CANIDIUM

SAP CPQ

TRANSFORMING CPQ PERFORMANCE

Canidium Supercharges Customer's CPQ
System with Blistering 5x Performance Boost

In today's fast-paced business environment, the performance of Configure, Price, Quote (CPQ) systems is paramount to maintaining a competitive edge. Slow CPQ processes not only hamper efficiency but can also significantly impact customer satisfaction and sales cycle times. Recognizing these challenges, our client approached Canidium, seeking to enhance their CPQ system's performance. This blog post delves into the transformative journey undertaken by Canidium, culminating in more than a 5x improvement in CPQ system performance for our client.

The Impact: A Leap in Performance

By Paul Trozan

Our client's CPQ system was plagued by sluggish performance metrics that hindered their operational efficiency. The results of our interventions were nothing short of transformative:

Adding a single product to a quote took approximately 50 seconds due to the massive number of calculations taking place. We were able to reduce this below 10 seconds - an 80% improvement!

Editing a product was cut in half, from a 15-second wait time down to 8 seconds.

Selecting attributes during the configuration process was taking a full 2 seconds, which was cut down to 0.5 seconds - fast enough that the user no longer notices any delay.

Such delays in the CPQ process can have cascading effects on sales cycles, affecting revenue and market responsiveness. These enhancements significantly accelerated our client's CPQ process, directly impacting their sales efficiency and customer satisfaction positively.

IN THIS CASE STUDY

1. Elimination of Unnecessary Event Triggers

Our initial assessment revealed that developers often caused scripts to trigger on events unnecessarily. While this made troubleshooting easier, it significantly bogged down system performance. By meticulously reviewing and removing these unnecessary event triggers, we streamlined operations and cut down processing time.

2. Reduction of Excess Quote Saves

We identified that the system was configured to save quotes excessively, a process that should ideally occur only once—at the end. By curbing these redundant saves, we ensured a smoother and faster CPQ process, significantly reducing wait times for both users and customers.

3. Optimization of Calculations

The CPQ system's cart calculations, especially for discounts and net prices, were executing multiple times due to complex configurations. We refined the calculation processes to prevent redundant executions, thereby enhancing efficiency and reducing processing times.

4. Streamlining SFDC Integration

Integration with Salesforce (SFDC) involved numerous redundant calls, each contributing to delays. By eliminating duplicate calls and batching changes into a single call, we minimized the number of necessary interactions, drastically improving response times.

5. Consolidation of Product Scripts

The need for substantial internal system processing time for running product scripts was another bottleneck. Consolidating these scripts from 8-10 into a single script significantly reduced the processing time. SAP has since acknowledged this as a bug, and it is no longer needed for CPQ performance.

6. Refining Script Internal Logic

We improved the internal logic of scripts, particularly in processing attributes. Optimizing for loops and reducing unnecessary SQL calls helped accelerate the configuration process, making it more efficient.

Conclusion

The journey of improving CPQ system performance is a testament to Canidium's commitment to leveraging technology for operational excellence. By identifying key inefficiencies and implementing strategic changes, we were able to significantly enhance the performance of our client's CPQ system. This success story underscores the importance of continuous optimization and innovation in today's business landscape, ensuring that companies remain competitive and responsive to market demands.

HOW WOULD I NOTICE SLOW CPQ PERFORMANCE?

Canidium consultants speak in plain language whenever possible.

Instead of an auto repair shop advertising "Is your torque converter running slowly?", we ask simple root questions, "Do you hear rattling noises from your transmission?"

These are the symptoms you should look for:

- Do you experience excessive delays when adding a new product to your quote?
- Find yourself checking your phone while CPQ is updating the calculations on your quote?
- Are you constantly frustrated when configuring a product because you have to wait for 3 seconds after every selection?
- Do you get bored waiting for a product configuration to load?
- Do you realize that your time-to-quote could be twice as fast if the CPQ system loads faster? That means reps can send out twice as many quotes as before!

WHAT IF SOME OR ALL OF THESE ARE TRUE?

Contact Canidium for an SAP CPQ Technical Diagnosis

ADDITIONAL INSIGHTS AND READING

How did Canidium identify and prioritize the specific bottlenecks addressed, like unnecessary event triggers, excess quote saves, redundant calculations, etc.? What analysis methods or tools were used?

Answer: We used the CPQ developer console to look at all the functions CPQ is performing. We are also able to add custom logs to this console in order to identify additional information that we need. This is a powerful tool that most CPQ professionals do not utilize to its full extent.

With such significant performance gains achieved, were there any trade-offs in terms of system functionality, user experience or data integrity that had to be considered?

Answer: No trade-offs occurred for this client. The functionality was the same from the beginning to the end. Since calculations were simplified, it also became easier to maintain and make changes in the future.

How extensible and future-proof are the optimization strategies implemented? Can they accommodate new product lines, business rules or integration requirements without re-introducing performance issues?

Answer: These are very future-proof strategies, as they simplify the processes and thus make future maintenance easier, as well as adding new features. As far as re-introducing performance issues - this is something that crops up very easily in development and must always be kept top of mind when developing new features. This is not a forever-fix - new changes must also follow these best practices.

What processes or mechanisms were put in place to prevent performance regressions from new code/configuration changes going forward?

Answer: This process doesn't have any safeguards against performance regressions - the new changes must also follow best practices. This is like you hired a maid to clean your house. It's clean now, but you need to keep it that way or it'll get messy again.

Beyond technical improvements, what organizational or process changes did you recommend to the client to sustain the optimized CPQ performance long-term?

Answer: Include performance in your DoD (Definition of Done) for new items. A new feature can be perfectly accurate and functional, but it is not considered complete until it passes a performance test.

ARE YOU READY TO FIX YOUR CPQ PERFORMANCE ISSUES?