Adobe Experience Cloud

Mrsd1

01st October 2022

#### Abstract

Adobe – The company that bagged the “Great place to work” and “Most admired company, by Fortune” and many other notable titles has produced a great deal of proprietary software’s and suites to cater to multiple domains such as Education, Creator space or Business and has been in the forefront of Digital Transformation, and their moto states – “Changing the world through digital experience”.

In this paper we present the report of what Abode Experience Management Productivity Suite provides and analyzing it in Software engineering perspective. An end-to-end content management platform that includes a set of tools to help medium and large-scale enterprises to manage their customer content engagement and delivery strategies. The biggest benefit of AEM is that it simplifies social engagement, management and delivery of website content.

Finally, we shall extend the report to state the future for this Productivity suite and the improvements that is being planned.

**Contents**

1. Introduction 2
   1. [Motivation 2](#_TOC_250027)
   2. [Expected Readership 2](#_TOC_250026)
   3. [GAP Implementation 2](#_TOC_250025)
   4. [Basic Definitions 3](#_TOC_250024)
   5. [Congruences and Linked Triples 4](#_TOC_250023)
2. Semigroup Congruences By Linked Triple 11
   1. [Describing Linked Triples Computationally 11](#_TOC_250022)
   2. [Finding the Congruences of a Semigroup 12](#_TOC_250021)
      1. [Add a Relation 13](#_TOC_250020)
      2. [Subpartitions 13](#_TOC_250019)
3. Evaluation 29
   1. [Benchmarking 29](#_TOC_250001)
   2. [Extensions 30](#_TOC_250000)

**Chapter 1**

**Introduction**

# Motivation

The digital content creator space has seen a great paradigm shift after the experience of the worldwide pandemic situation. This has led to increase in number of web footprint for various domains, businesses and individuals. As Adobe states the digital economy is a $1 trillion opportunity. Adobe Experience cloud offers a suite of products that allows business to manage their customer experience by providing them with real-time data, personalization and scalability wherever needed. Their offering appeals to enterprises with sophisticated use cases to benefit from a platform tech approach that binds insights, content, and orchestration together.

Adobes large customer base and their service being vital in some of the Fortune 500 companies has been a great inspiration to research about the suite of products that they offer. Their constant efforts in providing large percentage of turnarounds, reduced processing time and improving their customer workflow efficiencies served as a great motivation for this report.

# Product Description

# Application Domain Description and purpose

# Software Engineering process used to produce and maintain the system

Adobe creates products with the idea that revolves around building value for customers [3].

For the sake of understanding the principles involved in developing their suite of productivity software’s, we shall consider the “Control Panel” launched by Adobe Campaign. It is a microservice to empower its user to deliver a better administration experience.

As stated by Jaemi Bremner, who was a former developer at the Adobe Experience Platform, in her medium blog mentioned about how their product was developed and the principles that they adopted internally to design the system:

“Building software that enables teams is no simple task. It takes careful planning within each step of the software development process to keep in mind the users we’re enabling — from the design of user interactions to nitty-gritty details, error codes, button translations, and documentation.

In order to stay focused on making excellent enterprise software, we use the acronym CARE.”

Development Practices

Work according to a Definition of Done

Each team has a different definition of what “done” means, but it is important to have one and ensure that a story meets the defined criteria before being accepted.

Some criteria that are commonly specified by teams include:

Code reviewed for formatting

Comments/Javadoc added

Meets required test coverage levels

Passes unit and integration tests

Validated in the QA environment

Localization implemented

Without a well-defined DoD, it is easy to end up in a situation where a lot of things are halfway done and nothing is truly complete.

Define and adhere to coding and formatting conventions

Things like indentation levels and white space may not seem important, but having properly formatted code goes a long way towards readability and maintainability. Conventions should be discussed and agreed to as a team and then followed in the code.

Aim for high test coverage

As a project implementation grows in size, so will the amount of time that is required to test it. Without good test coverage, the testing team will not be able to scale and the developers will eventually become buried in bugs.

Developers should practice TDD, writing failing unit tests before the production code that will fulfill their requirements. QA should create an automated set of acceptance tests to ensure that the system works as expected from a high level.

There are custom frameworks available, such as Jackalope and Prosper, to make mocking of JCR APIs simpler to ensure productivity of developers while writing unit tests.

Stay Demo Ready

The system should be available for demo to the business at the end of each iteration. By keeping the system in a demo-ready state, the team will always be within an iteration of being production ready and technical debt can be kept to a maintainable level.

Implement a continuous integration environment and use it

Implementing a continuous integration environment will allow you to easily and repeatably run unit tests and integration tests. It will also decouple deployments from the development team, empowering the other parts of the team to be more efficient and making for more stable and predictable deployments.

Keep the development cycle fast by keeping build times low

If unit tests take a long time to run, developers will avoid running them and they will lose their value. If it takes a long time to build the code and deploy it, people will do so less often. Making short build times a priority ensures that the time we have invested into our test coverage and CI infrastructure will continue to make the team more productive.

Fine tune Sonar and other static code analysis tools and act on their reports

Code analysis tools can be valuable, but only if their reports lead to action on the part of the development team. Without fine-tuning the analysis that these tools provide, the recommendations that they generate will not be relevant and they will lose their value.

Follow the Boy Scout Rule

The Boy Scouts have a rule: “Leave it better than you found it.” As long as all of the members of the development team adhere to this rule and clean something up when they come across a mess, the code will constantly improve.

Avoid implementing YAGNI features

YAGNI (or You Aren’t Gonna Need It) features are things that are implemented when we expect that we will need something in the future, even though we don’t need it now. Ideally, we should implement the simplest thing that will work today and use continuous refactoring to ensure that the architecture of the system evolves with the requirements over time. This will allow us to focus on what matters and prevent code bloat and feature creep.

More help on this feature

Best Practices

Software Architecture

Content Architecture

# Business model used by the system

# Significant features of the system from technical and business perspective

# Any limitations of the system with respect to its functionality and qualities.

Enterprise software should be made for people. This sounds obvious…but often it’s not, so it makes sense why “enterprise software” is synonymous (to some) with, “something that’s really hard to use.” We believe it shouldn’t be that way, even though building enterprise software is no simple task.

People want to feel good about doing their job, excited about going to work, and not be dependent upon others where it doesn’t make sense. That, quite simply, helps to focus on creativity and to reach true potential — making work fulfilling and enjoyable. Enabling our customers to do what they like most is the purpose of the Control Panel team.

“It’s not always about innovation or technicalities when building a product. Sometimes — or most times — it’s about experiences that we create for the customer by our products. — [Prashant Jain, Senior Engineer](https://www.linkedin.com/in/prashant-jain-b4b29594/)

“My main challenge as a designer was to understand all the subtleties of very technical features and put it into something that your grandmother could understand.” — [Elliot Puzenat, Senior Experience Designer](https://www.linkedin.com/in/elliotpuzenat/)

# Any ethical concerns with respect to the development and use of the system

# A brief discussion of possible future trends in the context of this system, including improvements you may wish to recommend

# Bibliography

**Chapter 2**

**Business model used by the system**

Since the linked triples of a Rees 0-matrix semigroup are in bijective correspondence with its non-universal congruences, we can describe a congruence *ρ* computationally by its associated linked triple Γ(*ρ*), rather than by storing a set of pairs or a partition of the semigroup. Using this representation we can compute some important properties of the congruence much faster than by these other representations.

In this section, we explain several of the algorithms for computing with these objects, as implemented in GAP by the author of this paper, and where appropriate we provide pseudo-code or well-commented GAP source code.

**Chapter 6**

**Evaluation**

In this chapter, we will show the practical value of these functions, and briefly discuss future directions in which this work could be continued.

# Benchmarking

**Bibliography**

1. Adobe AEM vs dotCMS, Open-source alternative to Adobe AEM.
2. [What Makes Adobe Experience Manager (AEM) the Preferred CMS Tool for Enterprise Users](file:///C:\Users\Mohamed%20Rilwan\Downloads\What%20Makes%20Adobe%20Experience%20Manager%20(AEM)%20the%20Preferred%20CMS%20Tool%20for%20Enterprise%20Users), Grazitti Interactive, Dec 14, 2021
3. [How Developing Products With “CARE” Puts Customer Experience First](https://blog.developer.adobe.com/how-developing-products-with-care-puts-customer-experience-first-94ee2595848f), Medium, J[aemi Bremner](https://jaeness.medium.com/?source=post_page-----94ee2595848f--------------------------------) (Former Developer Advocate for Adobe Experience Platform.), July 14,2020.
4. [Development practices](https://experienceleague.adobe.com/docs/experience-manager-64/developing/bestpractices/development-practices.html?lang=en), Experience league