

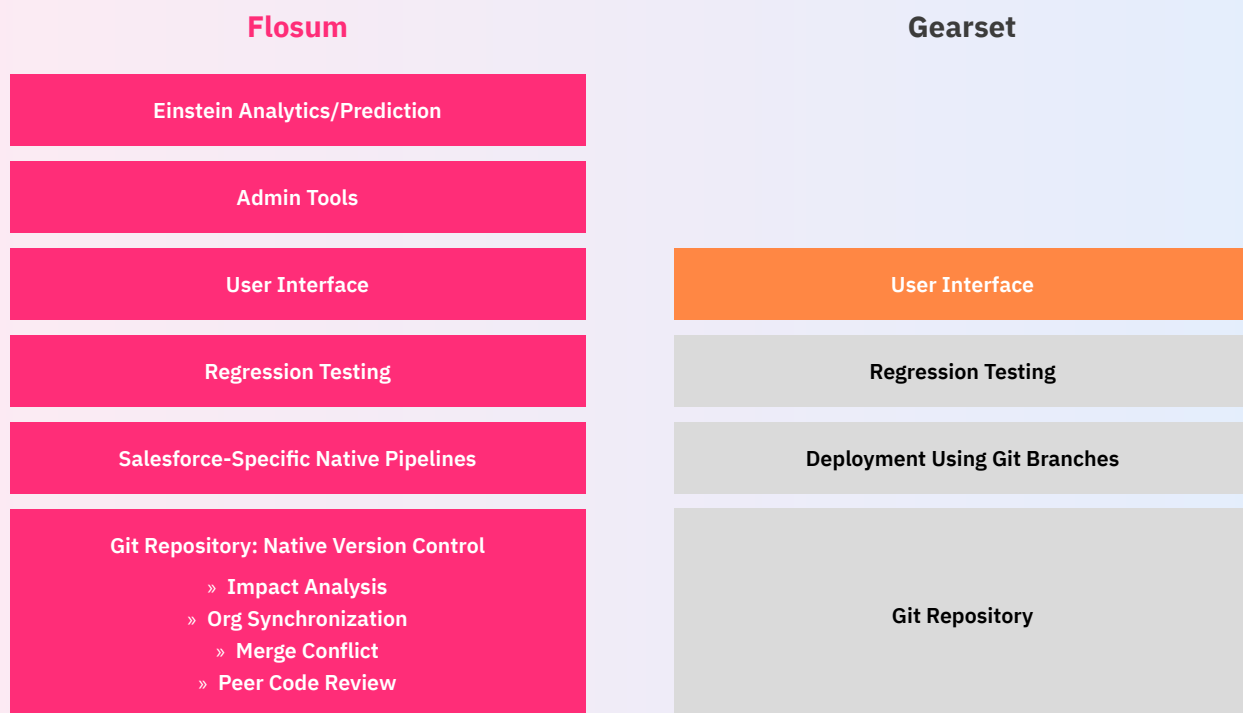
Flosum vs Gearset

There are numerous factors making Flosum different from Gearset. In this document, we will highlight said differences so you gain a thorough understanding of our solution and can better compare against Gearset.

The Big Picture

While Gearset sometimes works well for small teams in certain geographies, it's simply not a fit for larger teams that handle complex projects requiring parallel development. And, with its security disadvantages and Git-based limitations, teams will be happy to graduate to a more sophisticated product like Flosum.

Architecture Differences: Flosum vs Gearset



Platform

Flosum is the only release management solution built entirely on the Salesforce platform. Flosum runs in a Salesforce production org provided with your Flosum license that is part of your own Salesforce instance, subject to the same security processes and standards as the rest of your Salesforce environment.

That means data never leaves YOUR Salesforce platform, no third-party servers are ever used, and no one from Flosum ever has access to your orgs. Everything is 100%, completely secure.

Gearset, in contrast, is hosted on third-party servers where application logic, deployment processing, data migration and other functions occur.

Gearset must extract the customer's information – including their code, metadata and data – out of the Salesforce org and into these third-party platforms (outside of your IT boundary) for processing. Many security-conscious organizations are not comfortable allowing this level of access to their source code and information. It creates a “backdoor” to an organization's Salesforce security and introduces numerous security vulnerabilities.

More Secure

Unlike other providers, Flosum never removes your data and application source code from your IT boundary – it always remains under your security standards and processes.

Technology

While Flosum is accessed via a friendly Lightning-based UI or familiar IDEs including VS Code, Gearset's challenging interface is confusing for users – especially citizen developers and Salesforce admins. That's because Gearset is reliant on GIT, forcing users to learn an additional set of tools.

Due to the complexity of Gearset's structure, it creates problems for ordinary admins and even some skilled developers to properly utilize.

Flosum uses the Lightning Platform, delivering a significantly more streamlined and intuitive user experience.

Easier to Use

Gearset's GIT-based infrastructure is cumbersome and difficult to use for Salesforce developers, admins and release managers. Flosum, however, was built specifically for the Salesforce interface, so it is very familiar and simple to navigate.



Flosum is the only release management solution built entirely on the Salesforce platform making it the most secure solution available.



Complicated User Interface

Gearset's challenging interface is confusing for users – especially citizen developers and Salesforce admins.



Security issues to consider



Are there Back Doors through your Firewalls?

Most customers don't realize that because Gearset requires access to your org from its third-party servers, it effectively creates a "backdoor" access to customer information and applications in your Salesforce environment. Since Gearset also handles data migration, they not only have possession of code and metadata, but actual data as well.



Who has access to Data?

The personnel at Salesforce who have access to any of their customer's data all reside within the U.S. and are screened extremely diligently by Salesforce. Data center employees must complete an extensive background check including mandatory FBI, drug and criminal screening every six months. By using Flosum, a company ensures their data never leaves the Salesforce platform, so users can rest assured their data remains in safe hands.

Third-party platforms, like the ones on which Gearset runs, cannot guarantee the same standards, and they are often offshore, which in turn presents a large security risk.



Can Your Organization use Gearset?

Many countries such as Canada and Singapore cannot use third-party servers because, by law, data is not allowed to leave their country. Third-party servers also cannot be used by government, financial or healthcare institutions because of the highly sensitive nature of their data. Gearset is only hosted in the United Kingdom, Australia, and the United States, so a country with data residency legislation will run into challenges if they are outside of these locations. This reality can be very limited for companies.

Flosum, on the other hand, has customers in both Singapore and Canada, and numerous clients in the financial, healthcare, and government sectors. In fact, Flosum is the only release management solution approved for use on the Salesforce Government Cloud.



Flosum can run in any country where you have Salesforce and is the only solution of its type permitted to run in Salesforce Government Cloud.



With Flosum, data never leaves the Salesforce Platform, and users can rest assured their data remains in safe hands.



Security issues to consider



Is the Salesforce Security Guarantee Present?

Gearset is not native to Salesforce, which means that Salesforce does not perform a security review on non-native application functionality running outside of the Salesforce platform. And since Gearset runs primarily on third-party servers, Salesforce will not guarantee it meets their security standards.

In contrast, Salesforce has completely reviewed the Flosum app (because it sits entirely on the Salesforce platform) with Flosum, you have Salesforce's own security standards and practices protecting your data and source code.



Who Guarantees the Service Level?

Because Flosum runs entirely within Salesforce, Salesforce is responsible for all server maintenance and upkeep.

However, Gearset requires their own set of SLA's since their personnel are responsible for keeping services working outside of Salesforce on the third-party servers, while also depending on these servers server maintenance and upkeep. Not only is that an extra logistical step, it's a risk for business continuity.



What Compliance Certifications are Possible?

Most customers spend a lot of time ensuring their platforms adhere to compliance needs. Some customers must also adhere to industry regulations while others operate in a very regulated ecosystem. Salesforce is a platform that takes compliance seriously, consistently meeting and exceeding these laws and regulations. For example, the platform is 100% compliant with HIPAA regulations needed within the Healthcare and LifeSciences sector, as well as industry regulations and privacy laws required for the financial services vertical. Salesforce is also inherently compliant with all federal certifications.

Because Flosum resides completely within Salesforce, it can provide every single one of the certifications that Salesforce can, including HIPAA, PCI, ISO 27001, 21 CFR Part 11, 210, 211, 820, Annex 11, GAMP, and CGMP.

However, because Gearset is built on third-party servers, it does not carry these certifications. This is a crucial point for highly-regulated industries to be aware of as they keep compliance at the forefront of the evaluation process.



**Flosum is the #1 RATED app
on the Salesforce Appexchange.**

Features that Users Need

Gearset is fundamentally a basic product and lacks many features and functions. The features that Gearset does have rely heavily on Git for many of its functions, and while Git is the common tool for development, it requires extensive user knowledge of development. It was also not originally intended for Salesforce development's quirks and differences, which makes handling many of Salesforce's unique needs complicated, manual, and sometimes impossible.

Flosum, however, was purposefully built for Salesforce. It's fully integrated and may be used in conjunction with or without Git, making numerous processes easier while eliminating many Git-based pain points. Below are some areas where this comes into play with Gearset vs. Flosum.



Merge Logic

Flosum has its own merge conflict tool built specifically for Salesforce. Its ability to quickly analyze code and unique Salesforce components, provide side-by-side comparisons, and easily merge conflicts for all components (including Lightning) makes this process a breeze. For more information on Flosum's Merge Conflict functionality check out our [video](#).

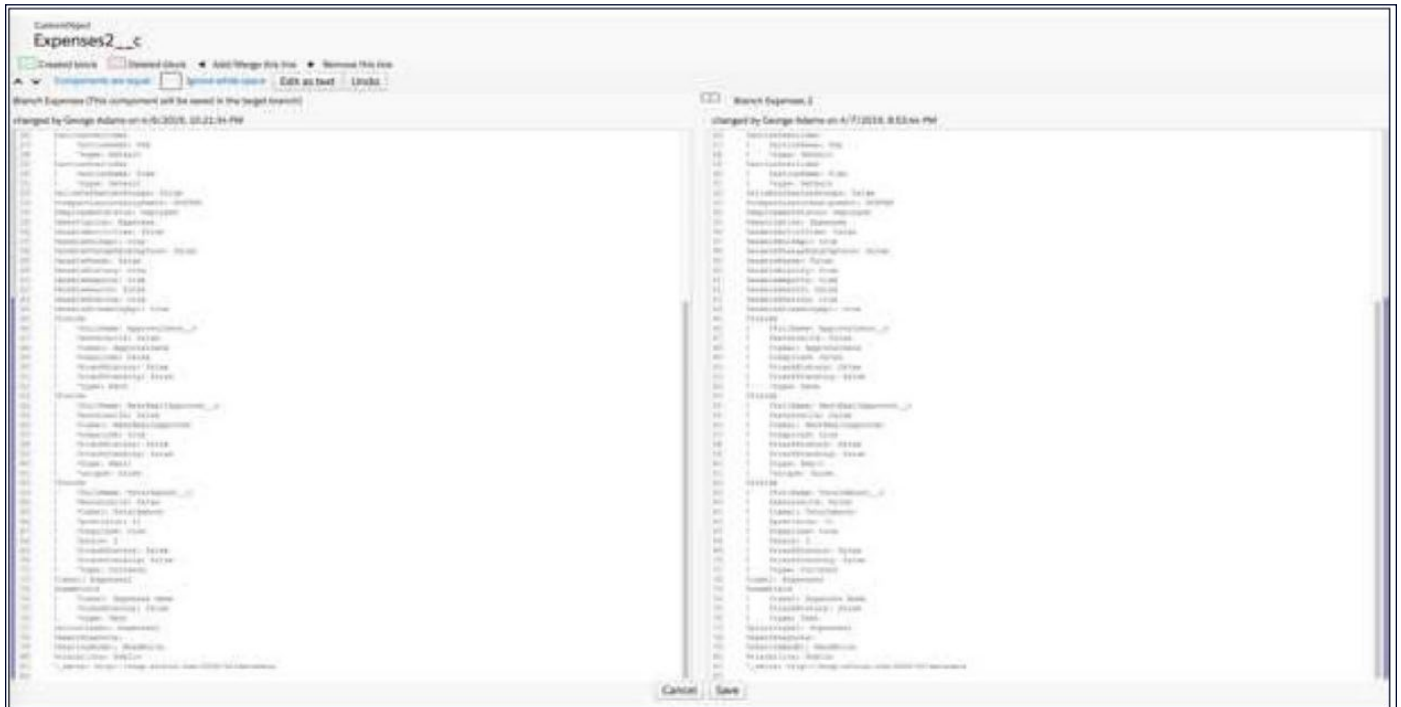
Gearset, on the other hand, has an extremely complicated conflict process relying heavily on Git for resolutions. Not only is this strategy extremely complicated, the algorithms utilized to decide which method will resolve a conflict are inconsistent, so the way a merge is handled becomes almost random.

In addition, this process relies completely on the user to manually decode the process and check through false errors or conflicts that the tool was unable to decipher. The time-intensive process can take hours or even days. In the example below we have a Git-based merge. See if you can quickly decipher how many merge conflicts there are.

```
54 54 <fullName>ApprovalHistory</fullName>
55 55 <fullName>ApprovalHistory</fullName>
56 56 <fields>
57 -- <fullName>ApprovalHistory</fullName>
57 57 * <fullName>ApprovalHistory</fullName>
58 58 <fullName>ApprovalHistory</fullName>
59 59 <label>ApprovalHistory</label>
60 -- <required>False</required>
61 * <label>ApprovalHistory</label>
62 * <fullName>ApprovalHistory</fullName>
63 * <required>True</required>
64 * <fullName>ApprovalHistory</fullName>
65 65 <trackHistory>False</trackHistory>
66 66 <trackHistory>False</trackHistory>
67 -- <type>Date</type>
68 * <type>Date</type>
69 69 <fields>
70 70 <fullName>NextEmailApprovalHistory</fullName>
71 --
72 72 <fullName>ApprovalHistory</fullName>
73 73 <fields>
74 74 <fields>
75 75 <fields>
76 -- <fullName>ApprovalHistory</fullName>
77 * <fullName>ApprovalHistory</fullName>
78 78 <fullName>ApprovalHistory</fullName>
79 -- <label>ApprovalHistory</label>
80 -- <required>True</required>
81 -- <fullName>ApprovalHistory</fullName>
82 * <label>ApprovalHistory</label>
83 * <required>False</required>
84 84 <trackHistory>False</trackHistory>
85 85 <trackHistory>False</trackHistory>
86 -- <type>Date</type>
87 * <type>Date</type>
88 88 <fields>
89 89 <label>ExpenseItem</label>
90 90 <fullName>ExpenseItem</fullName>
91 91 <label>ExpenseItem</label>
```

If you said there were 9, 12 or even 14 conflicts – that is not correct! In actuality, there are no conflicts! The developer simply changed the order of components, which caused Git to show them as conflicting, even though they still exist on different lines.

In contrast, Flosum’s technology makes merging a breeze. Our auto-resolve tool easily compares all of the code, regardless of order or placement. Here is the result of that same merge in Flosum.



As you can see, Flosum determined no conflicts occurred. If there were conflicts however, we would easily go through and resolve each one by pulling code directly into Flosum, rather than having to do the changes manually in each org like we would when using Gearset’s Git-based logic.



Appropriate for Salesforce Admins

Flosum was purposefully built with a **“clicks-not-code” philosophy** so that it can be easily used by any user that is already familiar with Salesforce.

Gearset, however, heavily relies on Git which requires extensive developer knowledge. Therefore, it becomes inaccessible to admins who lack an extensive development background. For example, the merge of conflicts could not be handled by most admins, leaving a large chunk of the development team in the dark. Furthermore, because the Gearset platform is so complex, it adds another layer of expertise needed to utilize their services.



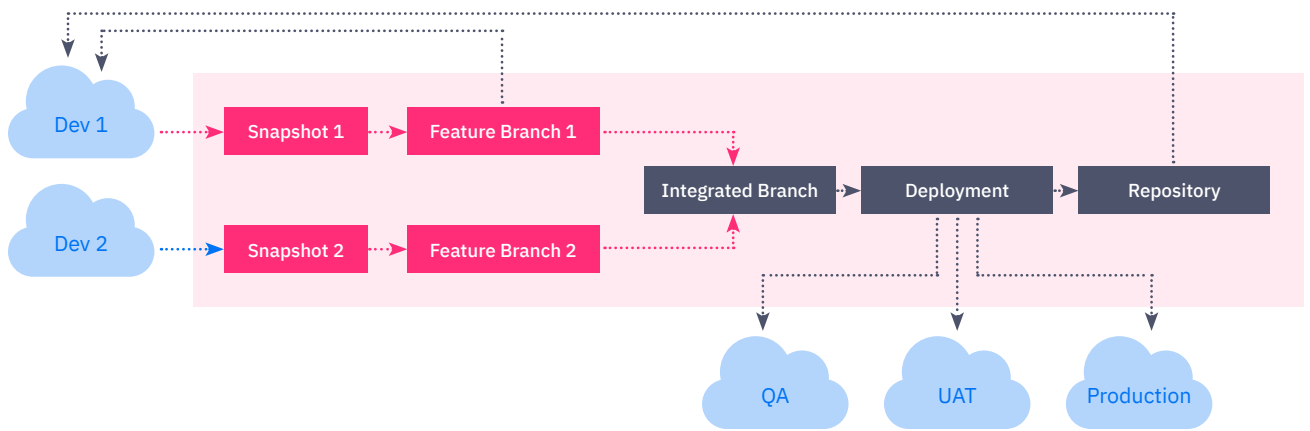
While Gearset claims to be “native”, a quick check on the Salesforce Appexchange will show they do not carry the Salesforce badge for native applications.



Simple vs. Complex Branching Strategy

Gearset branching is rooted in Gitflow. In most cases, in order to shift changes through a developer's sandbox from QA to UAT to Production, branch merging and sandbox merging must occur at each of the branch and sandbox levels. And with environment branches, you must have synchronization with orgs or may end up with clogged pipelines when a single user commits changes with issues. Multiply this by the large number of developers making changes, and it equates to huge headaches and a lot of wasted time just to manage the Gitflow branching environment.

In contrast, check out Flosum's branching strategy below:



Here, we have a code merge occurring at the integration branch level, which can be used to create a deployment and effortlessly move changes from QA to UAT to production. Flosum's Impact Analysis feature compares the code from each target branch directly with the code from each target org, and therefore eliminates the need for the QA branch, UAT branch and Production Branch. Furthermore, Flosum's technology ensures you never overwrite any code when shifting it from org to org. The result is 30% of developer time saved, and no headaches to deal with.



Flosum's technology ensures you never overwrite any code when shifting it from org to org.



Git Version Control for Salesforce is NOT Enough

Because developers are tied to the Git Version Control System, the process of keeping branches and teams in sync is complex and requires multiple steps.

The Gearset automatic merge function is very inaccurate and routinely creates errors. That means developers must perform a validation and manually check each merge before finally merging branches.

First, users must perform a validation, then check for merges. While Gearset possesses an automatic merge function, it is very inaccurate and routinely creates errors, so developers must return and manually check each one. Finally they must merge branches. Additionally, Git-based version control cannot handle declarative merge components or any of the complex components Salesforce utilizes, such as lightning components, static resources and aura definition bundles. While Flosum integrates fully with Git, it also offers a native Version Control System which is far easier to use, handles all Salesforce component types and saves developers significant amounts of time.

For more information about our Native Version Control System click [here](#).



Extensibility and Customization

Because Gearset is a basic platform, any customization required is difficult and costly, if even possible. In contrast, since Flosum is built entirely on Salesforce, it's extendable and provides any possible Salesforce customization.



Data Migration

While Gearset offers a data migration tool, but it has the same security issues described above. When data is migrated, it is extracted from your Salesforce environment directly into Gearset's infrastructure hosted on third-party servers, which can negate your standards and processes for security compliance.

Flosum's data migration tool, however, directly moves relational data between orgs without ever leaving the Salesforce platform.

For more information on our Data Migration solution click [here](#).



Flosum is more secure, easier for Salesforce teams to use, and has features to address the unique challenges of DevOps for Salesforce

In this article we have demonstrated some of the key factors that differentiate Flosum from Gearset. For further information, please reach out to us at hello@flosum.com