Adobe AEM: Good, Bad and Ugly

Abhishek Tiwari ⓘD

Published on: July 27, 2014

After working with Adobe AEM and several other JCR backed content management systems in last few years I have realised that there are a lot of mis-conceptions about Adobe AEM. As matter of fact, brain washing is quite common these days when Adobe AEM is on the table. Sometimes Adobe is on fault, on other occasions rhetoric of competitors, agencies and architects.

So I decided to help some of you by offering this list of the good, the bad and the ugly about Adobe AEM. This list is by no means comprehensive but I have covered most of things which I consider important when it comes to making a technical and architectural decision about Adobe AEM. I will be quite frank and honest here - I am not a Adobe AEM (aka Adobe CQ) expert, but I have a strong engineering and architecture background. Points raised in this list are brief and specific pointers. Lastly, this review is based on latest Adobe AEM 6.0. Let me know if I have missed something.

Good

- Open-source and standards driven architecture (JCR, Apache Jackrabbit, OSGi, Apache Felix, Apache Sling - you name it)
- Highly modular and decoupled architecture, every module is an OSGi bundle - whether it is AEM modules or custom modules
- Scalable and performant hierarchical content repository for all type of contents including digital assets
- Content-driven REST framework Apache Sling brings back the fun to work with JCR
- Deep integration with various Adobe marketing cloud products such SiteCatalyst (Omniture), Test&Target, Social, etc.
- Client-side personalisation is implemented using JavaScript hence content can be easily cached at downstream (CDN)
- Managing templates, configuration and code as content has advantages in flexibility and extensibility
- Essentially everything is a component including pages and components act as building blocks of pages
- Strong in-place editing, component drag and drop widgets and components
- AEM 6.0 introduces a new powerful HTML templating system **Sightly** which preferred over JSP and ESP
- Separation of concerns, with Slightly business logic can be placed in an external helper class separate from presentation layer
- AEM 6.0 introduces CRX 3 aka **JackRabbit Oak** with MicroKernel (MK) implementation with current variations TarMK and MongoMK, with others in the pipeline
- By default repository content can be stored as tar files (TarMK) which offers high-availability, failover and compatibility to CRX 2.0

- MongoMK offers high-scalability and read-throughput can be increased by adding new MongoDB replicas, MongoMK suitable shard based scenarios such as user generated content and social content
- Valid HTML5 templates supported by HTML5 widgets and components
- E-commerce integration framework with connector for various commerce engines including Hybris, ElasticPath, Intershop
- Solid digital asset management functionality with native integration with Scene7 for dynamic media delivery
- Simple visual workflow designer that covers most use cases and can be easily extended for new ones
- Decent multilingual support and management, DAM can be used for multilingual purposes
- Intuitive, user-friendly, and attractive authoring and management interfaces
- Appealing and consistent single UI pattern known as Coral UI[1] or CloudUI visual style used for all Adobe products
- Classic UI powered by ExtJS will be phased out and replaced by new touch-optimises UI with responsive design
- Coral UI is designed to provide unified and clean HTML5 markup and support new touch-optimises UI
- Responsive web design ready including support for responsive images
- Strong mobility strategy - numerous emulators for testing, previewing and editing content in context
- Leverages PhoneGap (Apache Cordova) JavaScript libraries to integrate device features to deliver device components
- Dispatcher on-disk cache along with newly introduced AEM's own in-memory HTTP Cache ("HTTP Cache Filter") can handle caching in most of scenarios
- Unique developer mode in authoring allows functional testing and finding code block rendering certain view a breeze
- AEM clustering can support a variety of setups and configurations, easy to add or remove nodes without downtime
- Live copy, rollout and multi-site manager are quite handy when deploying multi-channel/multi-site content
- Best-of-breed digital experience platform with strong strategy, technical vision and execution

---

[1]Coral UI/Cloud UI is Adobe version of bootstrap

Bad

- Adobe AEM is a product designed for engineers and sold to marketers as marketing platform causing major disconnect
- Adaptation of JCR/JackRabbit is still an issue, apart from Adobe AEM Jahia, Hippo CMS and Magnolia are only enterprise grade CMS using JCR/JackRabbit
- Apache Sling is convention driven and it is not JEE Servlet, despite being a great web framework unfortunately Sling is not used outside of Adobe AEM
- Apache Sling quite vulnerable to security flaws when inexperienced developers don't follow best practices it can expose whole repository content
- Apache Sling request processing cycle can be expensive when assembling page with large number of components, request process cycle also makes redirect expensive[2]
- Apache Sling maps URLs 1:1 to content repository paths, this behaviour is hard to change and request parameters are hack which means server-side personalisation is not ideal[3]
- No native support for dependency injection pattern, although Slice developed by Cognifide can overcome this issue (glues Sling and Google Guice together)
- Poor separation of concerns when using JSP/ESP templating, often business logic coded in JSP/ESP views, but use of Slightly or Slice can overcome this issue
- Roadmap for JCR is still not very clear- JSR 1.0 (170) and JCR 2.0 (283) is used by very few vendors and, draft proposal of next evolution JSR 2.1 (333) was published last year August
- Current e-commerce integration/connectors are not perfect, and there are various data syncing issues between AEM and e-commerce systems (promotion components, real-time updates, etc.)
- Although complementary solutions like SiteCatalyst, Test &Target and Audience Manager are highly sophisticated products, associated transaction/mbox costs is scary
- OSGi implementation Apache Felix still missing support for enterprise OSGi which means integration with existing Java EE technologies is not seamless
- Adobe's Cloud Manager is still evolving, unfortunately there is no clarity about under the hood provisioning technologies powering the cloud manager
- Adobe's Cloud Manager can provision default three-tier architecture (Author-> Publish -> Dispatcher), any other complex variation such as four-tier setup is hard to provision (Author-> Author -> Publish -> Dispatcher)
- Companies with heavy investment in Microsoft technologies and .net infrastructure find Adobe AEM hard to integrate and end-up using Adobe AEM as API backend
- Last but not least Adobe AEM TCO (total cost of ownership) is very high with return on investment is slow (ROI is not necessarily low) which is why Adobe focus is mostly large companies with

---

[2]"HTTP Cache Filter" introduced in AEM 6.0 can overcome this limitation, but you have to implement smart caching using this filter

[3]Again not an issue because Test & Target is performed on client-side and I personally don't like the server-side personalisation anyway

deep pocket

Ugly

- Adobe sales and account management team[4], seriously if Adobe wants to win the market they have to move away from fancy demos and emphasise on solution engineering and consultation
- Adobe professional services are not upto the mark[5] and often agencies and implementation partners see Adobe professional services by suspicion and as competitor[6]
- Lack of genuine talent and experts who know how to implement Adobe AEM is in proper way, most failed Adobe AEM implementations basically turned into learning project for architects and developers[7]

---

[4]This may be controversial but I know what I am talking about

[5]This is a hearsay argument, I have not first hand experience working with Adobe professional services so I could be wrong, but do cross-check

[6]Seriously, some agencies have refused to jump Adobe bandwagon just because they are concerned loosing account to Adobe professional arm

[7]I met many enterprise and technical architect with Adobe AEM experience and their understanding of platform really baffled me