



adaptTo()

EUROPE'S LEADING AEM DEVELOPER CONFERENCE

27th – 29th SEPTEMBER 2021

AEM Commerce extensibility using Adobe I/O

Carlos Duque, diconium & Markus Haack, Adobe



Markus Haack
Sr SW Engineer - Adobe

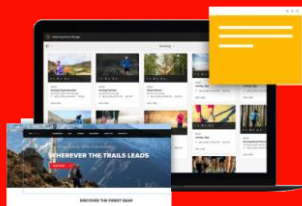


Carlos Duque
Sr SW Engineer - diconium



AEM Commerce Integration Framework & GraphQL

AEM Commerce Integration Framework (CIF)



Adobe Experience Manager

A powerhouse combo of Content and Digital Asset Management for exceptional commerce experience creation and delivery



Adobe Commerce Integration Framework (CIF)

Connecting commerce engines with AEM to provide a seamless content and commerce authoring experience with live product context

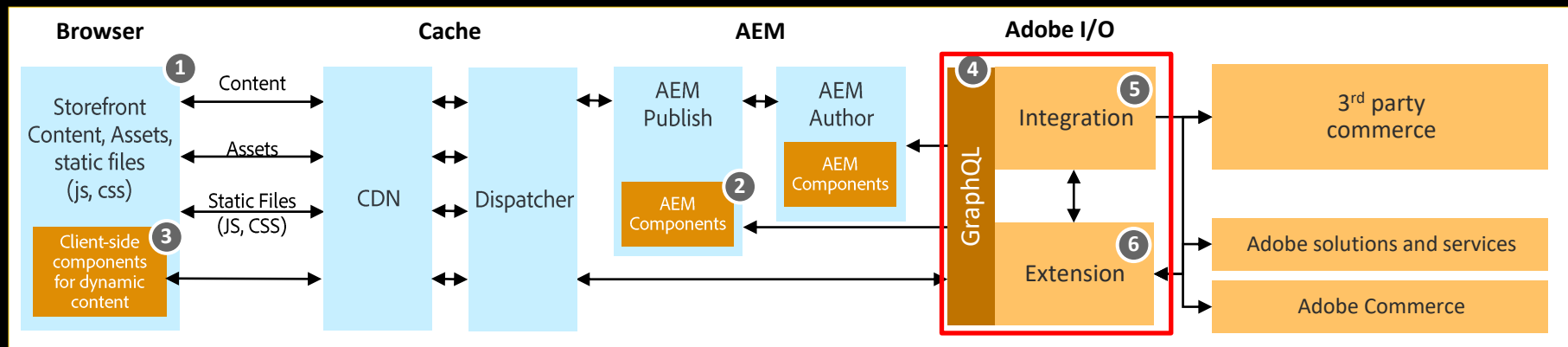


Commerce Engines

Any headless platform or set of services that provides commerce functionality and data across multiple business models and verticals



CIF Architecture



- 1 Browser requests page, assets and static files from Dispatcher / CDN
- 2 AEM Publish requests commerce data to render shopping experiences (e.g. product catalog)
- 3 Client-side components exchange data directly with services on Adobe I/O
- 4 GraphQL API endpoint on I/O Runtime. GraphQL resolver will invoke I/O Runtime actions
- 5 Integration actions transform request in 3rd party request, request data from 3rd party and transform response back in GraphQL object
- 6 Extensibility actions contain business logic, and may call other Adobe native or 3rd party services and systems

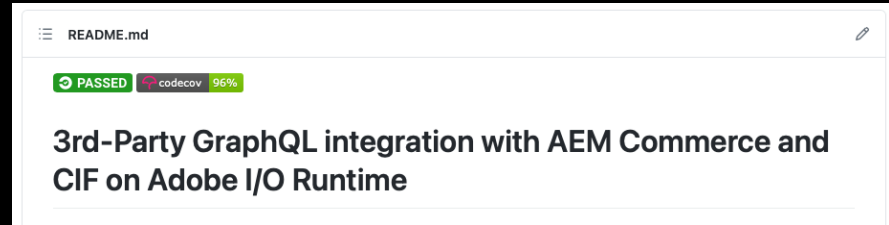
AEM Commerce & CIF & GraphQL

- Consume commerce data and use commerce transactions via GraphQL
- Used by CIF Core Components and CIF Authoring Tools
- Standardized around the Adobe Commerce GraphQL schema
- Adobe Commerce (Magento)
 - Direct GraphQL communication between both solutions
 - Extensions possible on Magento or via Adobe I/O
- 3rd party commerce solutions
 - Use Adobe I/O for translation & API mapping



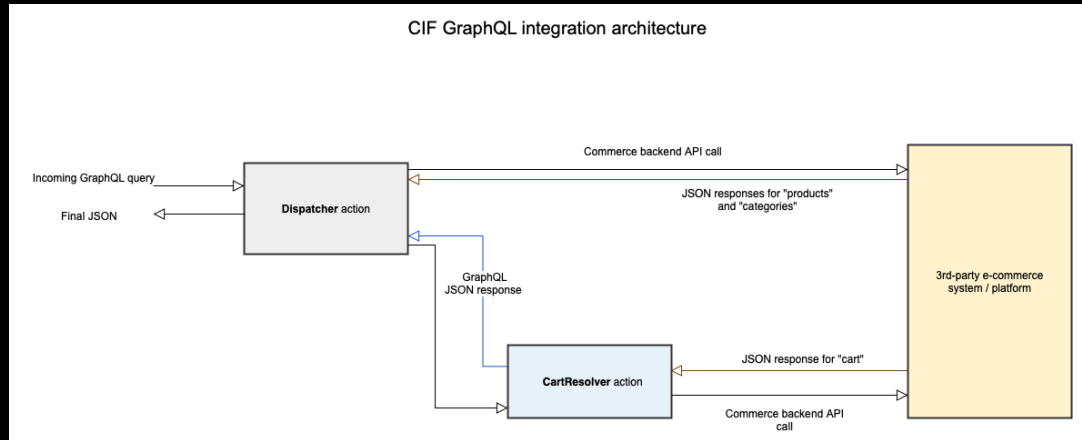
CIF GraphQL Reference

- Starter & reference project
- Best practices and example
- Frequently updated and aligned to latest CIF schema
- JavaScript, running on Adobe Project Firefly
- Not a full GraphQL mock implementation for CIF Core Components
- <https://github.com/adobe/commerce-cif-graphql-integration-reference>



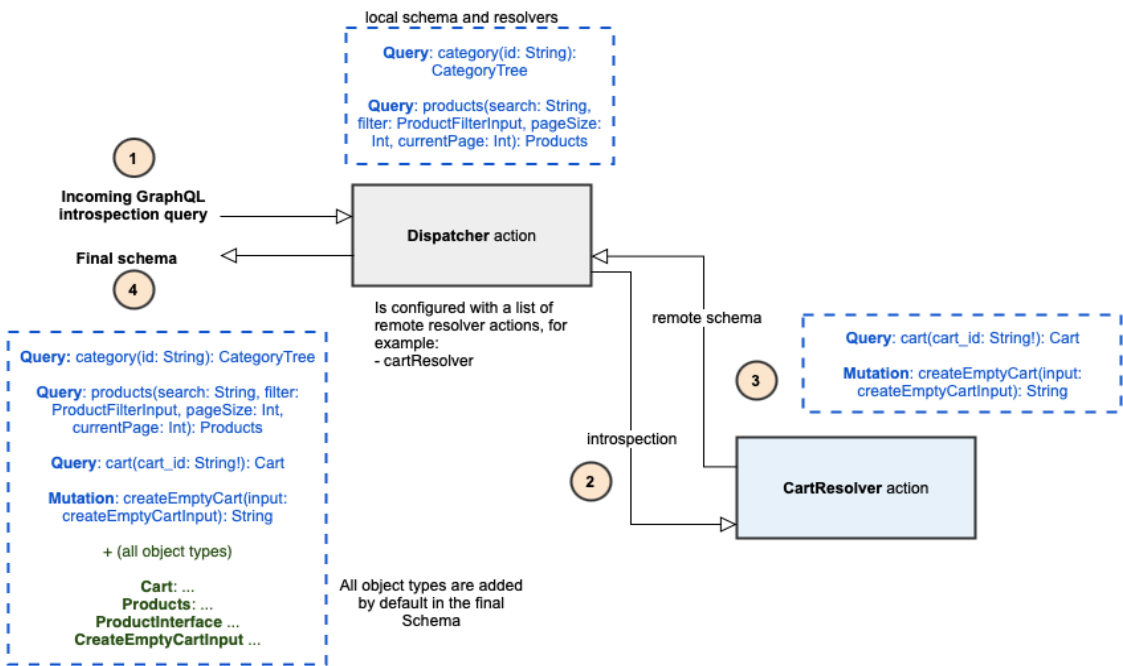
CIF GraphQL – Architecture

- **Dispatcher action** (!= AEM Dispatcher)
 - central action receiving incoming GraphQL requests
- **3 resolving options**
 - Built into the dispatcher (local resolver)
 - Delegation to separate I/O action (remote resolver)
 - Remote resolver via external GraphQL endpoint (not used shown reference)



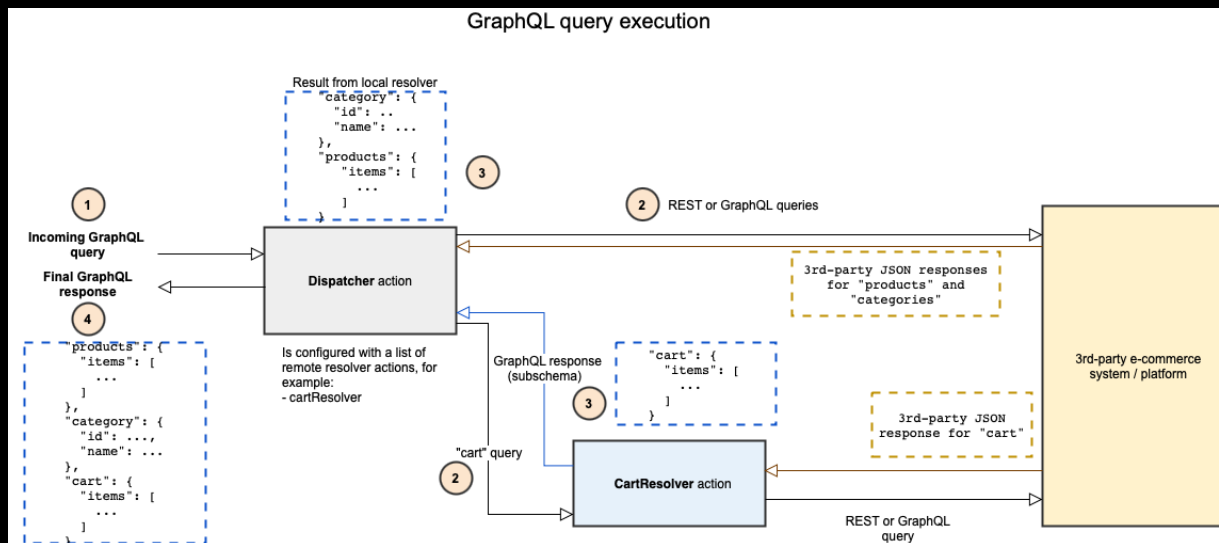
CIF GraphQL – Architecture introspection

GraphQL introspection: abstract architecture



- 1 • Incoming introspection query
 - Dispatcher generates local schema and registers local resolvers
 - For configured remote resolvers: forward introspection request
- 2 • Get introspection query
- 3 • Contribute their part of the schema back to the dispatcher
- 4 • Final GraphQL schema is built incrementally at runtime (and cached) by the dispatcher action

CIF GraphQL – Architecture query execution



- 1 • Incoming GraphQL query
- 2 • Local resolvers call commerce backend
• Resolving delegated to remote resolver actions
• Remote resolver actions start processing
- 3 • Resolvers (local & remote) get backend data and transform it into GraphQL objects
- 4 • Final GraphQL result is built by the dispatcher action by combining all the resolver results
• Final JSON send back to the client

- 4 steps
 - Configure Adobe I/O Project
 - Get “aio” CLI tool
 - Clone project from Github
 - Run & deploy via aio cli
 - `aio app run` & `aio app deploy` are your friends
- ... and ready for some GraphQL fun
- Adobe Firefly project has everything setup already
 - Code Structure, Tests, GitHub Workflows

CIF GraphQL Reference - Takeaway

- What is CIF and what is it for
- [CIF GraphQL Reference](#) for Adobe I/O Firefly
- How to get from CIF GraphQL Reference to Project
- References
 - [CIF & SAP Commerce](#)
 - multiple projects for home grown commerce services

CIF + SAP Commerce (Hybris)

- SAP Commerce (Hybris) Integration for CIF
- CIF Core Components in AEM
- Demo – seamless integration between both worlds
- Code Walkthrough
- Experiences with Adobe I/O, CIF in general

Two magic puzzle pieces

- <https://github.com/adobe/aem-core-cif-components>
- <https://github.com/diconium/commerce-cif-graphql-integration-hybris>

Demo Time

Demo 1: Product Search

- Video: https://youtu.be/t-k_dKxVxGU
- Code: <https://github1s.com/diconium/commerce-cif-graphql-integration-hybris/blob/master/cif/product/src/ProductLoader.js>
https://github1s.com/adobe/aem-core-cif-components/blob/HEAD/ui.apps/src/main/content/jcr_root/apps/core/cif/components/commerce/searchresults/v2/searchresults/searchresults.html

Demo 2: Registration and Login

- Video: <https://youtu.be/qwJ802Dt5ZM>
- Code: <https://github1s.com/diconium/commerce-cif-graphql-integration-hybris/blob/master/cif/customer/src/CreateCustomerLoader.js> <https://github1s.com/adobe/aem-core-cif-components/blob/HEAD/react-components/src/components/SignIn/signIn.js>

Demo 3: Add to Cart

- Video: <https://youtu.be/J9cvqyfWUjc>
- Code: <https://github1s.com/diconium/commerce-cif-graphql-integration-hybris/blob/master/cif/cart/src/AddProuctToCartLoader.js>
<https://github1s.com/adobe/aem-core-cif-components/blob/HEAD/react-components/src/components/Checkout/cart.js>

Demo 4: Checkout

- Video: <https://youtu.be/QH5pJaxphR8>
- Code: <https://github1s.com/diconium/commerce-cif-graphql-integration-hybris/blob/master/cif/order/src/PlaceOrderLoader.js>
<https://github1s.com/adobe/aem-core-cif-components/blob/HEAD/react-components/src/components/Checkout/checkoutContext.js>

Demo 5: Authoring Experience

- Video: <https://youtu.be/Vm8pvBP0JAw>
- Code: https://github1s.com/adobe/aem-core-cif-components/blob/HEAD/ui.apps/src/main/content/jcr_root/apps/core/cif/components/commerce/productcarousel/v1/productcarousel/productcarousel.html

- Standard NodeJS
- Easy to unit test
- Easy to have a proper CI/CD in place
- Nice documentation
- Amazing collaboration with Adobe team

Q&A

- [AEM Content & Commerce](#)
- [CIF Core Components](#)
- [CIF GraphQL Reference Implementation](#)
- [diconium CIF GraphQL for SAP Commerce](#)
- [Adobe Project Firefly](#)



Thank You