bloomreach

Going headless

bloomreach

Eleftherios Karamoulas

Solutions Architect

lef.karamoulas@bloomreach.com

bloomreach

Join at slido.com
#4170128

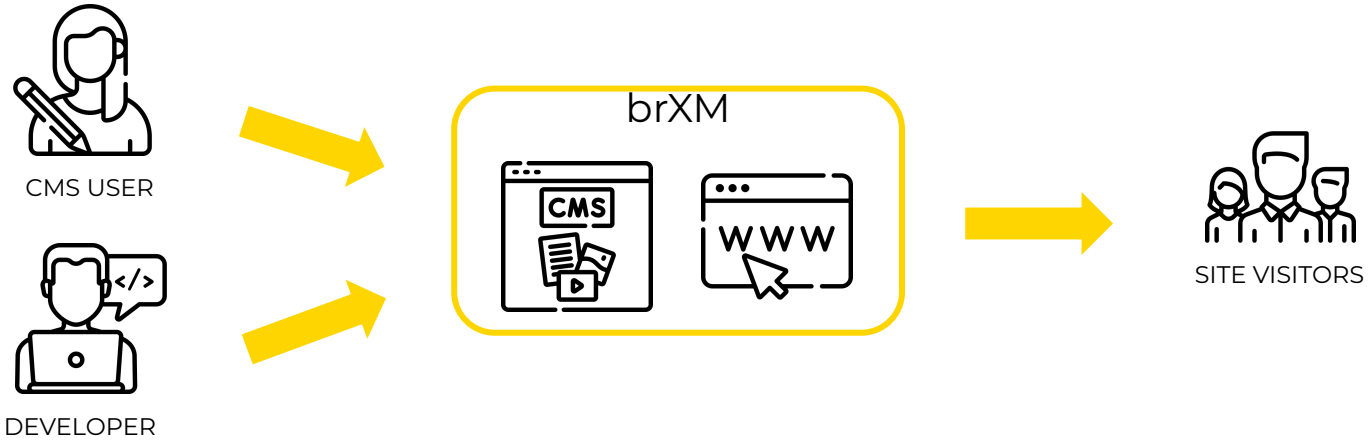
What is headless?

Definition of headless

A headless CMS is a backend-only content management system that separates content creation from its presentation, enabling flexible front-end development and multi-channel content distribution.

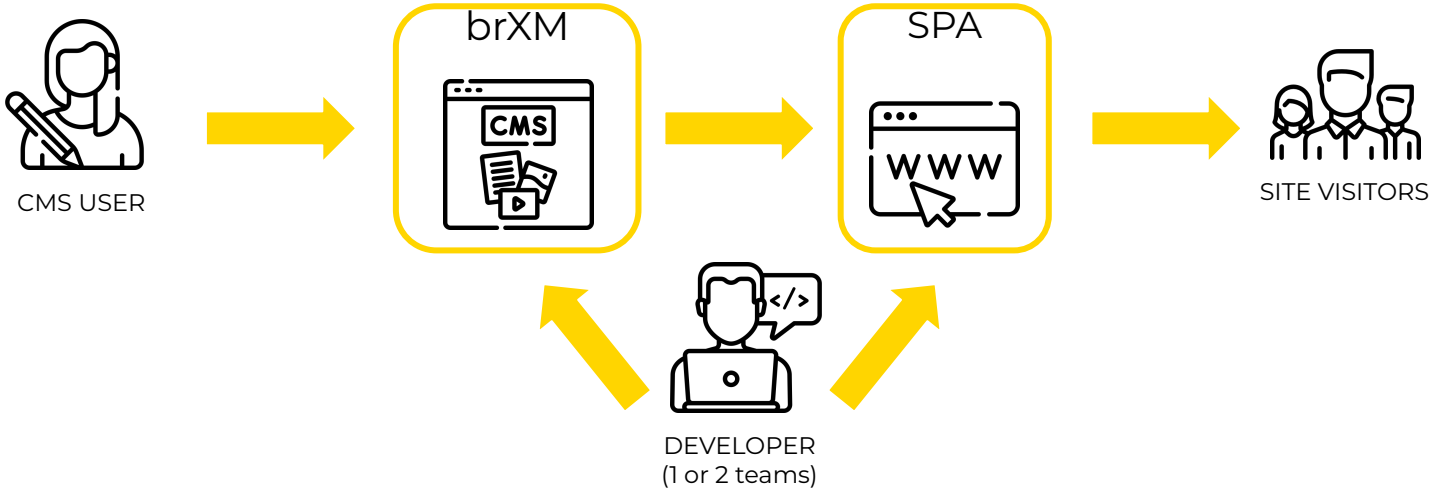
To better understand the above let’s see first a common headful architecture.

Headful architecture



- Both applications for content authoring and site delivery are part of the brXM platform
- The development team works on both CMS features and site functionality

Headless architecture



- brXM is responsible for content authoring and delivery only
- SPA is responsible for content aggregation and site representation
- Team(s) require both brXM and SPA expertise
- For CMS users and site visitors the experience remains the same

Benefits of going headless

Beware of the headless

While going headless is exciting and brings definite benefits, just like anything new and trending it also comes with cons.

So it is important to understand both in order to make a conscious decision that will impact positively your business.

Pros & Cons of headless

- Improved performance
- More flexibility
- Better scalability
- Streamlined Development Workflow
- Future-proofing

- Increased complexity
- Higher development costs
- Increased maintenance complexity
- Implementation challenges
- Dependency on APIs

Pitfalls & challenges

Pitfalls & challenges of headless

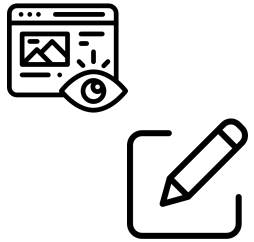
- Preview & editing functionalities
- URL rewriter (redirects)
- Sitemap.xml & Robots.txt
- Resource bundles
- Existing implementation transition
- Load of integrations on SPA side
- Deployment strategies and orchestration

Pitfalls & challenges of headless

In order to keep the preview & editing functionalities the SPA needs to meet certain requirements.

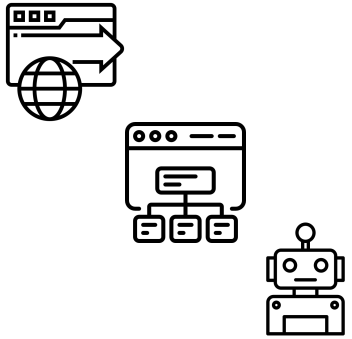
To lift this burden from the SPA team, the necessary functionality has been added to the SDKs offered by Bloomreach.

Hence the reason we advice utilising the offered SDKs because otherwise the SPA needs to implement all the underlying logic.



Pitfalls & challenges of headless

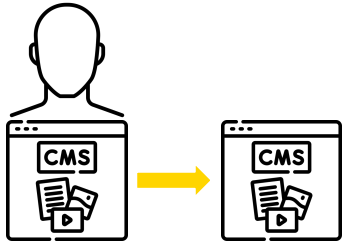
Redirect rules, Sitemap.xml & Robots.txt and Resource bundles are some of the frequently used features that were natively supported and delivered by the site application.



In a headless implementation, due to the nature of these features they are not supported by the Delivery API nor the SDKs.

In this case, implementations have to customly expose and consume such data.

Pitfalls & challenges of headless



Transitioning from headful might be more complex than starting a new headless project.

There are certain things that need to be considered upfront. To name a few:

- Will the same codebase, instances, domains be used
- Necessary coding changes
- Hand over responsibilities (brXM → SPA)
- 3rd party integrations

Pitfalls & challenges of headless

Understand the different deployment strategies (rolling vs stop/start) and how they should be orchestrated in a headless scenario.

Correct domain setup between the communicating entities can guarantee smooth transitions between deployments and switches.

Timing and order of deployments is crucial depending on the introduced changes.

Bloomreach SDKs

BLOOMREACH

What the SDKs offer

Bloomreach has developed a number of SDKs in order to support the integration between SPA implementations and brXM.

The frontend team can focus on their development while the SDKs handle the required communication between the two applications.

The SDKs also reduce the knowledge necessary for the preview integration in the Experience manager.

SPA SDK

The majority of the functionality is included in the plain Javascript SDK (spa-sdk) and all other SDKs rely on it.

The spa-sdk can also be used in cases where the implemented SPA project is based on a framework that Bloomreach doesn't provide a dedicated SDK for.

SDK flavours

Bloomreach offers dedicated SDKs for major frontend frameworks and example projects for principles commonly used (CSR/SSR).

- SPA SDK
- React SDK
- Vue2 & Vue3 SDK
- Angular SDK

- React demo
- Next demo
- Vue2 & Vue3 demo
- Nuxt demo
- Angular demo

What about headful?

State of headful

Headful is not dead. Plenty of clients still rely on this architecture and Bloomreach will continue to support it.

It is a proven solution that can compete against its successor. Development teams with brXM experience are familiar and savvy with its processes.

A faster time-to-market is observed in (simple) headful implementations due to the low complexity.

State of headful

Troubleshooting is more straightforward and frontend hot-fixing is easier to achieve.

For more complex implementations with numerous integrations a highly knowledgeable brXM/Java team is required.

A headful implementation makes your project rely heavily on brXM and consequently on its capabilities and limitations.

Pros & Cons of headful

- Reduced development time
- Seamless content previewing
- Rich feature set
- Scalability

- Limited flexibility in technology stacks
- Performance limitations
- Design and development constraints

Conclusions

Headless

- Allows for incorporating state-of-the-art technologies and web development practices
- Comes with an increased complexity for development and maintainability
- Requires skilled and knowledgeable teams for a seamless transition/implementation.

Headful

- Plenty site features and functionalities offered out-of-the-box
- Fast implementation and time-to-market with a skilled brXM team
- Monolithic architecture that depends heavily on brXM

Headless vs Headful

- Can have great difference between existing and new implementations
- Weigh team capabilities, costs and future maintenance before deciding
- Both are solid architectures adopted by our client base

Q&A

slido

Audience Q&A Session

ⓘ Click **Present with Slido** or install our Chrome extension to show live Q&A while presenting.

BLOOMREACH

bloomreach

Thank you