

# THE ULTIMATE SALESFORCE CPQ IMPLEMENTATION GUIDE

Discover the Way to Salesforce  
Quote-to-Cash Success!



# Table of Contents

---

3 Foreword

---

4 Introduction

---

5 Part I: The Benefits of Implementing Salesforce CPQ

---

6 Part II: How to Evaluate Implementation Partners

---

7 Part III: How to Plan Your Salesforce CPQ Implementation

---

10 Part IV: The 8 Project Domains of a Salesforce CPQ Implementation

---

15 Part V: Time to Implement

---

18 Part VI: Go Live

---

20 Part VII: A Final Word About Change Management in Your CPQ Implementation

---



## Foreword

For your business, modernizing the quoting process is a critical step in reimagining how you engage with customers. As such, implementing Salesforce CPQ is essential to your digital transformation. It's the foundation for providing superior purchasing experiences to your customers, making better decisions throughout your business, and using Pricing & Packaging (P&P) strategy as a true differentiator to win more customers.

It facilitates the work of your sales team, making them more efficient and allowing them to focus on delivering value for your customers.

In addition, it puts your business in the best position to meet evolving business and market requirements and can significantly increase your competitive advantage.

This guide contains all the information you need to successfully implement Salesforce CPQ. And if you follow the steps and best practices outlined in the following chapters, you'll set yourself up for long-term quote-to-cash success.



A handwritten signature in black ink, appearing to read 'Max Rudman'.

Max Rudman  
*Founder of SteelBrick, now Salesforce CPQ*  
*CEO of Prodlly*

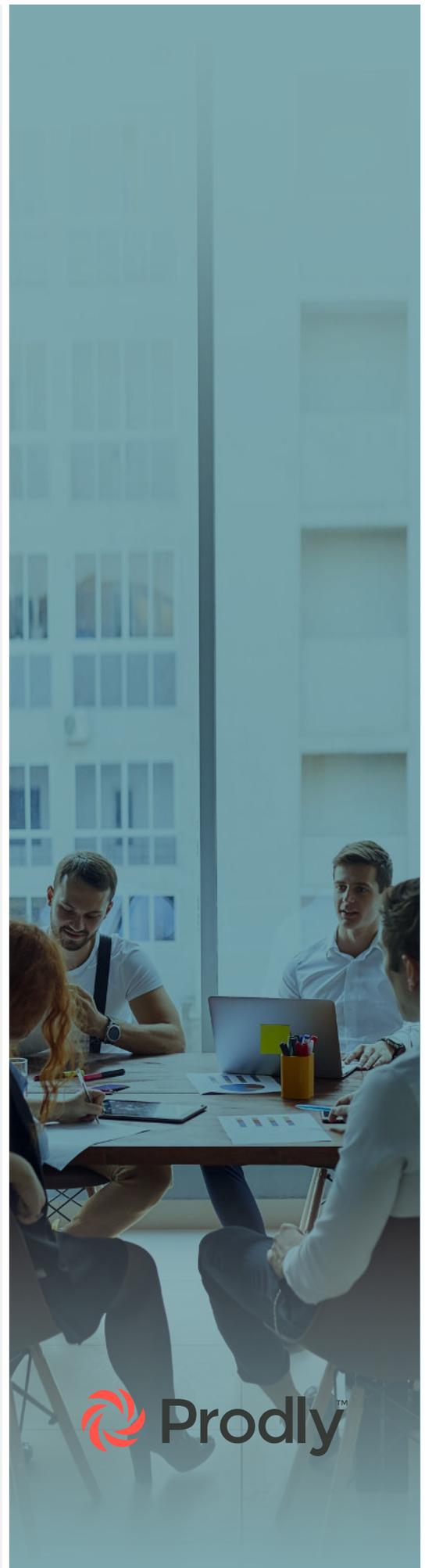


## Introduction

Is your company implementing digital transformation in Salesforce? Then moving from a legacy system to Salesforce CPQ is critical to your ability to quickly, easily, and accurately create quotes.

In this guide, we'll describe the benefits of implementing Salesforce CPQ. We'll examine how to plan for and manage the implementation in the most efficient, streamlined manner possible.

We'll explain that Salesforce CPQ is heavily based on configuration data. We'll examine the impact this has on implementation. Plus, we'll discuss how to greatly accelerate implementation timelines with a next-gen DevOps tool like Prodlly to manage your data.



# Part I: The Benefits of Implementing Salesforce CPQ

Is your sales team tracking leads and opportunities in spreadsheets or homegrown systems to generate pricing for quotes? This is labor intensive, prone to error, time consuming, and inefficient. In addition, it's a difficult or even impossible system to scale as your business grows.

In contrast, quotes produced with Salesforce CPQ are automated using preprogrammed rules. This ensures accurate pricing that takes a wide range of factors into account. These factors include account quantities, customizations, discounts, optional product features, incompatibilities, and multiple revenue models.

---

ACCORDING TO [SALESFORCE](#), USING CPQ SOFTWARE RESULTS IN:

- 10 TIMES FASTER QUOTE GENERATION
  - 95 PERCENT REDUCTION IN APPROVAL TIME
  - 2 TIMES FASTER MOVING FROM QUOTE TO CASH
  - 30 PERCENT QUICKER RAMP UP FOR NEW REPS
- 

Clearly, an automated configure, price, quote tool can, when implemented correctly, significantly streamline your sales process and boost your revenue.



## Part II: How to Evaluate Implementation Partners

Since CPQ will be the lifeblood of revenue operations, many companies use an implementation partner to make sure it's implemented correctly and quickly. When evaluating implementation partners, consider several factors.

Select an implementation partner or system integrator (SI) with downstream integration and process expertise. Look at references and reviews from companies like yours. Select an SI who's certified and a proven expert in Salesforce CPQ.

Examine how your implementation partner is connected. What other applications do they have expertise implementing? How does this affect your implementation?

Evaluate the methodology of your SI. The best method is a hybrid approach that consists of analysis and design, iterative build, and testing and deployment.

Consider the price of both Salesforce CPQ and your partner. However, think of more than just the financial aspects. Salesforce CPQ is a dynamic, long-term investment, so you want to pick wisely from the start.



# Part III: How to Plan Your Salesforce CPQ Implementation

## Change Management

You've invested in Salesforce CPQ to improve both the user and the customer experience by streamlining the selection and presentation of products. Naturally, this involves many changes to how your reps currently prepare quotes, as well as to the products they're quoting.

Therefore, planning your CPQ implementation starts with planning change management. Change management involves people, processes, and technology.

That means you want to get everyone aligned around this endeavor. You also want to make sure you have the right processes in place. And you'll need a robust release management tool that takes the complexity out of change management while still offering agility and flexibility.

---

## Assemble Stakeholders for the Implementation

It's key to the planning phase to get your team ready early on in the process. Identify your executive sponsor, assign relevant sales and product experts, and select end users for testing. You'll need input from Sales, Product Management, Finance, and sometimes even Marketing.

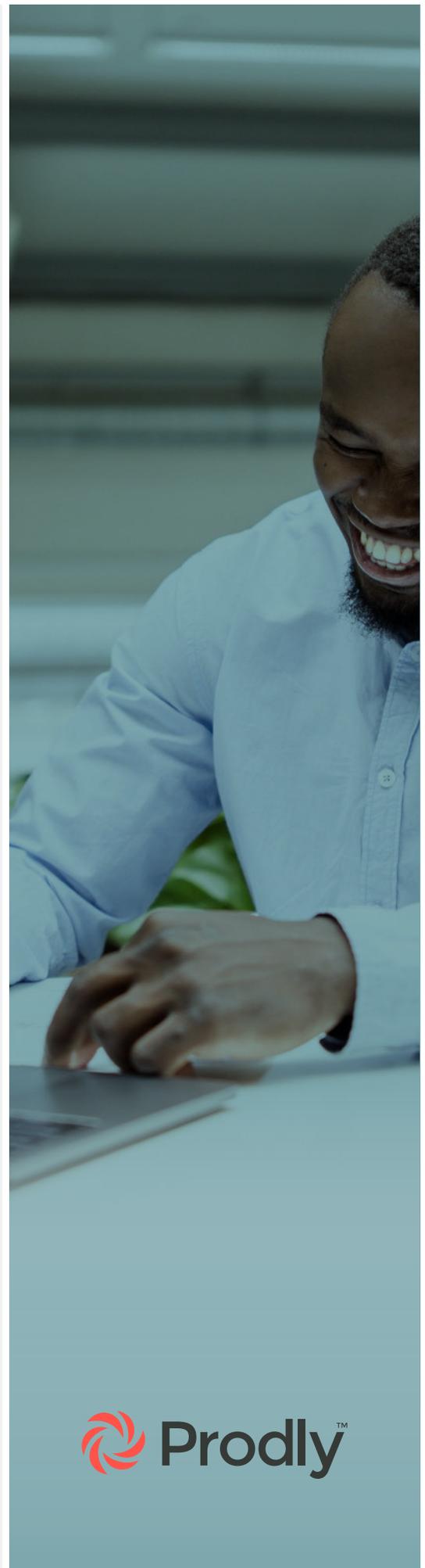
Remember other important areas of the CPQ implementation when identifying your SMEs. These include legacy data migrations, integrations, fulfillment, billing, revenue recognition, and commission calculations.

---

## Work Agilely

Even though CPQ is implemented through "clicks, not code," the [principles of agile development](#) still very much apply, especially:

- Early and continuous delivery of valuable software



- Business users and developers work together
- Embrace change
- Frequent delivery
- Working software
- Simplicity

When planning your CPQ implementation, don't try to do everything at once. Implement an agile process. Break your project into phases, and keep your first phase simple.

Work from a foundation where you're getting 70 to 90 percent of your needs met from day one. Then work through the remaining 30 to 10 percent as you maintain and improve the system. This will help you keep executive and user buy-in.

---

**FOCUS ON THE BIG WINS AROUND COMMON USER STORIES TO RESOLVE PAIN POINTS AND AUTOMATE YOUR PROCESSES. YOU WANT TO DELIVER QUICK ROI TO KEEP EXECUTIVE BUY-IN AND MOTIVATE USER ADOPTION.**

---

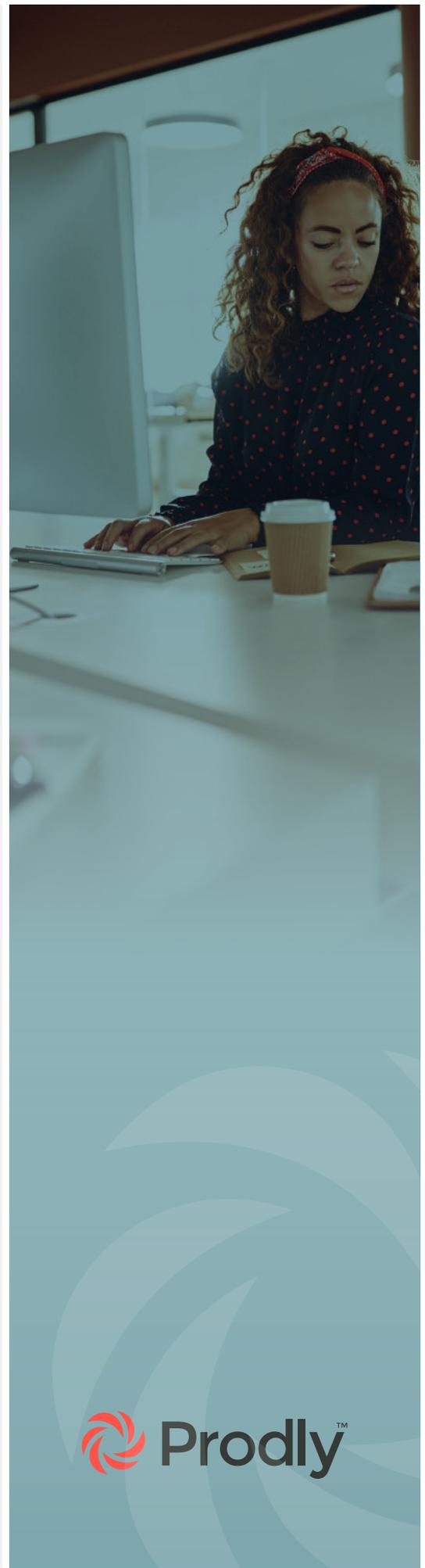
### Understand What Tools You'll Need

The majority of Salesforce CPQ's setup relies on the use of record data or [configuration data](#) to achieve specific functionality. Configuration data acts like metadata in that it describes how the system behaves. However, it's stored as record data in Salesforce's databases and therefore cannot be deployed to production via change sets or Salesforce DevOps Center.

Adding another layer of complexity, CPQ's configuration data consists of more than 50 related Salesforce objects that must be deployed in exact order to avoid breaking critical relationships.

You can use Data Loader to export, organize, and import the dozens of objects in CPQ while maintaining all their references to one another. However, this is labor intensive, time consuming, and highly prone to error.

We'll discuss release management options later in this guide. However, now is a good time in the project to evaluate tools so you'll have one in place when you're ready to start releasing.



## Collect Key Data and Determine a Single Source of Truth

Prior to implementing CPQ, many companies have product and pricing information spread out over multiple places. It may even be duplicated, or it may vary in different systems.

Determine how you're going to resolve conflicting data and which data source wins out. To do this, choose a single source of truth for your reference data and rules:

- Customer data
- Product data
- Pricing data
- Order data

For example, let's say an item costs \$20 according to the pricing spreadsheet Sales uses. But it costs \$40 according to your ERP. How do you determine which price is correct? And how do you then assimilate the correct price into Salesforce CPQ?

## Avoid Customizations to Prevent Unnecessary Complexity

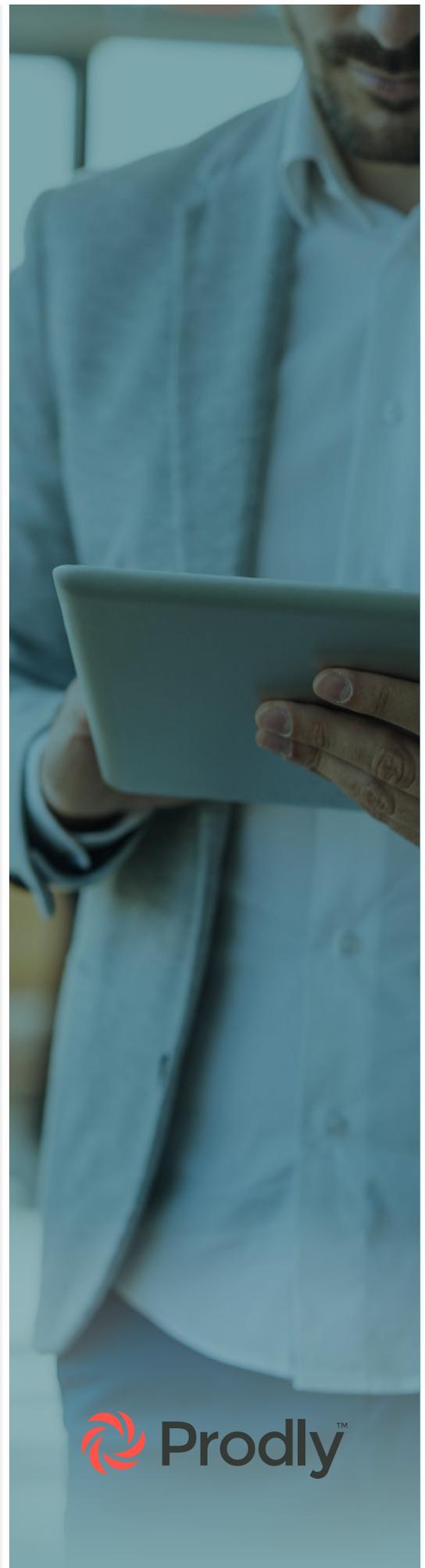
Salesforce CPQ is a powerful tool with significant capabilities. Oftentimes, too many customizations result in significant technical debt. That's why it's advisable to keep your design inside the box—and avoid customizations when possible.

It's often better to rethink how your business sells than it is to force something unnatural in the CPQ platform. Customizations build up [technical debt](#). Even with detailed documentation, the further you progress from the core platform, the more challenging they become to maintain.

Avoid complexity. Instead, consider changing your process to adhere to industry best practices.

Start simple and don't try to implement every edge case. Every domain builds on the previous in Salesforce CPQ. So it gets harder to roll out the next phase if you've already configured the tool to do things it wasn't designed for.

**Remember:** You're not trying to recreate the spreadsheet you've been using in Salesforce. Instead, you're transforming and streamlining the quoting process to make your sales reps more efficient and effective.



## Part IV: The 8 Project Domains of a Salesforce CPQ Implementation

There are eight areas of focus to define and build in a Salesforce CPQ implementation. To see the fastest time to value from your CPQ implementation, break it into phases. Start small, and keep things simple by delivering value to sales and then layering on complexity later.

### Do these first.

These three areas are fundamental and essential to all implementations. It's important to complete the analysis and build of these items first and in this specific order.

1. Product catalog
2. Configuration rules and constraints, also referred to as Bundles
3. Pricing models

### Do these later.

These items can be developed in parallel after the foundational items are complete.

1. Approvals
2. Quote templates
3. Guided selling
4. Renewals and amendments
5. Order management

---

### Product Catalog

The product catalog is the most important and fundamental part of your implementation. Use your implementation as an opportunity to:

- Rethink your quoting process with a focus on optimization and simplification
- Leverage the features of new technology
- Focus on user experience and sophistication
- Use Price Books to segregate products across regions and BUs

When developing your list of products, pay special attention to



what variations of SKUs are truly different products. There may also be SKUs you use to describe different variations of the same core product.

---

BY INVESTING IN THIS EXERCISE, CPQ CUSTOMERS HAVE SUCCESSFULLY TAKEN THEIR UNWILY PRODUCT CATALOGS FROM 145,000 SKUS TO AN EASILY MANAGEABLE 35 SKUS.

---

You'll deal with attributes like color, size, and regional variations in the next CPQ domain: Bundles.

Remember: Just because it's in your ERP, it doesn't mean it has to be the same way in CPQ. It's often easier to build your ERP integration with a simplified product model.

---

### Product Bundles

You can also refer to product bundles as product families, configurable products, attribute-based products, or suggested or required add-ons. You use them to configure and ensure technical validity of any products, or product configurations, you'll be selling.

Product bundles are:

- Any configurable product
- Not just "packages"
- Used to set attributes such as size, color, term, and geography

---

WHEN CREATING PRODUCT BUNDLES, THINK "SELLING A SOLUTION" INSTEAD OF SELLING INDIVIDUAL PRODUCTS. THINK "SELLING VALUE" INSTEAD OF PRICE.

---

### Pricing Strategy

Once you've determined permitted product configurations, you're ready to price them accurately. The basic catalog and bundle setup in Salesforce CPQ features a lot of good pricing data. However, it might not be the end-all for your company.



Nonetheless, it's best to start with the simple catalog before making any adjustments to the pricing.

Keep the following in mind:

- While bundles ensure technical validation, pricing ensures commercial validation.
- It's difficult to implement pricing without first finishing the catalog and bundles.
- When making a pricing strategy, consider the impacts to approvals.
- Use this as an opportunity to consolidate practices.

---

PRICING AUTOMATION IS FOR THE “GO-TO-MARKET PRICE.” APPROVALS ARE FOR THE “ACCEPTABLE PRICE.”

---

## Approvals

Let pricing automation eliminate problematic quotes and simplify the process when it comes to approvals. If you'll never approve certain margins or discounts, simply don't allow those levels to be saved on a quote. You can set floors, ceilings, or other types of limits inside of Salesforce CPQ to maintain pricing and configuration guidance.

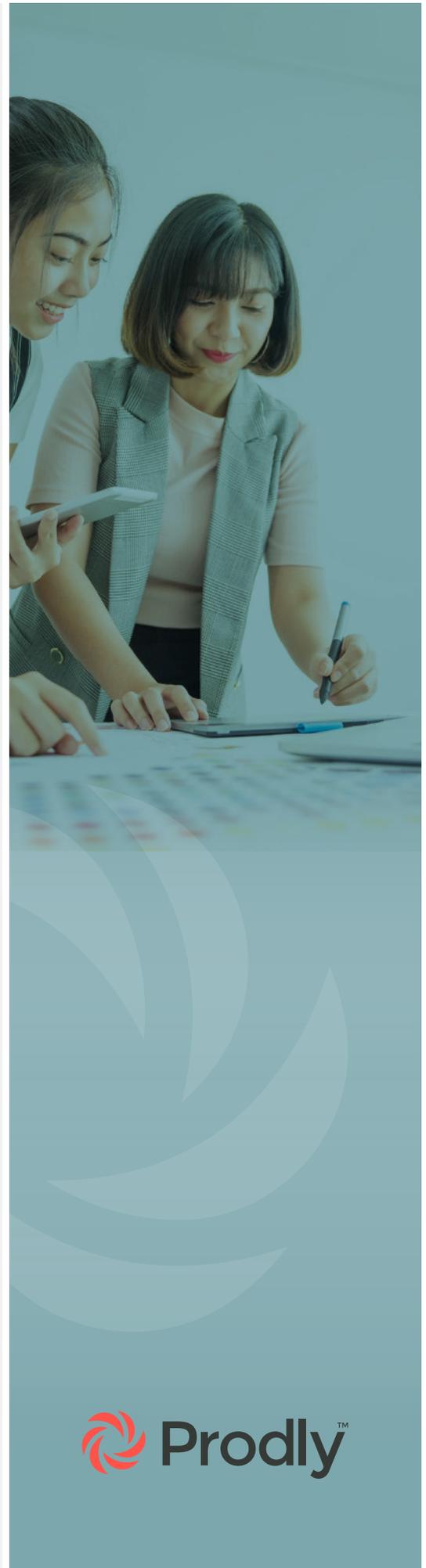
If you've done a good job on the foundational domains, you'll have already eliminated scenarios that previously needed review and approval.

For approvals that cannot be made redundant with pricing controls, analyze who's currently involved in the process and why. Determine whether there are redundant or superfluous approval steps.

Does every quote really need to go to approval? Can single approvers cover many functions? You may be able to consolidate or standardize approvals across business units or product lines.

When designing your approvals automation, consider the following:

- Set floors, ceilings, etc. to avoid the need for approvals.
- Do you need standard or advanced approvals?
- Implement immediate feedback for users.
- Consolidate and standardize approval logic.



---

## REVIEW APPROVAL RECORDS FREQUENTLY TO ELIMINATE BOTTLENECKS AND RUBBER STAMPS.

---

---

### Quote Templates

Quote templates are a powerful selling tool and the first domain where you need to really consider the customer experience. Design templates that present the technical and commercial details for any type of deal you're selling in an easy-to-understand manner.

Make sure you have clear requirements regarding the template's design, as well as an understanding of who has final say over it. This will help you avoid the following problems:

- Sending out PDFs of Excel spreadsheets or messy Word documents because you don't have a standardized template for presenting pricing information.
- Having a plethora of different templates that only differ slightly for various regions, business units, or product lines.
- Receiving a never-ending list of non-critical demands based on arbitrary, individual preferences.

---

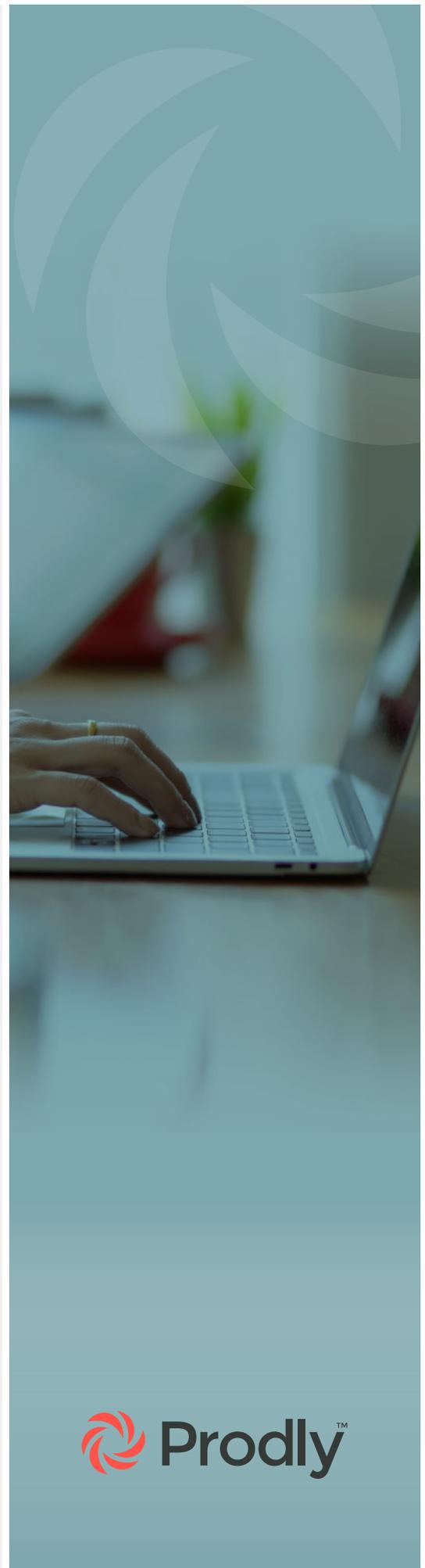
## THE QUOTE DOCUMENT IS A POWERFUL SELLING TOOL! STANDARDIZE QUOTE TEMPLATES AND CONSOLIDATE DOCUMENT APPROVAL AUTHORITY.

---

When creating quote templates, identify key needs. Consider different types of documents, branding and layout, clarity, and messaging. In addition, think about dynamic vs. static content, additional calculations, and localization.

### Guided Selling

The most important factor to consider with guided selling is the overall user experience when adding products to the quote. Guided selling is an advanced feature that introduces some rigidity to the product selection experience. Note that you can have a great guided quoting process even without using the guided selling feature of Salesforce CPQ.



You can achieve a similar construct to guide sales reps through the quoting process by creating automation with data known before the quoting phase. Leverage price books, product catalog optimization, search filters, and bundles to achieve a user-friendly experience.

---

### Renewals and Amendments

To manage recurring business or subscriptions, you need to determine the setup of the products and how they relate to renewals and amendments. You also need to decide how you'll manage renewal and amendment quotes and opportunities.

To properly manage renewals and amendments:

- Define when renewal Opportunities and Quotes should be created.
  - Consider upgrades, add-ons, and other amendments.
  - Document common customer change requests.
  - Determine and automate customer lifecycle touchpoints.
- 

ALWAYS KNOW YOUR FORECAST, EVEN YEARS IN ADVANCE. KEEP RENEWAL OPPORTUNITIES UP TO DATE, AND NIMBLY CHANGE COURSE.

---

### Order Management

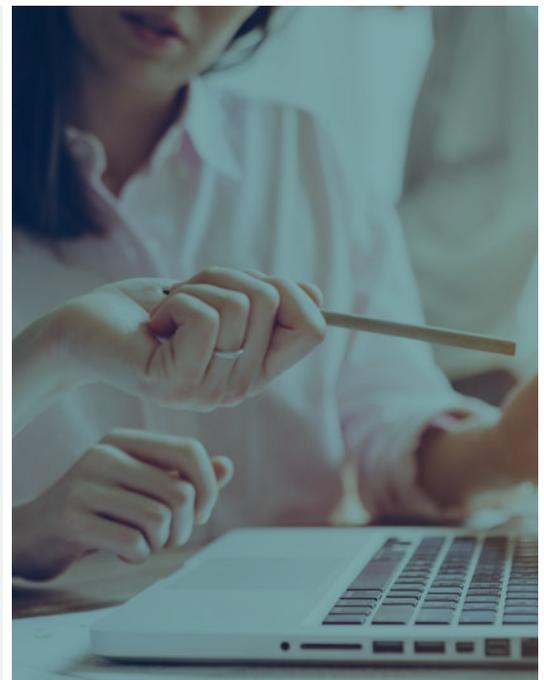
Salesforce CPQ also interacts with the Order object. Moreover, order management is a must-have if you physically fulfill products, need to provision products, or use Salesforce Billing.

Order management manages discrepancies between products quoted versus the products that were delivered, installed, and/or provisioned to customers. It allows you to split quoted items across multiple locations and start dates.

---

USE ORDER MANAGEMENT WITH CPQ TO MANAGE DELIVERY OF PRODUCTS.

---



## Part V: Time to Implement

This is where you implement the plan you created in Part III; where the theoretical becomes real. If you've taken the time to think through your strategy and carefully prepare, your implementation should be relatively smooth.

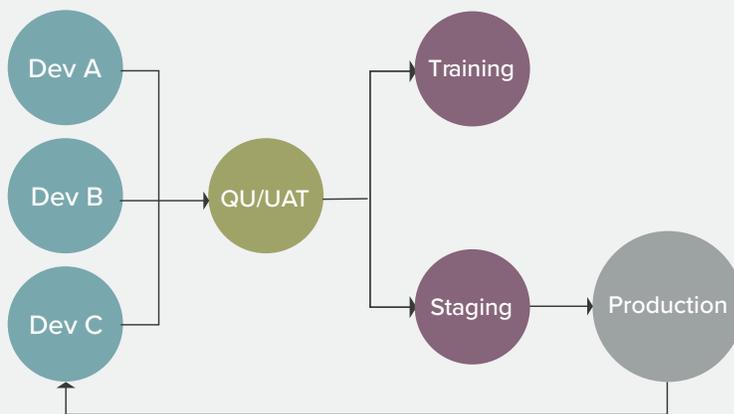
### Implement DevOps Best Practices

Adhere to DevOps best practices to remain nimble throughout the implementation. These best practices are:

- Work in dedicated environments to avoid stepping on each other's work.
- Establish source control to promote changes up the release pipeline and track changes to CPQ configuration for easy, targeted rollback if needed.
- Leverage automation to drive testing and promotion.

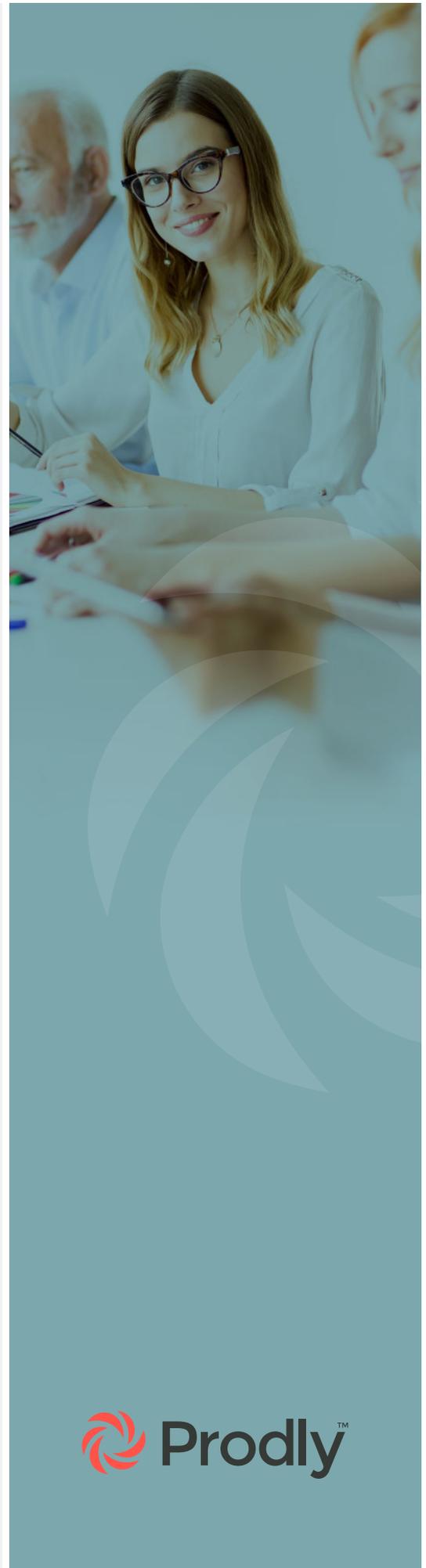
---

### Create Dedicated Environments With Sandbox Management



Begin by preparing lower-level environments like Developer, Developer Pro, and scratch orgs for the development process. Here, you'll build the implementation with all the features your users need without compromising their business processes in your production environment.

To follow DevOps best practices, assign a developer environment to every admin or developer on the project. [Populate each](#)



[org with current CPQ configuration data](#) from your production environment that relates to the project you're working on. During the development process, make sure to keep parallel orgs in sync.

This results in a smoother development process with fewer bugs and bottlenecks in the release pipeline.

If you don't seed the right CPQ records to your development environment, you won't have visibility of the full CPQ configuration. This is problematic because you're likely to introduce unknown "bugs" like duplicate products and discounts, as well as conflicting price rules. As a result, the implementation will be problematic and compromise your sales reps' trust in the system.

---

### **Establish Source Control to Automate the Movement of Changes, Enable Rollback, and Ensure Compliance**

---

SOURCE CONTROL IS A CHANGE MANAGEMENT STRATEGY WHERE YOU TRACK AND MANAGE CHANGES TO YOUR APPLICATION'S CONFIGURATION OVER TIME IN A REPOSITORY.

---

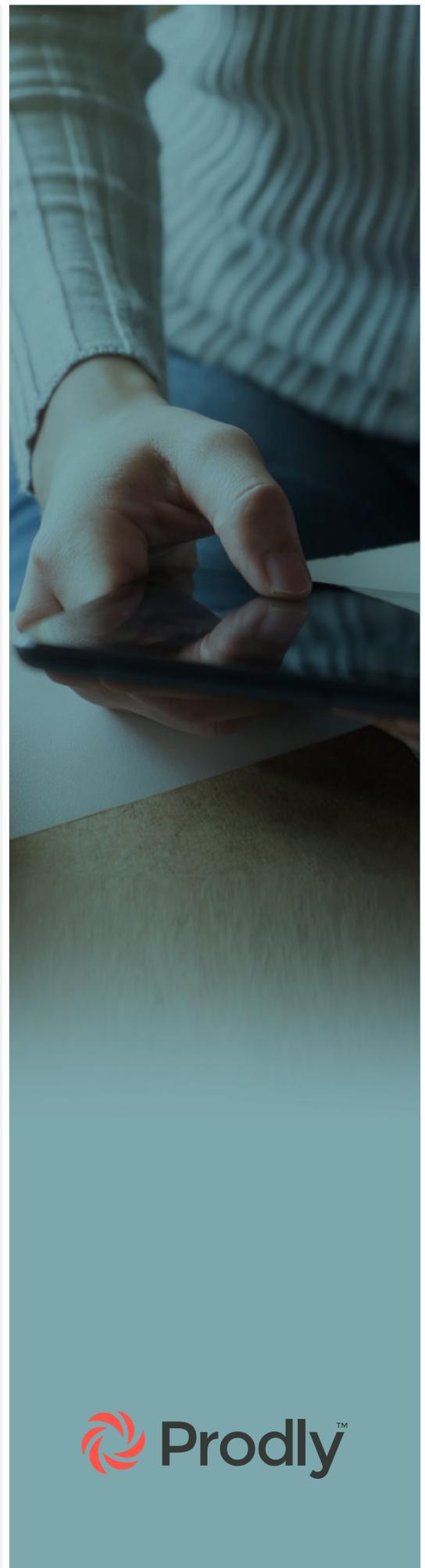
Source control allows you to effectively, safely, and quickly iterate on an app without breaking things. It allows you to automate the movement of both programmatic and declarative configuration changes through the release pipeline.

Plus, if you do break something, you can leverage the rollback feature as your safety net. You can quickly revert to a previous version and minimize the impact to users.

---

IN SALESFORCE CPQ, YOU'RE TRACKING CHANGES TO CONFIGURATION DATA. PRODLY DEVOPS IS THE ONLY TOOL THAT OFFERS INTEGRATION TO VERSION CONTROL SYSTEMS FOR CONFIGURATION DATA. IT INTEGRATES WITH GITHUB, BITBUCKET, AND AZURE.

---



To ensure compliance with SOX regulations, you need to document the who, what, when, where, and why of moving changes between your orgs. Source control does this for you. Prodlly DevOps offers a version control feature that provides a time-stamped audit trail of every change. In addition, it lets you generate an audit report for compliance purposes with just the click of a button.

---

### Regression Testing to Monitor Quote and Contract Accuracy

As you complete each project phase, you want to ensure that new features you've approved didn't accidentally change the existing quoting functionality before you release them to users. It's better to find an error with the quote calculation or an unintended discount before your sales reps or customers do! The best way to accomplish this is by converting items of your unit testing into regression testing.

---

**AUTOMATED REGRESSION TESTING ENSURES THAT ALL YOUR HARD WORK ISN'T WIPED OUT BY ADMIN MISTAKES OR SYSTEM UPGRADES.**

---

Regression testing ensures that CPQ still functions as desired after any changes to the configuration. This is regardless of whether the change comprises configuration data or code. Because we recommend working iteratively, you should build regression testing into your CPQ implementation.



## Part VI: Go Live

### User Acceptance Testing (UAT)

You need to get clear sign-off from your stakeholders that the new system is performing useful functions for them. Clearly document the reasons why you're building things, and set standards for their expected outcomes. That way, you can determine whether or not build items are meeting their criteria.

---

**ESTABLISH CLEAR USER TESTING CRITERIA AND USE CASES.**

---

### Educate Users About the New App in a Training Org

It's advisable to teach users how the new app works before letting them use it in production. You can use a sandbox populated with sufficient data to approximate your production environment, as a training environment. That way, users can freely experiment with CPQ without the worry of cluttering up production with test Opportunities and Quotes.

### Deploying Salesforce CPQ to Production Is Just the Beginning

Once the app meets all requirements and stands up to testing in a real-world environment, it's time to deploy the changes to production. But when you go live with your Salesforce CPQ implementation, it's not the end—it's the beginning.

Have you been working with an implementation partner? Then the point at which the initial Salesforce CPQ implementation is accomplished is likely when the consultant hands the implementation back over to you.



---

CONSULTING PARTNERS ARE KNOWLEDGEABLE ABOUT SOLUTIONS THAT CAN SAVE YOU SIGNIFICANT AMOUNTS OF TIME AND MONEY—LIKE PRODLY. AND THEY'LL WANT TO EMPHASIZE THE VALUE OF THOSE SOLUTIONS. AS SUCH, THEY OFTEN RECOMMEND THE TOOLS THEY'VE LEVERAGED TO QUICKLY AND EFFICIENTLY HANDLE THE IMPLEMENTATION.

---

### Establish How to Easily Report Results

You began the CPQ implementation with certain results in mind. Document and share the goals you want to achieve, as well as the metrics you'll use to show those objectives have been met.

Do you want to achieve a faster approval turnaround time? Better overall margins? Or higher average sales? Establish how to easily report these results.



## Part VII: A Final Word About Change Management in Your CPQ Implementation

As your business grows, you need an ongoing change management process so you can keep adapting to new customer demands, new market capabilities, and more.

---

COMMUNICATION AND COLLABORATION ARE KEY IN A NEXT-GEN DEVOPS CHANGE MANAGEMENT CYCLE.

---

Next-gen DevOps is about breaking down silos between IT and the business. You need to make sure the teams are having conversations with each other. You also need to ensure that everyone is aware of who's doing what, where, and why.

Last, but certainly not least, you must ensure everyone is making use of all the low-code and no-code tools that are available. This will make the change management process smoother, faster, and more efficient.

**To learn more about implementing DevOps best practices for Salesforce CPQ—request a [personalized demo!](#)**

