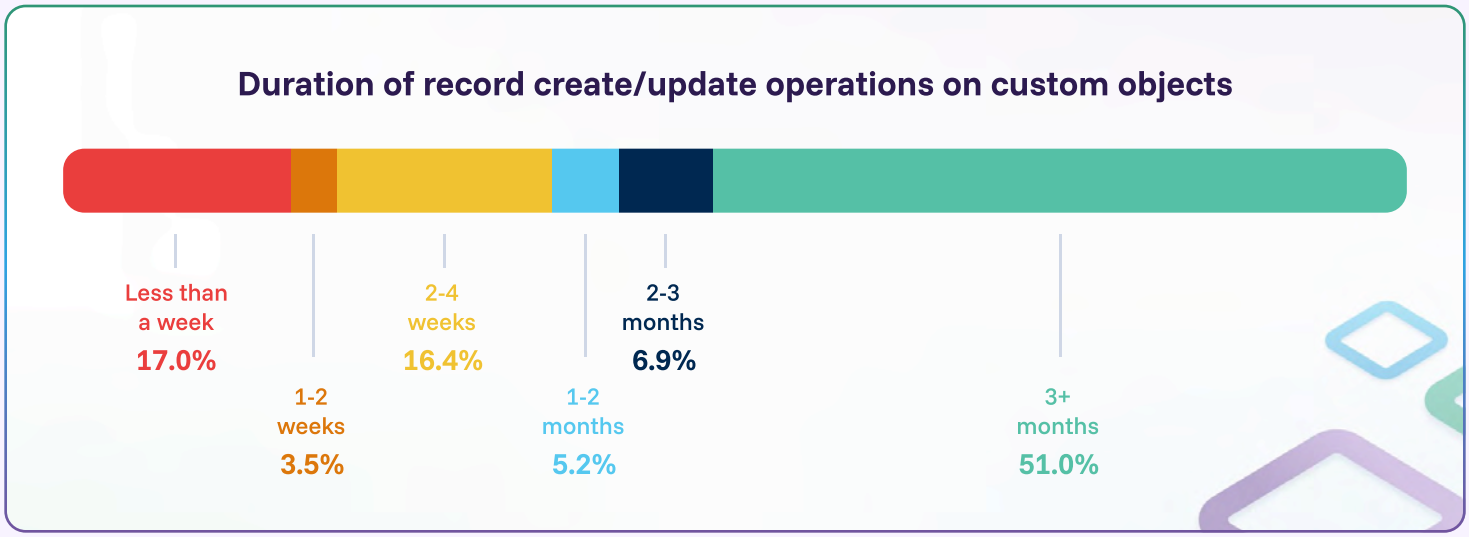


How data is populated over time

For custom objects built by Admins that had at least 1 record, we analyzed the last time any record was either created or modified on each object. We then compared that to when the object was created. We recognize that some custom objects are created to store more static data that changes very little after it is added: for example - target personas.

We discovered that 17% of custom objects saw no new records being created or updated after the first week after being created. This climbed to 36.9% after the first 30 days of custom objects being created. As much as 49% of custom objects saw no new records being created or updated after the first 90 days.

This could be because some of the objects hold static data so it is rarely updated after it was created. But not all of them. So this is a worrying statistic.



Key Findings

Talk to us

51%
of built custom objects never get used

66%
of built custom objects have less than 10 records

49%
of built custom objects see no new records being created or updated after the first 90 days

Custom field adoption

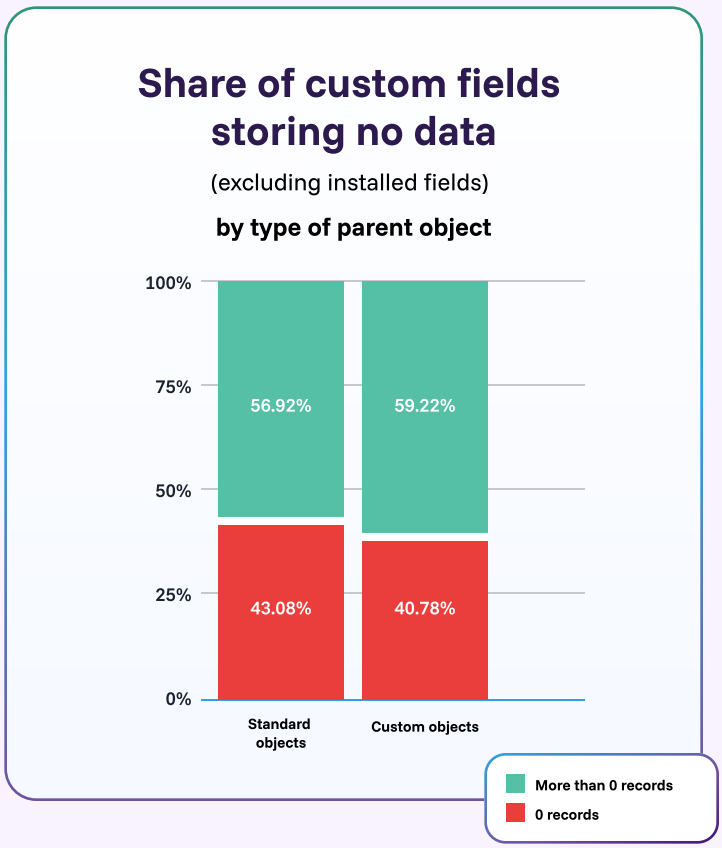
Fields with zero population

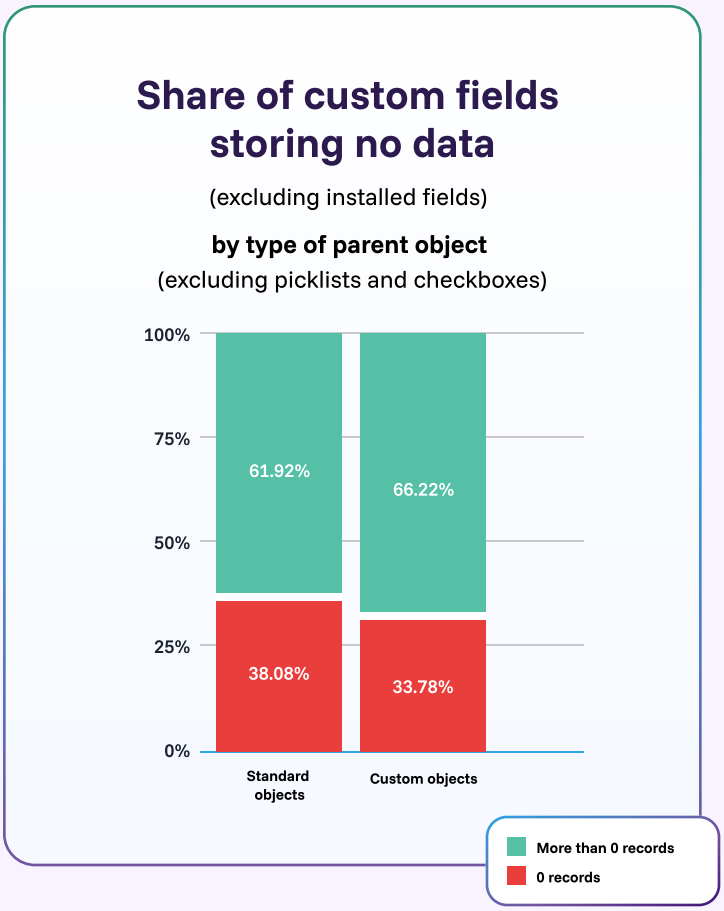
Among all custom fields created in 2023, our research revealed that 47% of them were added by Admins and 53% were added by installing Managed Packages. Once again, since real development effort is spent by teams on building custom functionality, we decided to exclude fields installed from Managed Packages from the analysis. So we analyzed the fields that had been added by Admins.

We discovered that 43% of custom fields on standard objects and nearly 41% of custom fields on custom objects held no data within a year of being created.

There are some types of fields where the 'null' value (meaning empty) is expected or is a valid value. Checkboxes are the most common example, but picklist fields can have null as the default value or one of the available values.

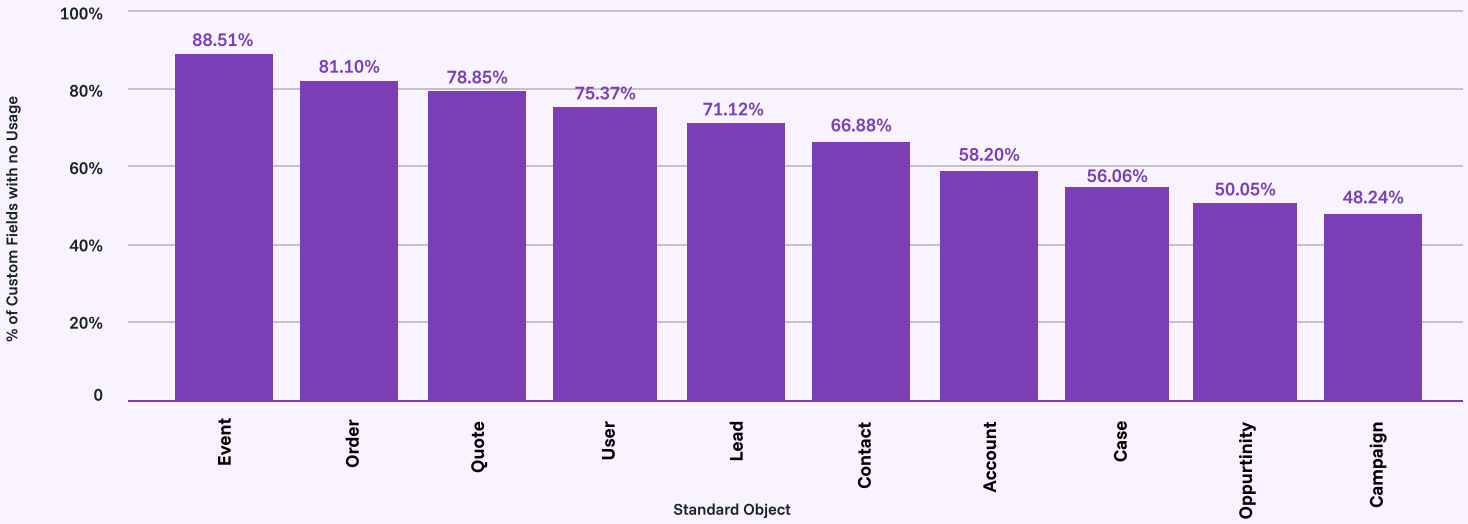
So we have also excluded those types of fields entirely from the analysis. Still, the figures are very high. 38% of custom fields for standard objects and over 33% on custom objects were held no data within a year of being created.



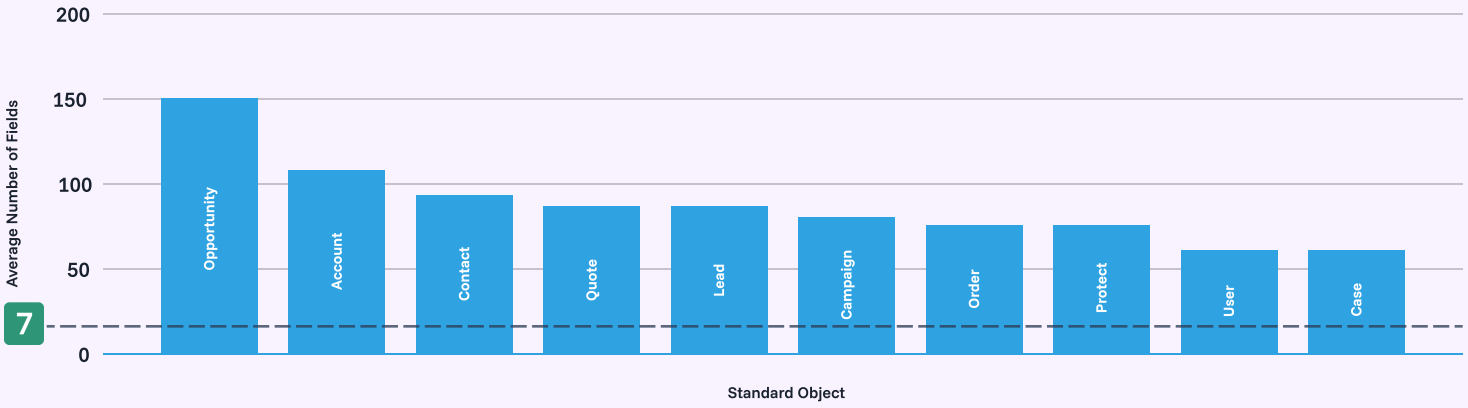


Standard and custom fields with zero population for key standard objects

We then analyzed the custom fields for the 10 core standard objects. For the core objects used in Sales and Service Clouds, the number of custom fields that never got used ranges from 88% to 48%. This is significantly higher than the average non-adoption across all standards objects, which was 38%. This is because these core objects are used by multiple different departments and usually see much more enhancement requests from the business.



If we look at the Opportunity object, 50% of requested custom fields are never used. In our previous report titled Untapped Opportunities In User Experience, we found that page layouts for standard objects had a huge number of fields. The Opportunity object, on average, has more than 150 fields on its page layouts. If we consider that 50% of those fields are never used, then this leads to unnecessary complexity and cognitive overload for the sales users. **The ideal number of fields for a great UX is 7.**



Key Findings

43%
of custom fields built for standard objects are never used

41%
of custom fields built on standard objects held no data

For the 10 core standard objects
between 48% - 88% of built custom fields are never used

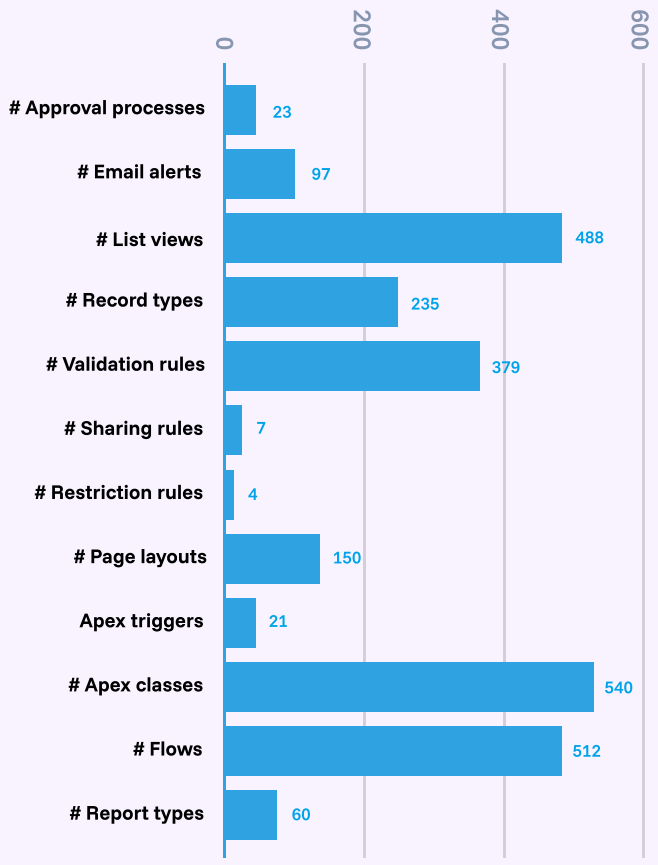
Additional customization

For the custom objects built by Admins in 2023 with 0 records, we analyzed the related metadata that was also created. This represented a development effort to create metadata such as automations or UX to support the business requirement.

We are able to provide this level of analysis because our Change Intelligence Platform creates sophisticated dependency trees and grids. These display the metadata that is related to any other metadata item. We have access to this data for the report.

For 100 representative Production Orgs, we looked at the metadata linked to just the custom objects with no data. This does include the metadata that is automatically added when you create an object: for example, default record type or default page layout.

- 540 Apex Classes were built referencing the unused objects
- 512 Flows were built referencing the unused objects
- 379 Validation rules were built and never used
- 150 custom Page Layouts were never opened
- 498 custom List Views were never opened



Full breakdown of types of features built around unused custom objects

Cost of development

The effort to build and deploy the custom objects and fields, and then implement related functionality, is always significant.

This effort includes analyzing requests, communication with stakeholders, meetings, emails, Slack discussions, design, documentation, development, testing, deployment, and user training. There is also the hidden opportunity cost. If the teams spend all this effort building functionality that never gets used, then the business both never solves the problem they wanted to address, and they wasted time they could have spent delivering something else.

There is also a cost to the implementation team which is the accumulation of technical debt and org complexity, which makes any future changes more challenging. Based on our research, we can estimate that **in any given year about 40%-50% of development effort contributes to Org complexity and technical debt.** Over the course of a few years, that percentage compounds to the point where the accumulated complexity makes it impossible to use and configure Salesforce anymore.

Bear in mind that we have not considered the technical debt added by unused Managed Packages. We did not look at metadata related to standard objects that were never used. Nor automations tied to custom fields built on standard objects that were never populated. We also did not look at complex automations that were built and never run. There is still plenty of other development activity which was outside of the scope of this report but is likely to reveal similar levels of waste.

The huge cost to users is the unused fields that over-complicate the page layouts. This impacts user adoption and data quality. We know from our previous issue Untapped Opportunities in User Experience that record pages on Opportunity object have on average more than 150 fields. That is an overwhelming number of fields to put in front of the user. If 50% of those are not being used at all, as data above suggests, then that is the case for removing them altogether or at least from the user interface.

In the Research Methodology chapter at the end of this report we have given some ballpark estimates on the time taken to implement each of the different metadata types. You can then work out the impact for your own Org.

> in any given year about 40%-50% of development effort contributes to Org complexity and technical debt.

#expertsreact

> The findings regarding the extent of waste have astonished me

Jodi Hrbek

Founder & Event Host, The DemoOrg DemoJam

What is surprising about the data, and what isn't?

Because it's relatively easy to spin up solutions on the Salesforce platform, there's a tendency to jump in and solve problems with much fewer questions and analysis than would typically be required for traditional software development. I wrote a book and launched a coaching program specifically to help Salesforce professionals avoid this because of the amount of wasted effort, re-work, and lost opportunity to create value that I've witnessed, so I certainly expected to see some waste. However, even with that perspective, the findings regarding the extent of waste have astonished me.

I'm struggling to explain how it could be that more than half of custom objects were never populated with any data, especially since Managed Packages or Industry Clouds that install copious custom objects in an Org have already been excluded.

Objects with zero records likely aren't an indication of a problem with end-user adoption but are more likely a symptom of a failed intake or governance process. Without proper prioritization, solutions get built for issues that turn out to be a not-so-high priority after all and, therefore, never get the required mindshare or resources to be successfully deployed. I've also seen this stem from stakeholder apathy in which they asked for a shiny set of features but didn't invest the required work to implement and operationalize the solution. In these instances, solutions tend to die on the vine before seeing the light of day.

Regarding the objects that had some utility but little retention, I suspect the number is slightly inflated based on the definition of adoption being tied to the number of records created after going live. Complex Orgs and custom apps will often have objects built to support a data model with the expectation that there would be a small number of records indefinitely. Even if we split hairs a little on how to account for that small use case, the fact is that a tremendous amount of solutions are getting built that aren't being used.

As far as unused custom fields are concerned, I can think of a few use cases that might explain why there would be fields that one would expect to be blank most of the time, such as ones added for the "just in case" edge case or for a part of the business process that only happens under specific scenarios. Still, so many custom fields, particularly those on standard objects, without any data at all is alarming, although not necessarily surprising.

#expertsreact

Custom fields can be even more pernicious than custom objects because they're so easy to add. Stakeholders know this and often request an on-the-fly addition, thus popularizing the all-too-frequent refrain familiar to every Salesforce Admin, "I just need you to add a few fields." Yet, more often than not, once you dig into the 'why' behind the ask and the desired business outcome, what they need either already exists or warrants something entirely different.

Because they're so easy to add, business users and even some Salesforce Admins don't recognize that adding fields—including choices made about the field type and where they are located—can be as impactful to the Org data model as adding new objects and therefore warrant the same level of scrutiny.

What is the 'expected' level of waste?

The beauty of the Salesforce platform is that it doesn't work like traditional software. With clicks, not code, and in a relative jiffy, we can build apps to take down clunky legacy databases and rogue spreadsheets posing as business processes. Because of the relatively light lift, we can solve problems—including the vexing ones that cause issues but may not have the highest business impact—with a config solution that wouldn't warrant the cost of traditional software development. Salesforce solutions, by definition, are agile and can deliver fast time to value. We can experiment and try things we couldn't do in a more traditional development process.

While it does require governance and a product management mindset, we mustn't over-rotate by treating Salesforce configuration exactly the same as custom dev, which would water down the secret sauce that makes the platform so powerful. We should leverage its benefits for innovation and speed, and consequently, in doing so, we may see some "waste" over time as an expected output.

If there's a high-priority business initiative, it may make sense to set up a quick point solution, even if it's known that it will be short-lived.

If a team has an adjacent use case they want to explore solving on the platform, it might be advisable to quickly set up their process and let them take advantage of a simple custom app.

The critical element to utilizing Salesforce in this manner is to be intentional about the trade-offs involved if even a tiny amount of effort is expended, as there's always an opportunity cost. Furthermore, teams must ensure point solutions, interim use cases, and low-cost problems are met with a proportional solution and not with bloat and overkill that would defeat the point of the platform's flexibility. And, of course, all solutions must be monitored over time and adjusted or retired accordingly.

Even mature organizations with solid governance will have some waste, which is good as it indicates a willingness to take advantage of Salesforce's unrivaled ability to easily capture, share, and report on data for exploratory use cases or point solutions. Older Orgs will inevitably have fields that have become defunct over time that will need to be maintained in the system for context or compliance. This is expected, and well-maintained Orgs will have regular audits and cleanups to remove, archive, or adjust the label and description to reduce confusion and improve data quality.

#expertsreact

How does wasted development hurt an Org?

From an end-user standpoint, unused solutions are wasted real estate and add needless complexity to their digital experience. For Admins, anything added to the system has a tax over time regarding system limits, Org complexity, and the mere presence of more things to consider when making changes.

In addition to being a tremendous drain on time, field proliferation adds to user frustration when navigating through the system, running reports and list views, and trying to determine the “actual” fields to use for filters or in a display.

Admins face a similar issue when faced with automation or other requirements on an object and no clear way to identify which data points are or aren’t valid. Fields that tell the story for only a tiny fraction of records or, worse, indicate the presence of data that doesn’t actually exist anywhere in the system, hamper the ability to leverage data for story-telling, analytics, or more advanced use cases that AI might deliver.

However, the more significant impact isn’t on the Org itself but on the organization that lost an opportunity to derive real business value or failed to solve a stated problem and the Salesforce team that did the work in lieu of other critical activities and missed a chance to put a win on the board.

#expertsreact

Vanessa Grant

Certified Salesforce Business Analyst & Product Owner

MOSAIC

What is surprising about the data, and what isn't?

Sadly, I can't say that I am surprised by the key findings. They speak to what so many Salesforce professionals and customers are seeing, but maybe didn't have the actual statistics to prove – implementing and maintaining a healthy Salesforce Org is HARD. It requires a ton of resources, analysis, process, documentation, and governance.

Unlike Petco, which will usually hand you a care sheet on how to keep your new hamster healthy and alive, Salesforce account executives are usually not as clear with new customers on how much they will need to invest outside of the signed Salesforce contract in order to maintain and optimize their investment. It's not enough to hire a freshly certified Salesforce Administrator or give every superuser admin rights. Customers need experienced Salesforce professionals that will say "NO" so that their Orgs don't turn into Frankenstein.

There are currently 14 skills highlighted in the Salesforce Admins Skills Kit. A Salesforce team (even if it is just a solo Admin) needs to wear ALL of those hats. Missing any of those hats leads to technical and design debt. Most of my consulting engagements were with Salesforce customers that learned this lesson the hard way. They would accumulate debt in their Org and find themselves 5-10 years down the line with an undocumented, poorly architected, Frankenstein-like mess. This led to them needing to invest huge amounts of time and money to re-implement Salesforce in brand NEW Orgs because it would be easier to burn the old Orgs to the ground and start fresh than to undo the damage already done.

So how to explain this data? In my opinion:

1. Poor Product Management - Either the right projects weren't prioritized or the right resources weren't put on the right projects. Product Management is all about maximizing business value given limited time and capacity. Unused objects, fields, and managed packages are a sign of wasted resources.

2. Poor or no Business Analysis - If the custom objects and fields that were built were never used, then the wrong things were built. Period. Which leads to…

3. Poor Problem Solving - If the business problem and the desired business value isn't clearly understood, it's impossible to really know the best way to create a solution in Salesforce for it.

#expertsreact

4. Poor Data Management and Process Automation - Once the solution is in place, how will the business process be changed to incorporate it? Who is responsible for that data in those custom fields getting entered properly? Do they have to do it manually or can it be automated? Without thinking through these answers, it would not take long at all for the data in those shiny new objects and fields to get stale, become untrustworthy, and therefore become unused by the business.

5. Poor Change Management and Communication - You can build out the BEST Salesforce solution, but if no one knows about it or how to use it, then it will not be successful. Thoughtful roll out plans include planning, training, communication, success metrics, and feedback collection.

6. Poor Design - If solutions were created, but never adopted, poor design choices could be a factor. There have been many times that I have seen solutions get built out in Salesforce that could solve the business problem and get to the desired outcome… but the users hate it. There is no point in building something if no one adopts it. It's so important to remember that solutions are created for humans and that the quality of their experiences has a huge impact on the success of any given Salesforce project.

7. Poor User and Security Management - I've seen many Orgs that have far too many people with admin rights because they did not adhere to the principle of least privilege. When Salesforce experts within an organization aren't supported in deciding who should and shouldn't have admin rights, or there isn't clear governance or guidance around who is allowed to have them, then people without the know-how will get the power to create objects and fields on whim, often straight into production, and rarely documented.

   Bad reasons to give users admin rights include:

   - They want them.
   - They are the CEO, so of course they should have them.
   - They had those rights at a previous organization.

These bullets represent significant challenges for all Salesforce teams, but they are ultimately consequences of a deeper issue - a mindset that treats Salesforce as a "set it and forget it" tool rather than a robust platform requiring ongoing care and investment. Too often, Salesforce teams and solo Admins are tasked with rapid configuration without adequate time or resources for analysis, prioritization, and sustainable maintenance practices.

Salesforce teams need to be empowered from the top to work in the best interest of the Org. Salesforce customers need to learn that what is in the best interest of the Org is ultimately what is also in the best interest of the company.

#expertsreact

What is the 'expected' level of waste?

It is impossible to avoid technical debt entirely. In fact, eliminating it entirely is probably ALSO a sign that projects were not prioritized properly. The important part is to make sure that the waste isn't impacting the users or getting to the level where it impacts the long-term health of the Org.

There is always going to be some level of technical debt, but you want to make sure that you have governance, documentation, processes, time, and resources in place to manage it. The removal should be prioritized appropriately and the method for deprecating things should be consistent.

Ian Gotts wrote an excellent article on technical debt as well that I think would make great supplementary reading here.

How does wasted development hurt an Org?

Wasted development isn't neutral, it is a net negative. This shows up in the following ways:

1. Time wasted to create it in the first place as evidenced by one of the key findings in this study.

2. Cost of delay by not using that time to build things that would add value to the business.

3. Time wasted by the users trying to get their work done around the wasted development.
   This includes a lot of what Jodi covered. It is hard to build a report when you're trying to choose between fields "Date Signed," "Date Signed2," "Contract Signed," etc. and only one of those fields is currently in use. This is especially true if the fields aren't documented properly.
   I once saw an Org that had 800 fields on the Account object. Just imagine the time wasted by the users to navigate to the information they need, let alone to create meaningful reports!

4. Time wasted to deprecate the development work that didn't add value to maintain the health of the Org and avoid bloat.

5. Time wasted in trying to find great Salesforce professionals that will even want to work on an Org that is bloated and that users hate. We are a passionate bunch in the Salesforce ecosystem, and it is hard to keep us if we aren't empowered to do great work.

Bonus negative – You'll waste your AI's time, or even worse, your AI will get confused by the waste and it'll impact its effectiveness. If you want to prepare your Org for an AI-driven future, you'll want to make sure that your AI is being fed with meaningful data. Garbage in, garbage out!

Expert recommendations

Jodi Hrbek

Founder & Event Host, The DemoOrg DemoJam

Increasingly, the Salesforce platform is becoming an enterprise application inside organizations. With that, we're seeing the rise of the Business Analyst (BA) and Product roles to facilitate more rigor and analysis around Salesforce solutions. Companies that haven't adopted this model or smaller Orgs that can't justify more headcount can incorporate these concepts as a critical ingredient for evaluating, prioritizing, and designing Salesforce solutions—even for those "just-in-time" changes.

It is my experience that Salesforce teams are generally well-practiced in conducting the due diligence required for a new deployment or at the onset of a significant project, but small enhancements and "Business as Usual" (BAU) changes garner much less analysis or consideration. Business leaders contribute to this when they have expectations of rapid turn-around times and demands for quick-fix solutions because, "It's Salesforce, not software."

Common culprits include:

- **The sales leader who "just" needs a few fields.**
- **The team that needs a "quick way" to track their new process.**
- **The group that "had an idea" about something they want to add to the system.**

This Change Intelligence Research Report helps change the narrative about how easy it is to add fields and build Salesforce solutions, ensuring teams consider a more holistic view regarding the impact on the system, the business, and the cost of building the wrong thing.

As an ecosystem, we must continue reinforcing the message that just because we can doesn't always mean we should. There is an opportunity cost for every solution we deliver. If we're solving the wrong problem, if we're solving a $1 problem with a $100 solution or vice versa, or if we're solving the problem with a solution that no one wants to use, we're missing critical opportunities to drive real business value, not to mention the toll this takes on Salesforce Admin time—a precious commodity in most organizations—and their ability to put wins on the board, an essential element for their professional progression.

Salesforce advanced the conversation a few years ago by creating the BA Certification and adding Product Management to the Admin Skills Kit. We must continue to emphasize and grow these skills in our Admin teams, including the importance of asking questions and engaging in conversation, analysis, and planning before writing or executing stories. We need to empower them, so they have the confidence and the skills to advocate for further analysis if needed when pushed for a quick change by a boss or project manager by socializing the high cost of wasted work when that step is skipped.

An Admin recently confided that they "don't have time to ask all those questions because they have too much they need to get built in their sprint." Another Admin recently shared they get pressure when they spend too much time upfront in discovery and design. With a platform that lends itself to quick builds we must break some institutionalized bad habits of thinking Salesforce changes don't warrant the same level of rigor as other systems, as this research explains what's at stake. The question shouldn't be, why aren't we building yet? The default question—which we can all reinforce—must be how could we possibly start building when we don't yet know enough about the problem we're solving?

Expert recommendations

Vanessa Grant

Certified Salesforce Business Analyst & Product Owner

While there is the larger problem of Salesforce customers (and sometimes even consultants) needing to get into that “shift left” mindset generally, there are a number of activities that Salesforce teams can try to start incorporating into their processes to avoid the kind of waste that we see in this study.

Put on your Product Owner / Product Manager hat

It is essential to approach Salesforce projects with a product ownership mindset. By working with stakeholders, and perhaps even starting up a Center of Excellence (CoE), Salesforce teams can get direction on how to prioritize work based on business value, get stakeholder alignment, build trust among stakeholders, and educate them on everything that goes into delivering successful Salesforce solutions.

Implementing proper Salesforce DevOps practices utilizing version control, robust testing, and controlled release processes ensures smooth, repeatable project delivery cycles with higher quality and less technical debt accumulation. Aligning the Salesforce team on these processes and having them documented in a system like Jira can also go a long way for tracking work and making it visible to stakeholders.

Taking care of a Salesforce Org is a team sport. Even solo admins will likely some day have to call a consultant in for a one-off project. It’s important that governance is also put in place so that the Org can be guided by the same principles no matter who is doing the development on it. Besides documenting configuration and code standards for an Org, documenting team values and best practices like “clicks, not code,” the Salesforce Well-Architected framework, guidance on naming conventions, and even criteria for identifying and removing unused customizations like objects, fields, automations can go a long way for the long-term health of an Org.

Incorporate Business Analysis for ANY configuration or code change in your Salesforce Org

It is essential to approach Salesforce requests, big and small (“can you just add a field that…?” requests) with a business analyst mindset. Before implementing any new customization in Salesforce, it's crucial to thoroughly understand the underlying business needs. This ensures that what is being built will effectively address the right problems and drive actual business value. Begin by clearly defining the specific business challenge to be solved and document the current state of how that process works. Gather consensus among stakeholders on the problems, pain points, and future desired state.

Mapping out the existing business processes in detail, identifying handoffs, inefficiencies, and areas for optimization will help the team to avoid missing steps. It’s also important to try and optimize the business process before you add technology. If you have a bad business process, then throw Salesforce on top of it, you just have a faster bad process.

Ongoing collaboration with stakeholders to validate requirements, user stories, and the team’s understanding of the impacted processes will not only make the stakeholders feel heard, it will also help manage scope. Documenting along the way will also help future Admins of the Org understand what was built and why it was built that way.

While business analysis is crucial, balance is also key. Don't skip analysis, but don’t overanalyze either. Xavery Lisinski has an outstanding jelly mold analogy on this.

Expert recommendations

Vanessa Grant

Certified Salesforce Business Analyst & Product Owner

Incorporate Change Management and User Experience in your definition of done

It is essential to approach Salesforce work with humans in mind. Developing new Salesforce capabilities is only half the battle - maximizing user adoption requires comprehensive change management practices and attention to user experience.

Engaging stakeholders early and often to understand impacts will improve the quality of your solutions. Providing ample communication, training and support resources, and a thoughtful rollout plan with a feedback mechanism to users will also help to smooth their transition.

Communicating and convincing business leadership on the importance of these practices and why taking the time to do things right (instead of simply “fast”), is essential for the long-term health of a Salesforce Org. While these practices are neither easy to implement nor to keep up, they are absolutely worth it to avoid the fate of so many Salesforce customers who have had to re-implement Salesforce because their original Orgs became so bloated with technical debt.

How Change Intelligence can help

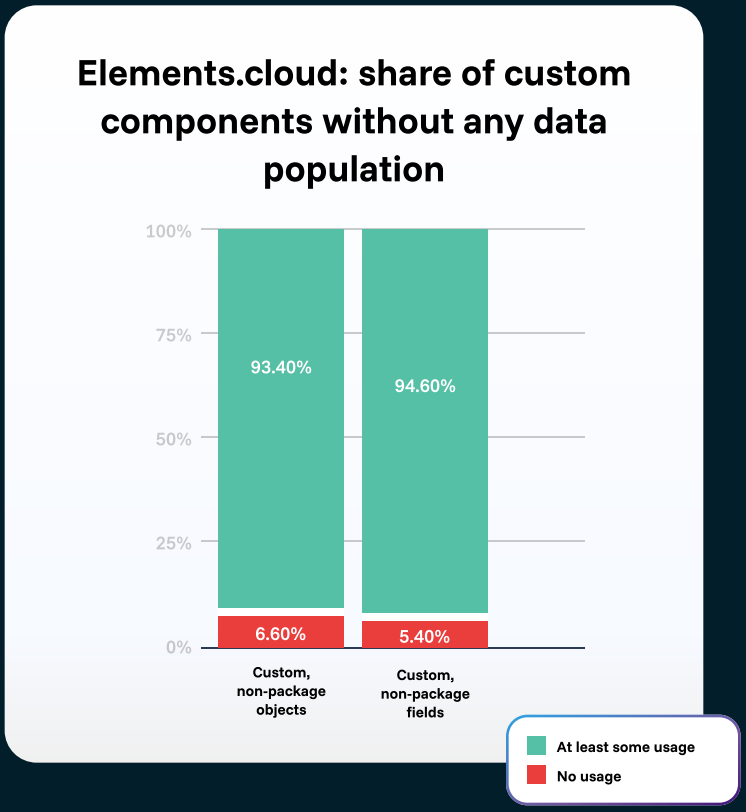
Building the right solutions for the right problems can be mastered, and the cost of wasted effort need not be as high. The Elements.cloud Salesforce implementation team utilizes the Elements.cloud Change Intelligence Platform capabilities and follows rigorous business analysis. The results are clear and a stark difference from the research findings.

1 7.8x lower waste

The level of waste was 7.8x times lower than the average::

- **6% of custom objects held no data**
- **5% of custom fields held no data**

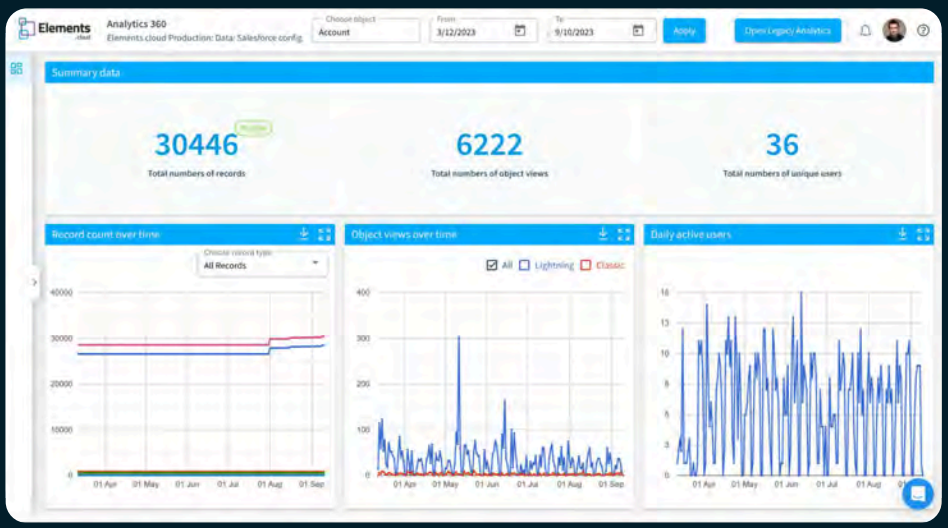
This shows that even with strong business analysis in a business that is growing and changing very rapidly you can still end up building customizations that are not used. But this is dramatically lower than the average in the Salesforce ecosystem.



2 Adoption analytics

The Elements.cloud Analytics360 dashboards show what is being adopted and what is the trend of activity over time. You can assess both record creation and modifications over time.

If you have Salesforce Event Monitoring licenses, Analytics360 uses this data to provide more granular insights, like which users (profiles/roles) utilize your objects.



Elements .cloud

3 Visual requirements capture

The research shows that too little time is spent on business analysis and understanding the true business needs. This can be a lack of rigor or it could be that there are miscommunications because requirements are captured as textual documents.

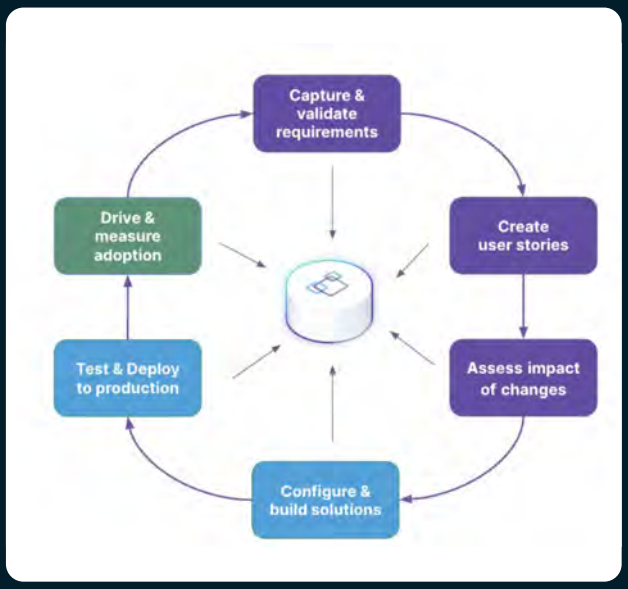
Using process mapping with the business you can quickly get a shared agreement on what the true need is. A detailed process diagram becomes a visual story map and allows you to capture work to be done. And ElementsGPT can accelerate that mapping by auto-generating process maps from a text prompt and the LLM's knowledge of best practice processes. This can be a useful starting point.

Salesforce recommends the UPN process mapping approach and has training courses in Trailhead. The Business Analysis Certification requires a knowledge of UPN.

4 Single source of collaboration

The power of a Change Intelligence Platform is that it centralizes all the change documentation, allows it to be connected, so that all documentation can be seen in the context of any change. The documentation includes requirements, process maps, ERD, architecture diagrams, metadata dictionaries, metadata documentation, metadata dependencies, metadata change logs, user stories, and releases. With this information it is possible to provide impact analysis on any change considering the technical, business, and regulatory implications.

A Change Intelligence platform plays the strongest role in the first 3 purple boxes (Business Analysis). But the documentation that is created is then referenced and added to during the blue (DevOps) and green (Operations) boxes.



Elements .cloud

5 Powered by AI

AI can supercharge the business phases of the implementation lifecycle. Whilst it can dramatically speed up some activities it can also improve the accuracy of the outputs.

Process map generation: ElementsGPT can take a prompt and draw a process diagram in the UPN format. The input could be the transcript of a user interview, a procedure, or a summary of a sketch, diagram or flowchart. ElementsGPT uses the input, the LLM's knowledge of process best practices and the internal prompts to develop a process diagram. ElementsGPT can be used to create a new diagram or to drill down from an existing activity box. When drilling down, ElemensGPT also uses the inputs, outputs and activity box information to guide the process diagram creation

User story creation: You can highlight one or more activity steps from a UPN process diagram and ElementsGT will generate multiple user stories that follow the best practices. ElementsGPT uses the inputs, outputs, activity text, and resources, and combines them with the internal ElementsGPT prompts to write the user stories. They even have acceptance criteria tailored to the systems you will be using that are described as resources.

Solutioning: ElementsGPT can take the user acceptance description and acceptance criteria and then analyze all the metadata in a Salesforce Org to suggest a solution. The solution could suggest the reuse of existing metadata or the development of new metadata. This should reduce the levels of technical debt.

Metadata documentation: ElementsGPT and Einstein1 use metadata labels, APIs and descriptions to provide its results. The more accurately these reflect how the metadata is used, the better the AI results. Elements metadata dictionary makes it easy to maintain this documentation as it enables the metadata to be seen in the context of the processes where it used and the dependencies with other metadata.

Research Methodology

Where does the data come from?

Elements.cloud is a Change Intelligence Platform that allows Salesforce customers to connect, scan, and analyze their Org configurations. The results of our analysis are then stored in the Mongo database and used to feed insights within our application, like aggregate analytics, dependency trees, dynamic metadata grids, and others.

How did you access the data?

Access to the production data is severely restricted in Elements. The VP of Product performed analysis using Mongo Charts and Aggregation Pipeline to get the datasets.

How did you choose the data?

The report is based on aggregating data across 100 anonymized Salesforce Production Orgs. While we sync and analyze 50,000 Orgs monthly, they all vary greatly in size, scope and complexity. We randomly selected 100 Production Orgs that show sufficient volume and complexity, and that had scheduled Apex jobs set up in our Managed Package to calculate usage data. The sample of 100 Salesforce Orgs was deemed to be statistically significant.

Among 100 randomly selected Orgs, all had a significant Salesforce user base, ranging from a few hundred users to tens of thousands. They all have at least 2 Salesforce Clouds, with the majority having 3 to 4 Salesforce Clouds implemented.

We were able to identify Orgs that match the required criteria using specific queries in our MongoDB and then use their Mongo record IDs to further query collections that store dependency information. The dependency collection is where we store information about relationships between individual metadata components.

The queries we ran and the dataset we generated did not include any information that would allow us to identify the specific Orgs or companies. Even we don't know which companies were included in the dataset.

How many components were analyzed?

Across 100 Production Orgs, and looking for metadata that was created by users between 1st of January 2023 and 1st of January 2024, we found:

- **3644 custom objects, 794 of which were non-package objects**
- **84,834 custom fields, 39,619 of which were non-package fields**

How did you measure usage / adoption in Salesforce?

Adoption can be measured in many ways. What is the 'right' level of adoption really depends on the business context behind each field or object. There might be 'Lawsuits' objects, used to store records of any lawsuits against the company or its employees. Having zero or little records in this case would be a very good business metric. Equally, there might be an object for storing target market segment information, which in an established market would most likely not see any record changes after a while.

Considering that Salesforce is primarily tracking customer data, and that even in instances where records are expected to not change much over time, it is a fair assumption that 0 records captured for an object and 0 records filled out for a new field within the first 12 months after creation are indicators of lack of usage.

Elements' customers can schedule Apex jobs within our Salesforce Managed Package to calculate both object and field data population. This allows users to get the total number of records on each object, including daily changes, and the total number of records where each field is not null. The results of those jobs are stored in our Mongo database. This data was pulled from the sample Orgs for our analysis.

Calculating wasted effort

So that you can make a business case for investing more time in rigorous business analysis and product management, we wanted to provide assumptions for the average time it takes to implement metadata - communication with stakeholders, configuration, testing, deployment, training - for each type.

They are listed alphabetically and are average estimates:

- Apex Classes: 10 hours
- Apex Triggers: 2 hours
- Approval Processes: 2 hours
- Custom Fields:
  - Basic Fields (String, Boolean, Currency, Date/time, Email, Phone, Percent, Text area, Rich text area,): 15 minutes
  - Complex Fields (Lookup, Picklist, Multipicklist, Double, Auto number, Metadata relationship, Master detail): 30 minutes
  - Very Complex Fields (Url, Address, External lookup, Summary, Formulas): 45 minutes
- Custom Objects: 1 hour
- Custom Page Layouts: 2 hours
- Custom Record Types: 1 hour
- Email Alerts: 30 minutes
- Flows: 8 hours
- List Views: 15 minutes
- Report Types: 30 minutes
- Restriction Rules: 30 minutes
- Sharing Rules: 30 minutes
- Validation Rules: 1 hour