



Vue Storefront Integration

custom backend integration architecture



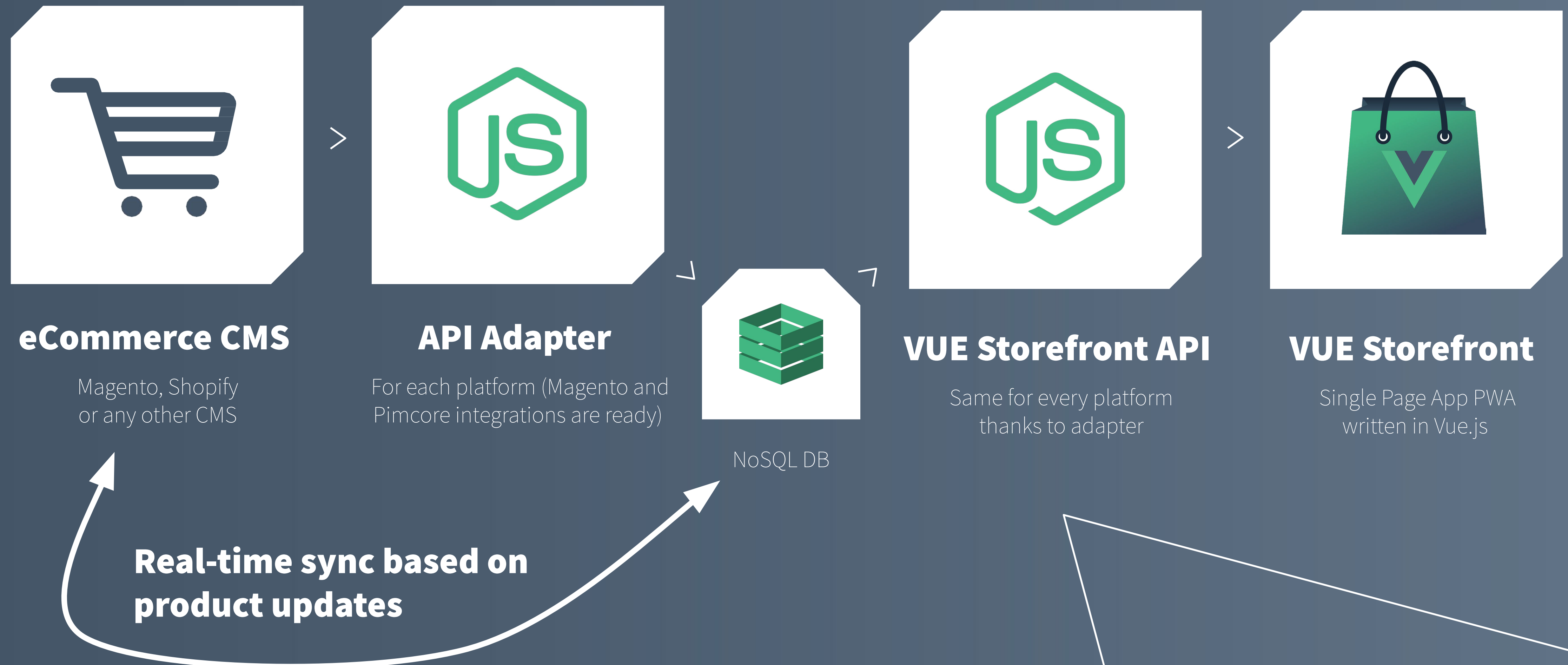
Agenda

Vue Storefront integration architecture discussion:

1. **Generic: *Vue Storefront standard*, generic data formats, Elastic.** [See tutorial](#)
2. Direct: *Vue Storefront for X*, custom data formats, direct API calls instead of Elastic
3. Dedicated: custom application based on Storefront UI

Architecture

Vue Storefront is platform agnostic and can be connected via API to virtually any eCommerce CMS. It doesn't require backend changes to your current store which makes it an excellent choice for this scenario.



Frontend Architecture

Vue Storefront Core

Contains all standard eCommerce features, state management, offline capabilities and database integration.

It's upgradable as a NPM package and extendable via extensions.

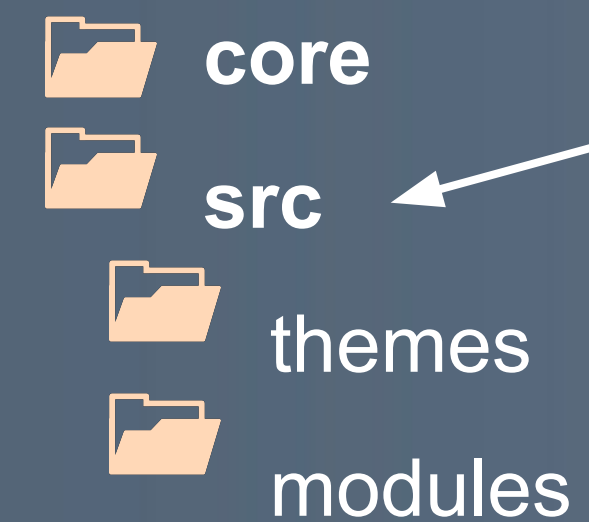


Modules

(e.g. new payment methods, mailchimp, integration, Google Analytics integration)

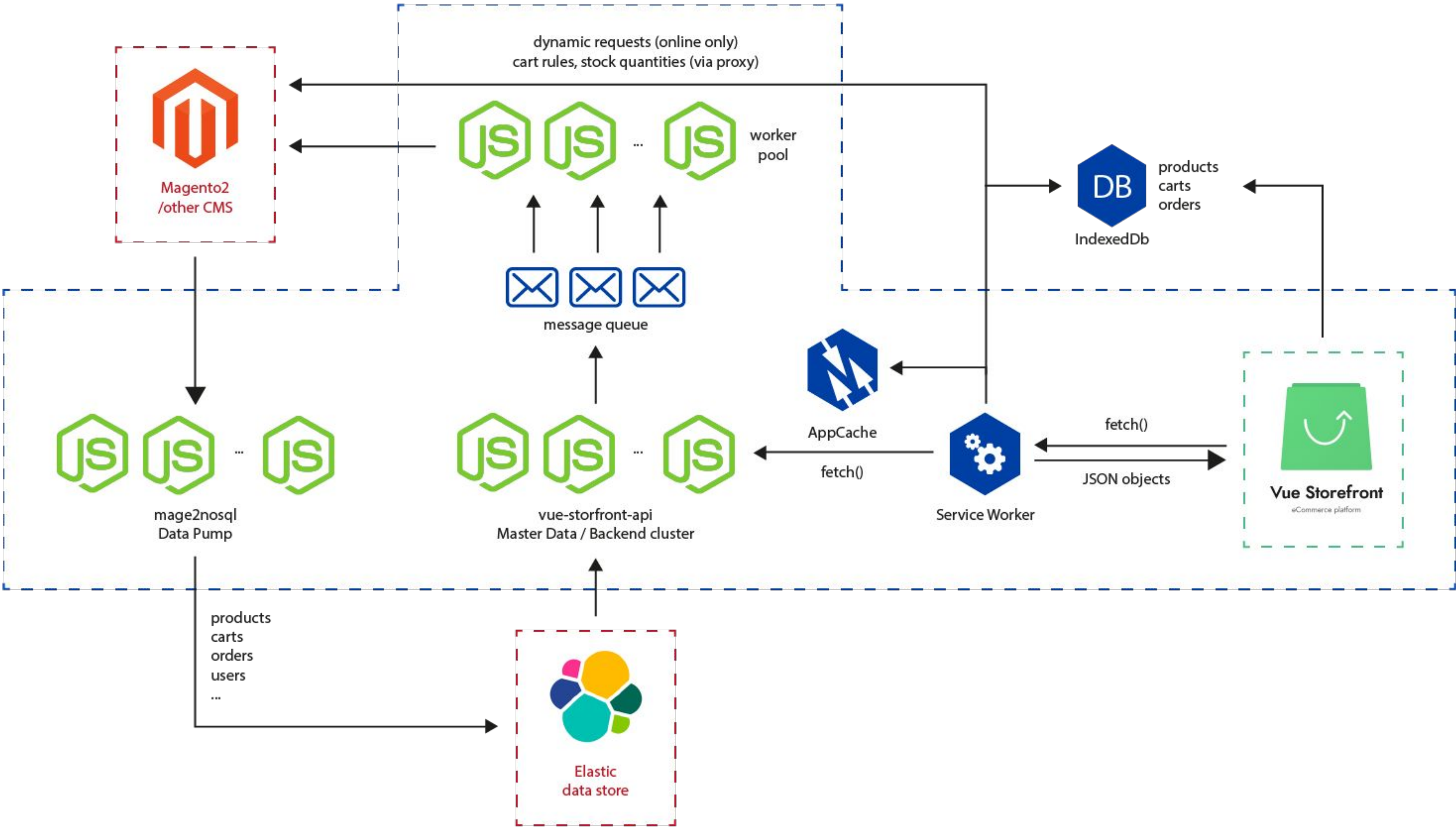
Theme

For each platform (Magento and Pimcore integrations are ready)

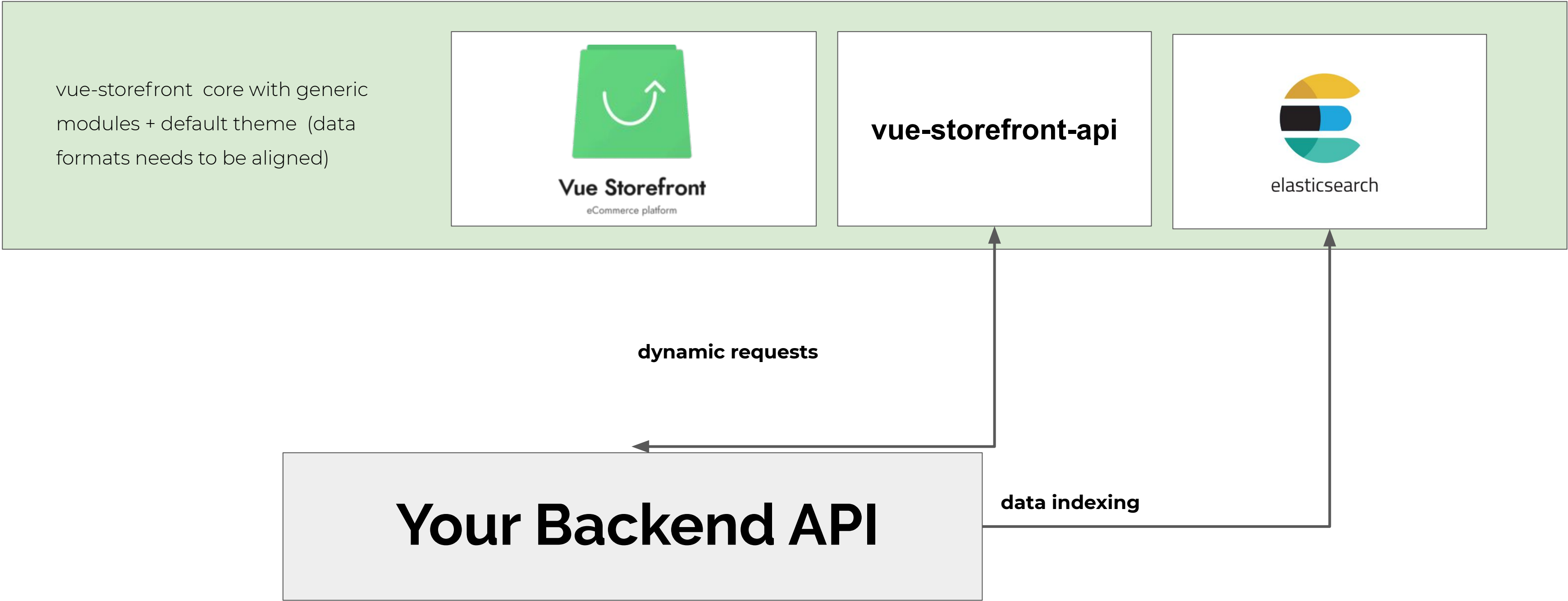


Your playground!

General Vue Storefront Architecture



Generic: Vue Storefront connector



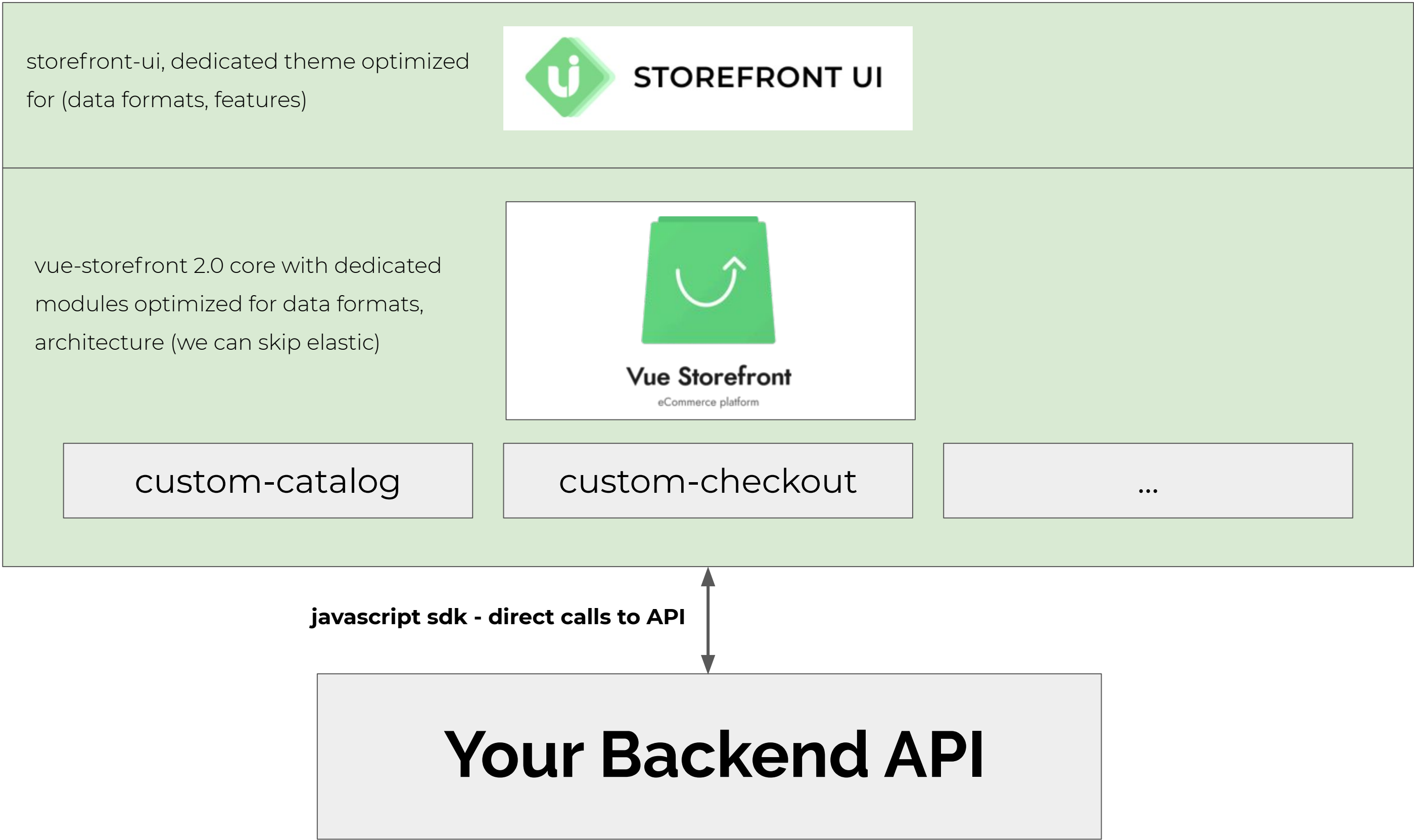
Pros

- + Visibility + OSS marketing: Vue Storefront, Community, 50 partners (agencies), trainings, docs all included,
- + All VS features (SSR, additional modules) - OOTB
- + short time to market,

Cons

- generic data formats (might introduce some limitations),
- architecture assumptions that might be limiting: eg. elastic search,
- generic solution - so users might want to change to any other backend platform

Direct: Vue Storefront for X



Optimized: Vue Storefront for X

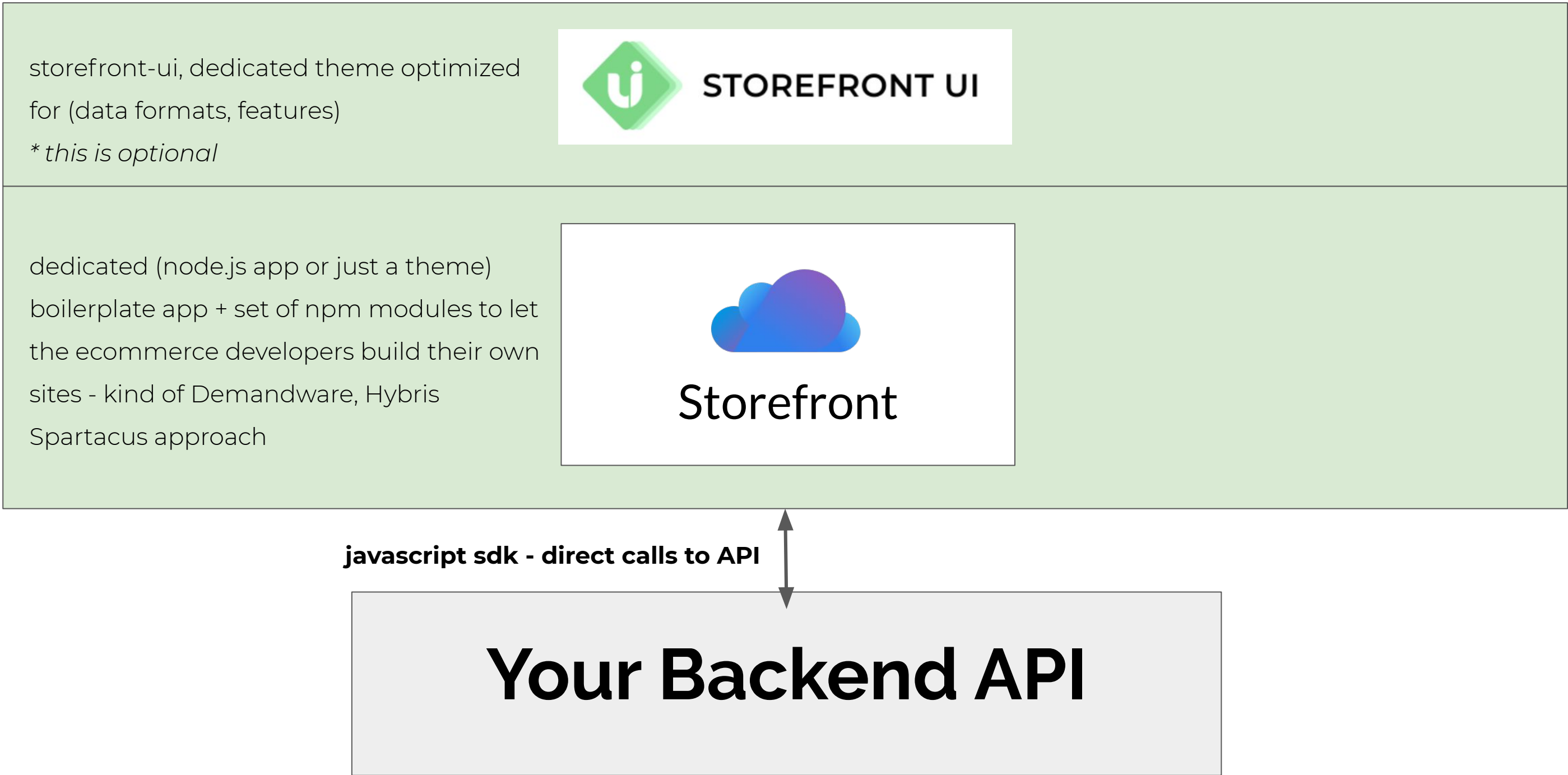
Pros

- + All the benefits of “Vue Storefront Connector”
- + Most of the docs + trainings + VS partners ready to work with new frontend,
- + Stronger Value Proposition + kind-of vendor lock-in,
- + We recommend publishing it as OSS to grow the community,
- + Optimized data flows in/out API,
- + Optimized data formats,
- + Dedicated, upgradeable theme - to leverage on UX best practices

Cons

- limited compatibility with Vue Storefront (Core, but not modules)
- part of the dedicated code that is not maintained by the community

Dedicated: Custom Frontend App



Pros

- + Optimized/dedicated data flows in/out API,
- + Optimized/dedicated data formats,
- + Strong USP / kind-of vendor lock-in,

Cons

- no community, no partners, no docs, no trainings :-),
- relatively longer time to market,
- no core foundation to base on
- no community maintenance and improvements
- no benefits from existing and future vs integrations, features and improvements

Framework - B: Storefront UI approach

Re-usable, extendable component library. To be used with any app / boilerplate (theme, Vue Storefront, Nuxt.js - whatever):

Details:

- <https://slides.com/filiprakowski/storefrontui#/>
- <https://docs.storefrontui.io/customization.html>



eCommerce Software House

Divante.co

pkarwatka@divante.co

501 601 055



Forbes



Deloitte.



Clutch