

4 STEPS TO IMPROVE YOUR SALESFORCE HOTFIX STRATEGY WITH PRODLY APPOPS

Develop patches faster, easier,
and with less risk



Table of Contents

3 Introduction

4 What Is a Salesforce Hotfix?

5 When Do You Need a Hotfix?

6 Salesforce Hotfix Workflow: Inefficient and Costly

7 Never Make a Hotfix Directly in Production!

8 Create a Hotfix Strategy Before the Next Bug

9 Leverage Prodly AppOps to Improve Your Hotfix Strategy in 4 Steps

12 Easily Make Salesforce Hotfixes with Prodly AppOps



Introduction

This quote illustrates perfectly the approach many successful Salesforce architects, developers, release managers, and admins take when it comes to dealing with bugs.

With the agile approach that enables fast release cycles and drives a very high ROI from Salesforce, what matters is getting a feature out quickly at an acceptable level of functionality. Then afterwards, you can budget in time to troubleshoot and—if needed—make patches.

However, many devs and admins who follow governance and release best practices find making hotfixes in Salesforce challenging. In this eBook, we'll describe what a Salesforce hotfix is, when you need one, and why the Salesforce hotfix workflow is inefficient. We'll also explain the value of creating a prod hotfix strategy before the next bug—plus, we'll discuss four ways Prodlly AppOps can improve your hotfix strategy.

“It’s not the prevention of bugs but the recovery—the ability to gracefully exterminate them—that counts.”

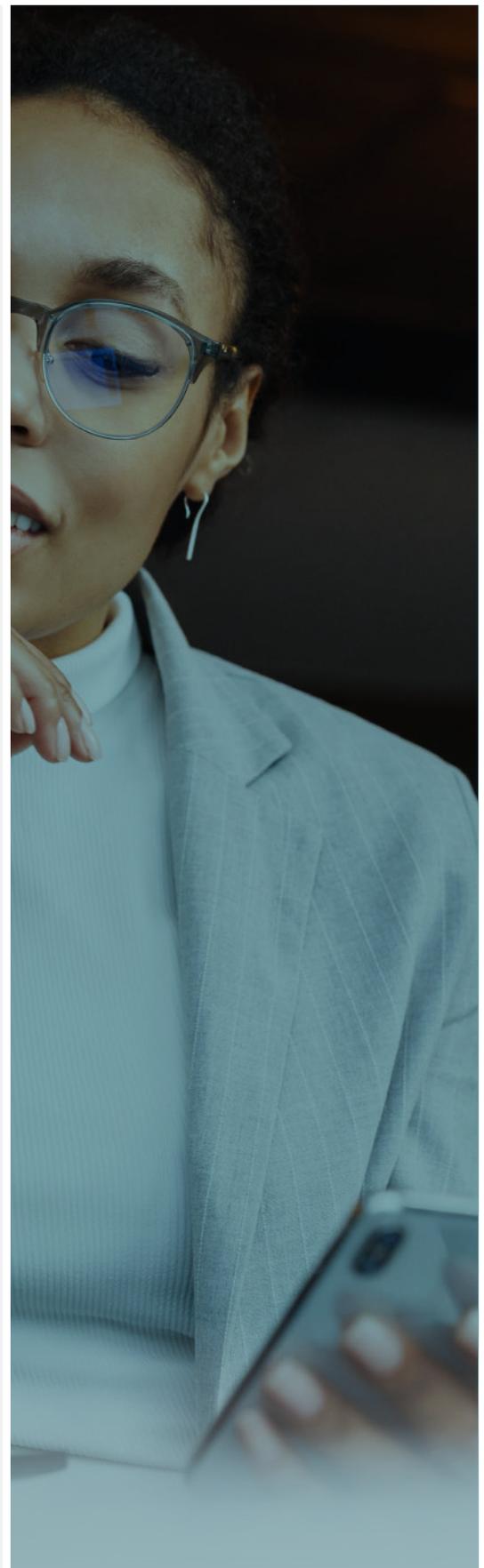
- Victoria Livschitz,
Principal Architect,
Sun Microsystems



What Is a Salesforce Hotfix?

In Salesforce, a hotfix is a simple configuration change that addresses a key issue impacting day to day business. Oftentimes, this patch is required due to a bug a user reports when using Salesforce—so you need to fix it immediately.

It's key to understand that although a hotfix may not involve a major change to your production environment, it's business critical and therefore urgent. Because it addresses a bug that's adversely affecting your business operations, it can't be postponed until your next major release. That's why you need the ability to implement a patch as quickly and smoothly as possible without getting bogged down in error-prone manual processes that delay the process.



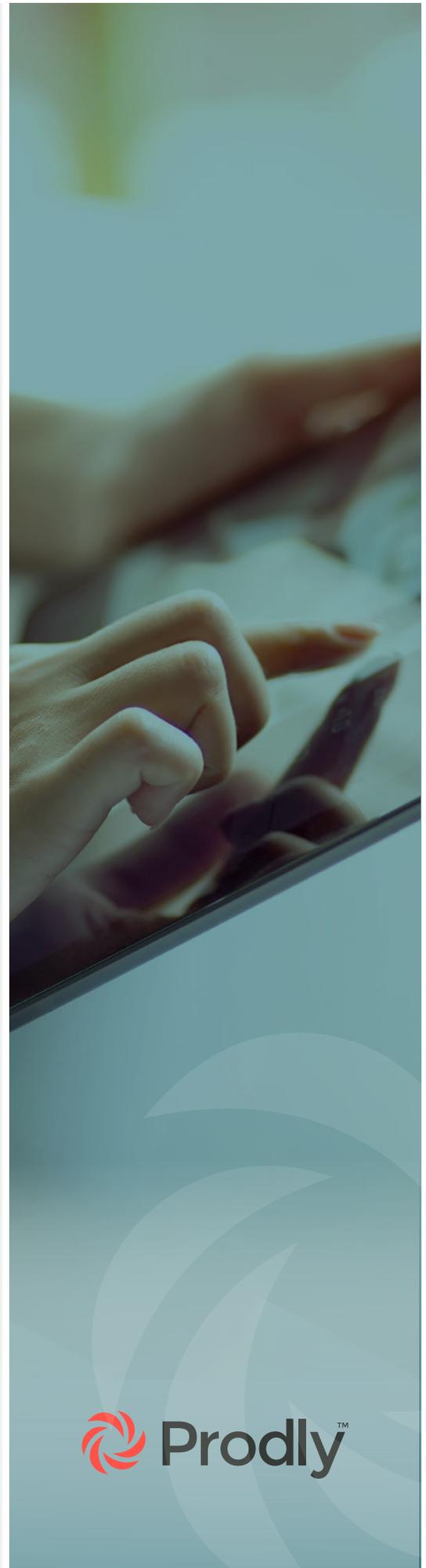
When Do You Need a Hotfix?

Consider the following true story:

A leading luxury brand sells attire and accessories via its e-commerce portal. But one day, the unthinkable happens: Due to a glitch, customers receive a 100 percent discount on specific high-priced items once they add them to their shopping carts. Within a couple of hours, the business discovers the issue. While its IT team scrambles to fix the issue, the company is forced to send out apologetic emails canceling the orders.

This nightmare scenario actually happened to the online store of Marc Jacobs in January 2022. Although the company was able to minimize the financial impact of the event, its decision to cancel the orders instead of honoring them resulted in a customer experience that was subpar—and that likely had a negative impact on consumers' perception of the brand.

Although we don't know if the aforementioned business used Salesforce, this story illustrates that if you struggle to configure and deploy hotfixes in a timely manner, you're not alone. And when it comes to creating a hotfix environment in Salesforce, traditional data set building is slow, laborious, and prone to error—even with Data Loader.

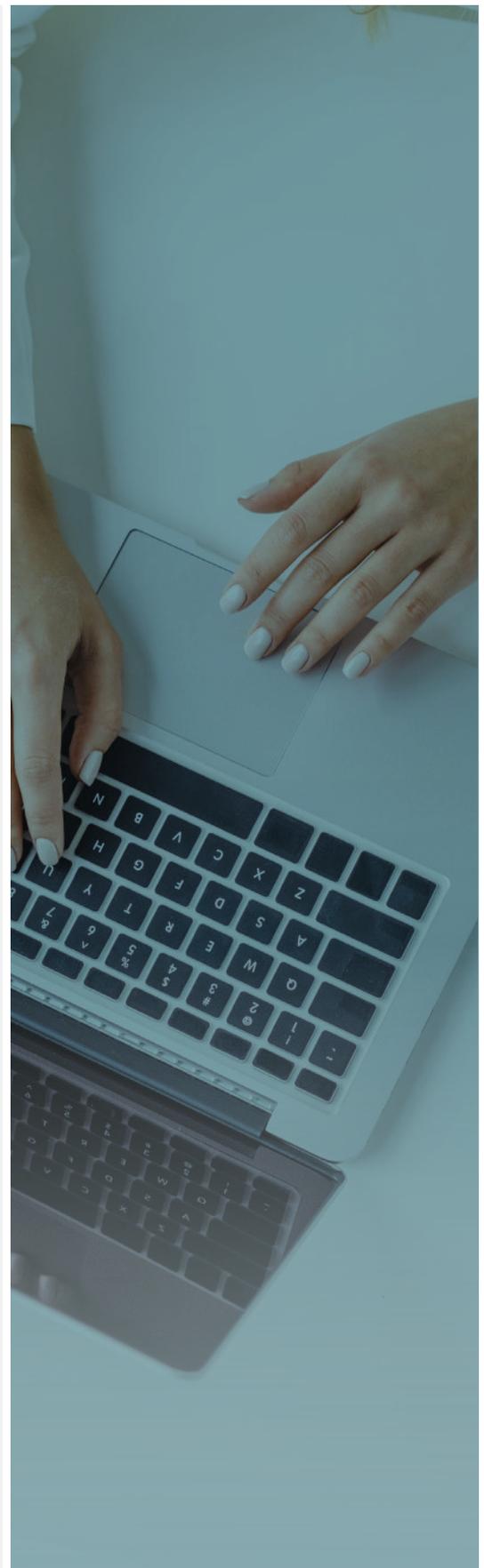


Salesforce Hotfix Workflow: Inefficient and Costly

If you're implementing a patch in Salesforce, the process probably looks something like this:

- Purchase a Partial Copy sandbox or Full Copy sandbox to use as a hotfix environment—at 20 percent or 30 percent of net spend.
- In the hotfix environment, determine what's causing the issue. Then fix the issue and test it.
- Use change sets and Data Loader to promote the approved change to your production org—i.e. create an outbound change set in the hotfix environment, create an inbound change set in the prod org, validate the inbound change set, and deploy the change set.
- Use change sets and Data Loader to back promote the change to your other sandboxes.

Unfortunately, this strategy can be cost-prohibitive due to the expense of purchasing a sandbox to use as a dedicated hotfix environment. Moreover, the multiple manual steps required for each change set deployment make this process highly labor intensive and time consuming.



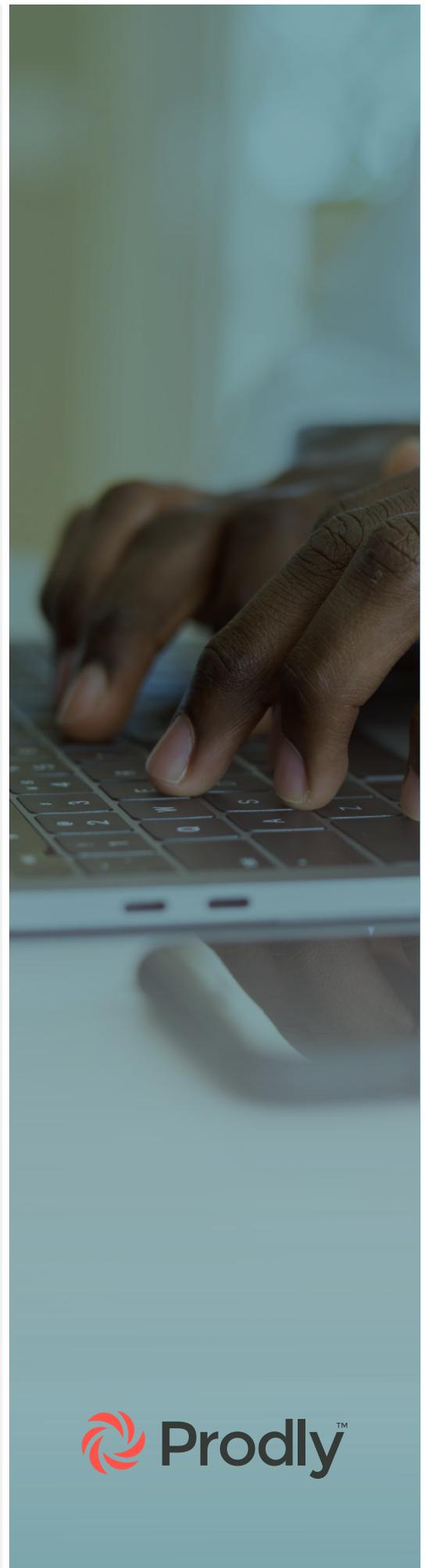
Never Make a Hotfix Directly in Production!

Knowing how expensive and inefficient it is to implement a hotfix in Salesforce, you might be tempted to rush things and make the patch directly in production.

A word to the wise, however:

Never, ever, make a hotfix directly in your prod org!

Why? Because if the patch introduces any new issues, you'll be stuck with a much bigger headache. And that could affect your users' experience even more!



Create a Prod Hotfix Strategy Before the Next Bug

To be prepared to make a mission-critical change in a timely manner, don't wait until a serious issue arises to design a hotfix workflow. Instead, get ahead of things and establish a production hotfix strategy before the next bug occurs. Your strategy should:

- Appoint team members to configure the hotfix
- Designate an environment where you'll make the change and test the fix
- Specify how you'll promote the fix into production
- Determine which other orgs will need the patch and how you'll back promote it
- Include a post mortem to analyze the event and prevent future issues



Leverage Prodlly AppOps to Improve Your Hotfix Strategy in 4 Steps

Now you have a basic game plan for dealing with hotfixes, let's take a look at how Prodlly can improve your strategy. Prodlly AppOps lets you automate your hotfix workflow and rapidly fix, test, and promote essential changes to production using the following steps.

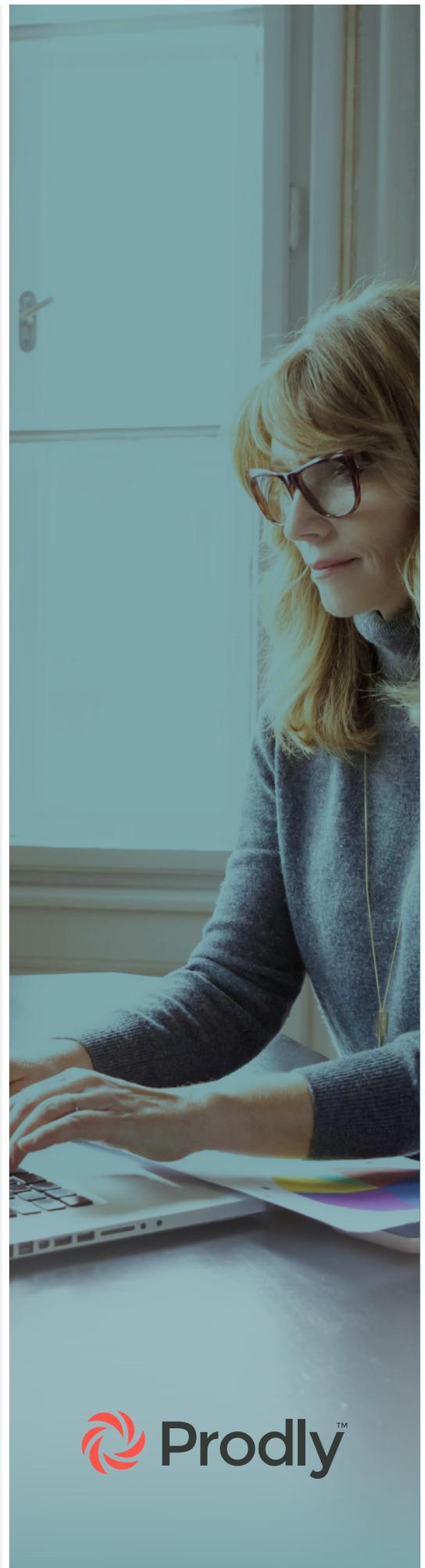
1 Use a Free Environment for Dedicated Production Support

With Prodlly AppOps, you don't need to purchase a new Partial Copy or Full Copy sandbox as a dedicated hotfix environment. You can simply use a free Developer sandbox or Scratch org, because AppOps will let you easily and quickly move data into it. This saves your Partial Copy or Full Copy sandboxes for higher value projects in Salesforce.

2 Seed the Impacted Configuration to Your Hotfix Environment

Your hotfix environment needs to be a reproduction of your production org—but you don't have time to recreate all the data from the prod org in the sandbox.

In Prodlly AppOps, you can compare your orgs to see any metadata differences between your prod org and your hotfix environment. If there are any errors, you can easily review and resolve them to sync your hotfix environment.



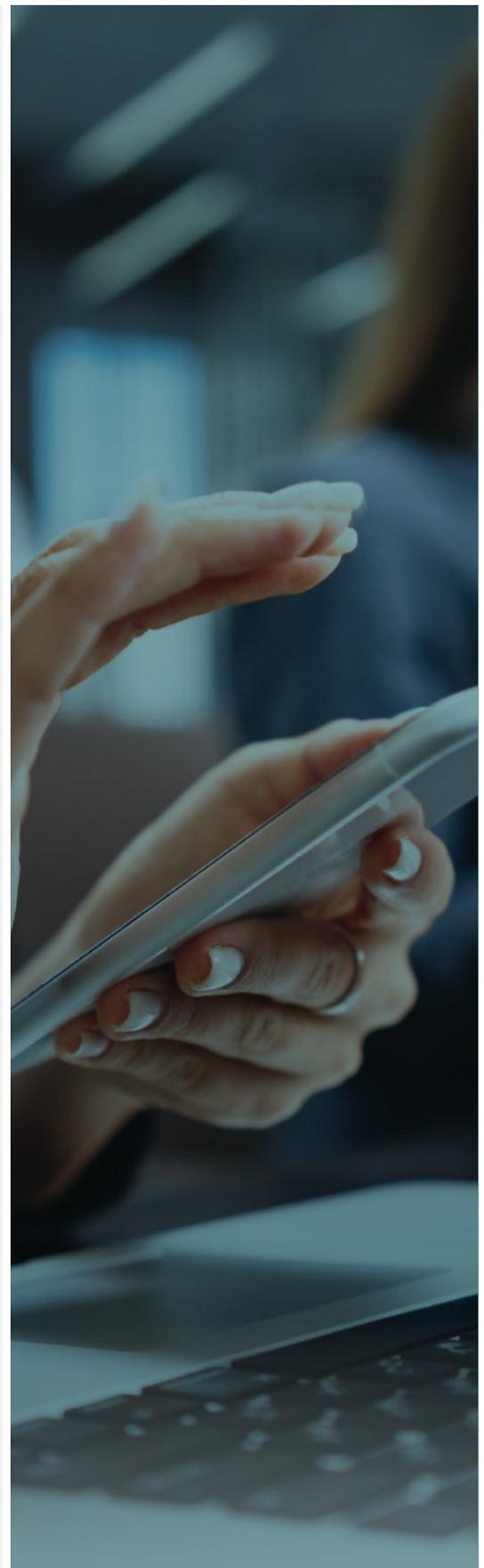
You can use sandbox seeding to automatically migrate the affected data to the hotfix environment. The seed sandbox feature allows you to quickly select data and upsert it to your hotfix environment. Simply select a root object in your prod org, along with any related objects. Next, temporarily deactivate all events for the duration of the deployment to avoid complications. Finally, implement the deployment.

Note that AppOps even provides prebuilt templates for common data seeding needs that can save you even more time.

3 Promote the Hotfix Back to Production

With your hotfix environment set up, you can determine what the problem is, configure the required change, and test the change without any risk to your prod org. Once you're satisfied the change works, AppOps makes it easy to seamlessly move the change back into your production org—regardless of whether the fix involves metadata or data.

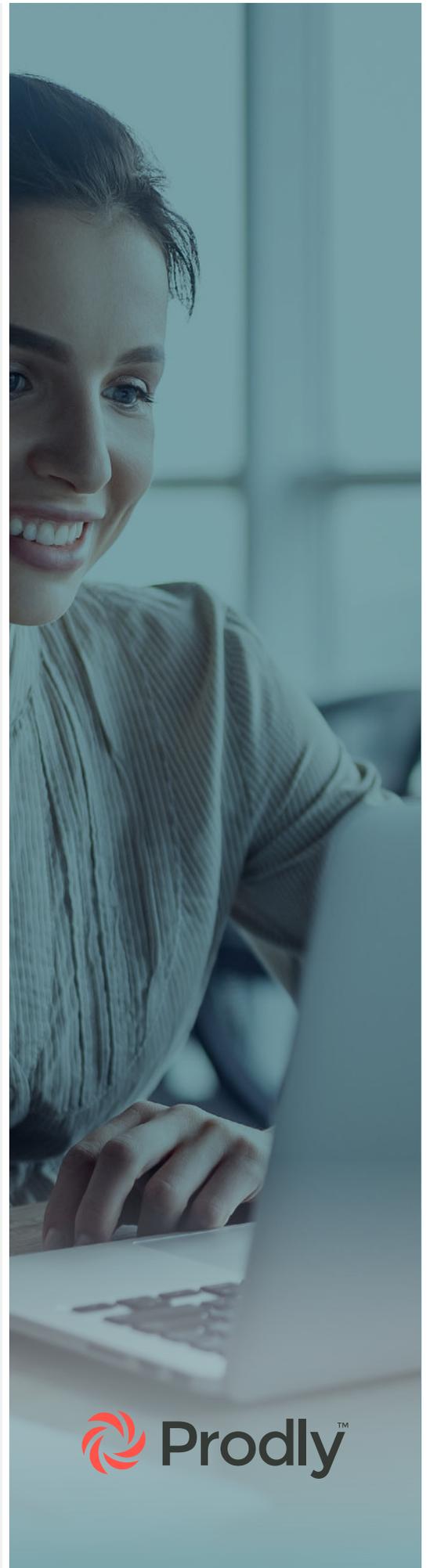
Instead of having to use change sets and Data Loader to manually push the change to your prod org, you can leverage Prodlly AppOps to automatically migrate the metadata and data while still maintaining complex relationships. Simply select the configurations you want to deploy, and then deploy all the metadata and data changes at once.



4 Sync the Hotfix to the Rest of the Orgs in Your Stack

While you've been rushing to fix the bug, your teammates may have been working on the next release. It's essential to sync the rest of the orgs in the release path with your production org and training environment so they don't reintroduce the bug or cause new ones.

Prodly AppOps allows you to sync data to up to five orgs simultaneously. So once you've promoted the hotfix back to production, you can quickly and easily sync the metadata and data changes from production to the other orgs in the stack. The benefit is that by doing this, you'll bring your other sandboxes up to date without having to refresh them.



Easily Make Salesforce Hotfixes With Prodlly AppOps

Let's face it: Bugs happen. And when they impact your day-to-day business operations, you need to fix them—fast. With a robust hotfix strategy and Prodlly AppOps, you can quickly, efficiently, and easily create a hotfix environment, fix the issue, and promote the change to your production environment and other orgs. And you can do all this without the high costs of purchasing an additional Partial Copy or Full Copy sandbox in Salesforce.

To learn more about Prodlly AppOps, contact us today at www.prodly.co.

Sources:

Atlassian Tutorials: Gitflow Workflow

Salesforce Ben: What Is a Salesforce Sandbox?

Salesforce presentation: Release Management 2_25

