



How Artifact Uprising leverages Precog for advanced self-service data access



ARTIFACT UPRISING

Who is Artifact Uprising and how do they use data?

With Artifact Uprising (A|U) users can create custom framed photos or entire photobooks for important family events like weddings, birthdays, or anniversaries. The application is simple to use and lets you design the entire photo project online within the Artifact Uprising application.

Users access the application through the Artifact Uprising website that is built on popular website platform Magento. Tracking user journeys and getting this data from the website is straightforward. However, the backend application that lets users design and build the actual photo album is a custom built application that runs in Amazon Web Services (AWS) using the DynamoDB database. DynamoDB stores data in a complex semi-structured way that does not work with analytic tools. This created a major challenge for the A|U analytics team trying to understand how customers were using their application to create new photo books. Important information like the date of the event and nature of the event for a photobook, or if a project had been abandoned halfway through completion were critical for the business to know.

The Challenge



Since the information about the customer project is stored in DynamoDB, getting access to this data was a challenge for the analytics team that uses Looker. If an A|U analyst wanted to know how many Wedding photobooks were created in a year getting this data was very difficult. Initially A|U hired a consultant to write some custom code to extract the needed information from DynamoDB, but this approach proved to be slow, cumbersome and was not very cost effective. Another approach they used was to print out hundreds of entire projects in PDF form and manually review them to determine the titles and dates among other information. Once again, slow and expensive, not a scalable approach.

A chance meeting between a Precog co-founder and the Artifact Uprising CEO led to a discussion of how Precog might be able to help them gain self-service access to the critical data the business needs.

The first thing to understand was the current overall analytics solution architecture at A|U. The database storing the information they needed was

AWS DynamoDB. The information was extracted using the custom code developed by the consultant and then loaded into a Snowflake Cloud Data Warehouse. Once it was in the DW it was accessed using Looker BI tool to build the reports required.

The goal was to replace the custom code with a more flexible, no code approach that lets the analysts and end users get the exact data they needed, when they needed it, without any coding or engineering resources required.

Carly Kaufman, Director of Data and Analytics at A|U engaged with the Precog team to start implementing a new approach to self-service data access.

“Analytics is all about access to data and time to value. Precog has enabled our analyst team to explore unstructured photo project data without any engineering effort. What was previously a black box of data is now at our fingertips with just a few clicks.”

The new architecture with Precog looks like the following:



Real Self-Service Data

With Precog in place analysts and other end users were able to easily access all the data in DynamoDB through a simple data browser interface, then create the precise tables they needed to analyze any aspect of the photobooks being created.

No longer did they need to print out huge PDF files and manually review them, saving them hundreds of hours of work. Additionally, since access to data became much easier and more agile, users could focus on actual

use cases that would save the business money or help avoid potential revenue shortfalls. One recent example is the global pandemic. While overall sales remained excellent since lots of people were stuck at home creating photobooks, the A|U team was concerned how all the cancelled weddings might affect revenue. With Precog in place they were able to analyze how many photobooks in the previous year were weddings, then compare this to the same time period this year and make some projections of potential lost sales in this particular market. The Artifact Uprising team has used Precog to access lots of other data to better understand their customers, orders, and projections throughout the business.

Self-service analytics describes those data analysis platforms where the average, non-technically trained business user can:



Add new data sources without expert assistance



Combine the analysts into dashboards or reports without expert assistance



Create metrics, combinations of metrics, charts, filters, and other analytical elements without expert assistance

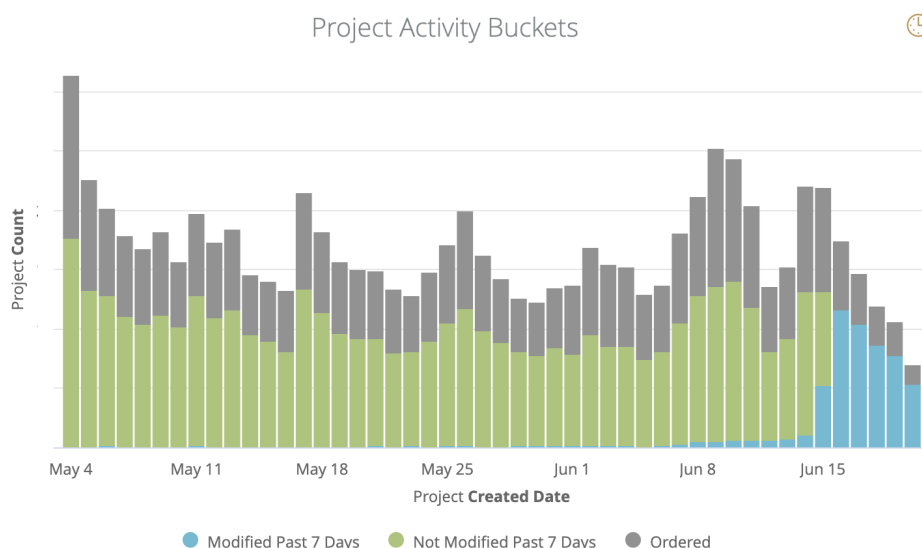


Make the data ready for analysis by performing any needed transformations without expert assistance

The Payoff

Following are some example reports created with data from the Artifact Uprising application and data accessed using Precog. These reports allowed A|U analysts to more closely track the status of books in progress and also what kinds

of books they were, like Weddings, family trips and the year the event took place. The first report, a bar chart helps them understand when a photobook in process might be an abandoned project. The second report, a word cloud helps them easily understand what are the most popular kinds of photobooks being created so they can align marketing efforts accordingly.



Summary

Many popular SaaS applications leverage modern databases like DynamoDB, MongoDB, Couchbase and others. Since these DB's store data in a complex semi-structured manner its difficult for analysts and Data Scientists to easily gain insights. Precog solves this problem and is highly cost effective. With Precog the Artifact Uprising analytics team is now able to quickly and easily access any application data they need to perform complex analysis for the business. Thanks to Precog A|U is able to better understand their customers needs and communicate with them more effectively, which means more happy customers!

