~~The~~ Our road to migrate from a legacy tech stack to a modern one.

ConFoo.ca DEVELOPER CONFERENCE

DiliTrust

Summary

01 Break down the problem

02 Choose a frontend framework

03 Choose a backend strategy

04 Identify commonalities, tools and migration sequence

05 Adapt or perish!

Context

Getting ready to growth

Business

- Growing rapidly
- Offices in Europe as well as North and South America
- Nearing100,000 users
- Eager to seize opportunities and increase market share
- Partnerships, mergers and investment in the horizon
- ISO-27001 certified

Tech

- Quickly becoming outdated
- Using a CMS as a framework
- Constraining and challenging to use
- Strict SDLC and security requirements

End of life announcement

In 2019, Drupal announced the end of life for Drupal 7 to be at the end of November 2022.

Break down the problem

Frontend, backend, architecture , infrastructure & priorities.

Choose a frontend framework

Decouple the frontend

React or Vue?

Defined criteria:

- Security
- Robustness
- Easy to learn
- Small footprint
- Ease of integration

In the end, Vue seemed to be more advantageous because it felt easier to learn, and the fact that it came with easy to integrate tools, that were officially supported.

Extended criteria	React	Vue
Release date	2013	2014 feb
Licence	MIT	MIT
Notable companies that use it	• Facebook • Instagram • Netflix • Apple	• Facebook • Netflix • Adobe • Apple
State Management Solutions	Redux, inspired by Flux. Heavy to implement, not easy to learn without using third party library.	Vuex, inspired by Flux. Officially supported and maintain by the Vue core team. Simpler to implement.
Router	React-router	Vue-router Official support.
Security	Built-in XSS protection	Built-in XSS protection
Component deployment ready	Yes	Yes
GitHub stars	146k (used by 3m repos)	160k (used by 80k repos)
Stack Overflow activity	199,130 questions	52,822 questions

Choose a backend strategy

Upgrade? New framework? New programming language and framework?

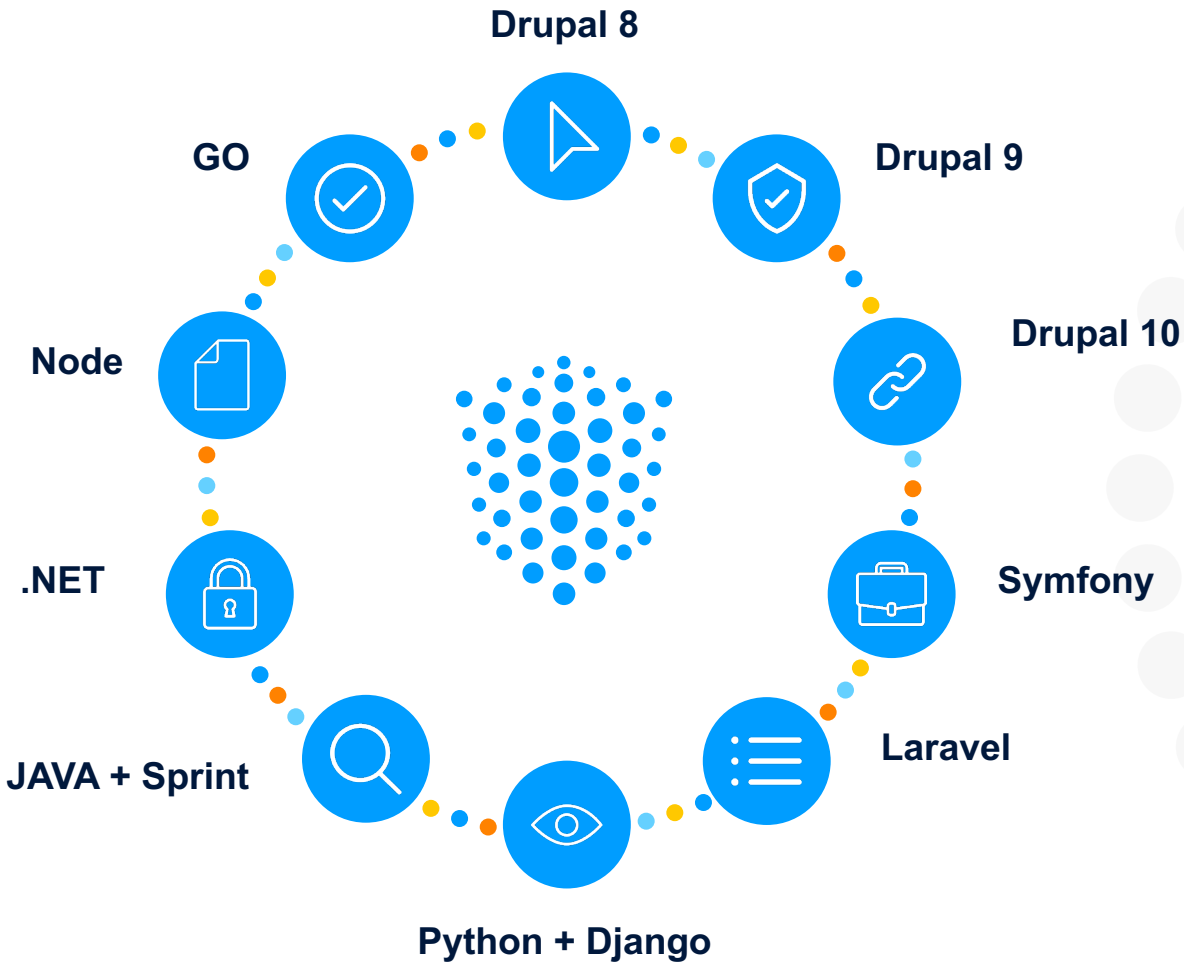
No stone left unturned

Beyond the technical criteria

Defined criteria:

- Security
- Common Vulnerabilities and Exposures (CVE)
- Maintainability
- Community acceptance
- Market Share – with 5 to 10y projections
- Learning curve
- Portability
- Web Service integration
- Scalability
- Speed of development

Symfony seemed to be the choice to build complex, well-structured, scalable and maintainable enterprise applications. Suitable for the migration of a long-term project, particularly because of its predictable six-year release plan.



Identify commonalities, tools and migration sequence

Getting things done

Moving code, creating services and defining new procedures and strategies, while continuing to support business needs.

Critical path features
Mission critical features with large code footprint, heavy use of ACL and granularity, very complex data structures, housing other features.

Complex features
Not so small code footprint, have ACL or granularity, not so simple data structure(s), are related to other features – even critical features.

Non-complex features
Small code footprint, little to no ACL or granularity, simple data structures with little to no relationship(s) with other features - especially critical features.

*** Transition strategy**
Initially called for a roll back procedure, which implied two-way data synchronization. This proved to be challenging. A different approach was chosen.

FE Decoupling *
FE out of Drupal base code, yet still calling Drupal APIs. Full component analysis, and the creation of a reusable and extensible library – 1.5y

ACL + Granularity *
One of the most complex subjects thus far. Permissions are everywhere!

Drufony
A PHP class that allowed the usage of Symfony as a library with dependency injection. Thus, making Drupal capable of calling Symfony services.

Data Synchronization *
DTO based ETL services in Symfony available in Drupal through Drufony, Initial migration + real-time one-way data synchronization.

Remember the previous context?

It changed!

Business

- Over 100,000 users
- €130M investment
- Hyperlex + AI
- More offices, new markets
- More partnerships, mergers and investment in the horizon
- ISO-27701 certified, SOC2 report, GRDP and PIPEDA compliance

Tech

- FE decoupled
- In the middle of a monumental migration
- Symfony and Drupal coexistence: Drupal still on the driver seat
- Product continuously raising the bar
- Need to accelerate the migration
- Must set an ecosystem capable of supporting our business plan

Adapt or perish

We are committed, but we must shift gears and support business.

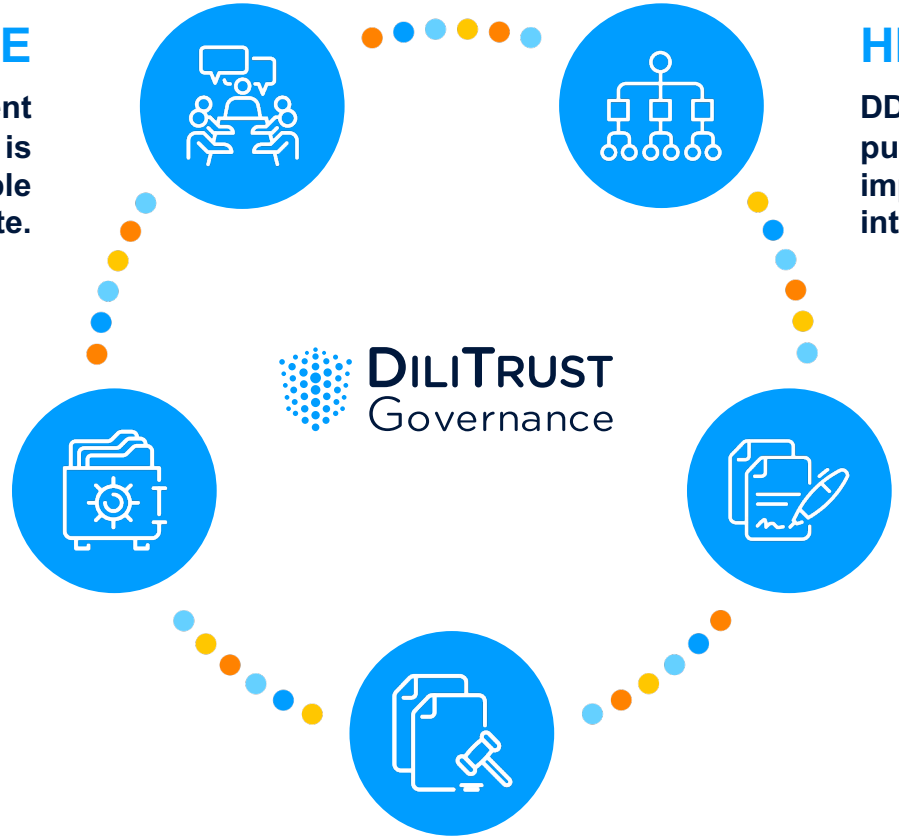
Our Products

ARTIFICIAL INTELLIGENCE

Contract management, document summary, meeting minutes and more. AI is implementing an array of APIs consumable by the Dilitrust suite.

SERVICES

DiliPass, DiliSign, BucketWrap and +



HEXAGONAL ARCHITECTURE

DDD framework with agnostic code written in pure PHP that relies on interfaces. Initial implementation of such agnostic code and interfaces is done in Symfony.

DRUPAL BRIDGE

A Symfony class that allows bootstrapping Drupal within Symfony. Abstract component implementation - interfaces. Thus, opening the door to what we now know as the MIKADO migration.

INDUSTRIALIZATION

The knowledge gathered during the early stages of the project, has led us to clearly identify the way to mass migrate – Generic layer with an accent on abstraction

DiliTrust

Thank you for your attention

ConFoo.CA
DEVELOPER CONFERENCE

dilitrust.com