

# C4 MODEL

/ La solution pour  
standardiser mes schémas  
d'architecture ?

JEROME GAUTHIER

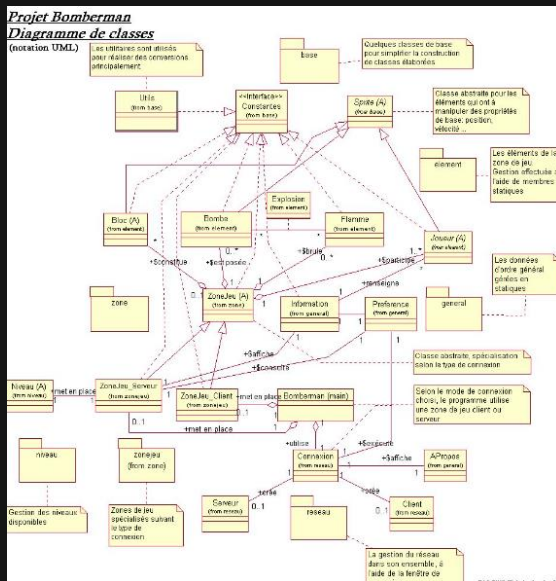
SOPRA STERIA

## / SUMMARY

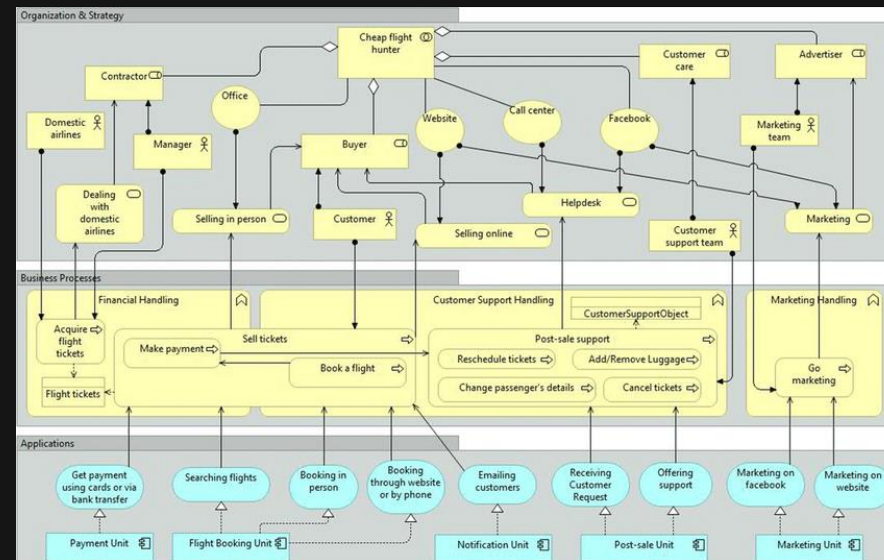


1. C4, C'EST QUOI ?  
Pourquoi ? Principes ?
2. STRUCTURIZR  
Diagram as code
3. DÉMO  
C4 model avec Structurizr
4. POUR ALLER PLUS LOIN  
À emporter

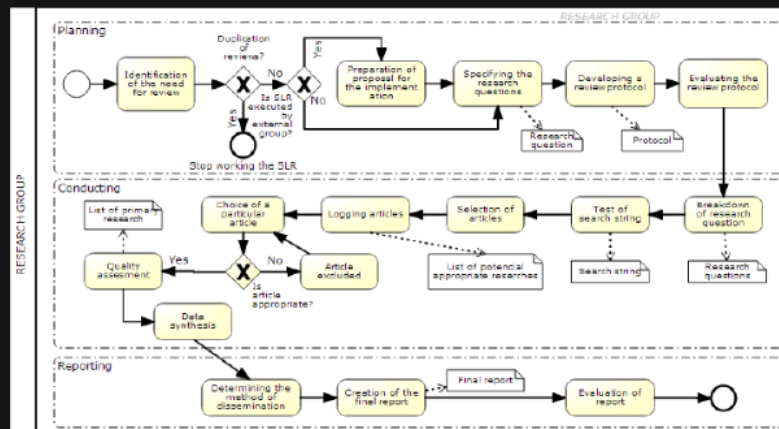
# COMMENT MODÉLISER



UML



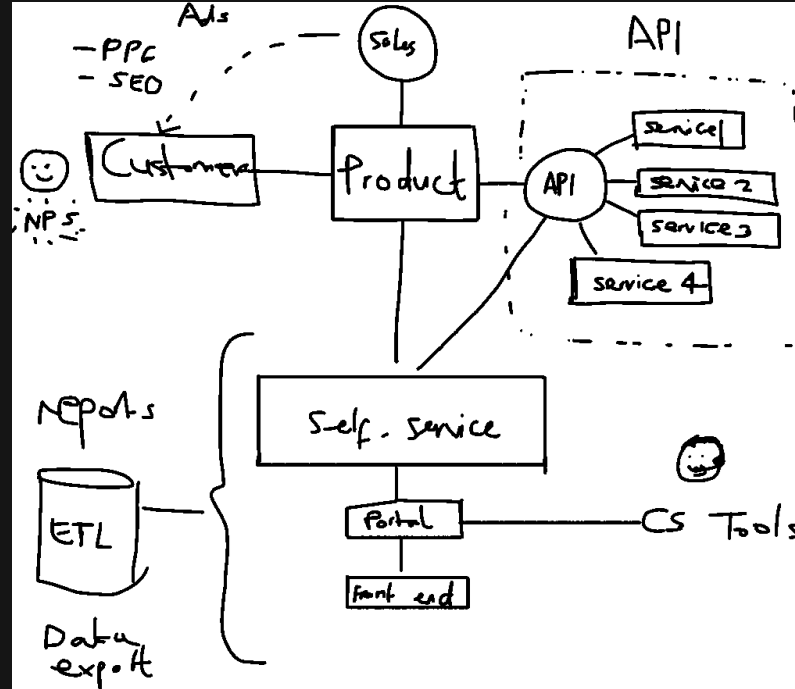
ArchiMate



BPMN

MERISE, SysML, DMN, IDEF, et bien d'autres...

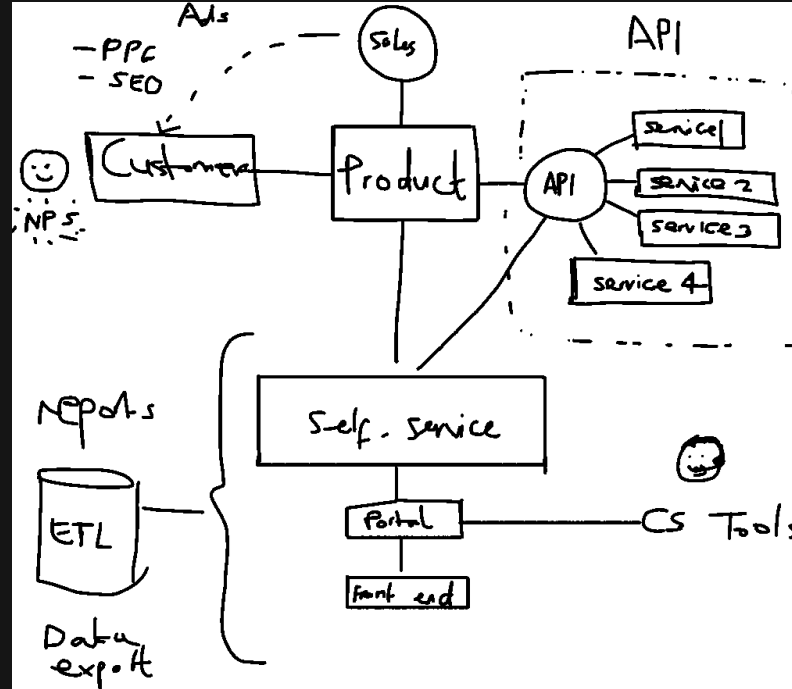
# / COMMENT MODÉLISER



Richard Holmes, The PM's Guide to Drawing on a Whiteboard

BOI-BOITE

# / COMMENT MODÉLISER



Richard Holmes, The PM's Guide to Drawing on a Whiteboard

BOI-BOITE



Besoin de communiquer sur l'organisation d'un système

# / LE C4 MODEL

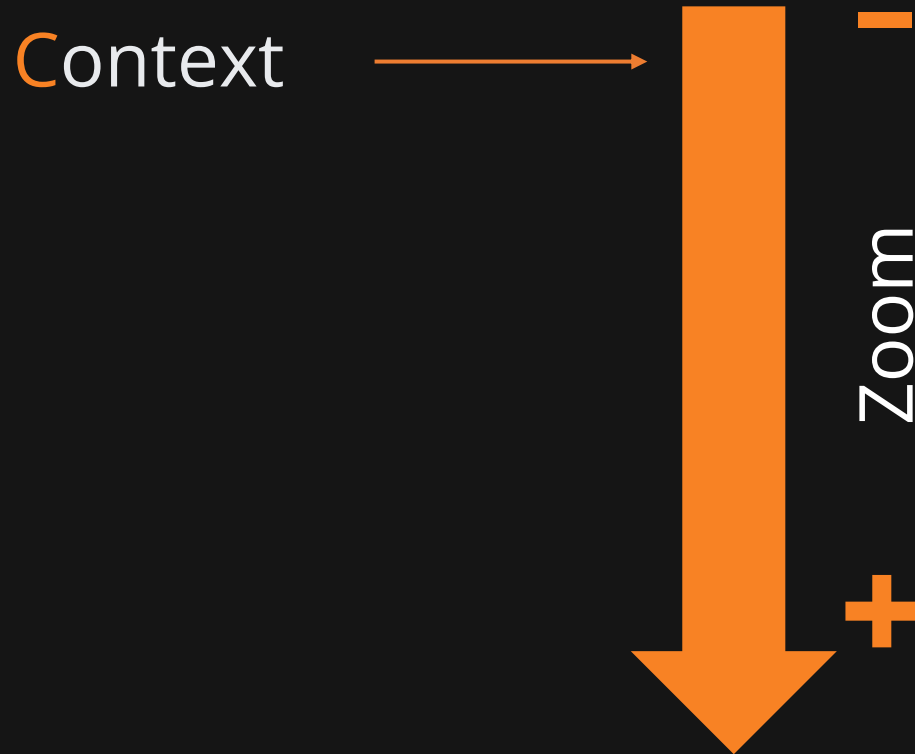
4 Niveaux d'abstraction / Diagrammes



Simon Brown  
[c4model.com](http://c4model.com)

# / LE C4 MODEL

## 4 Niveaux d'abstraction / Diagrammes



- Système logiciel dans son environnement
- Exemple : Domaine applicatif



# / LE C4 MODEL

## 4 Niveaux d'abstraction / Diagrammes

Context

Containers



- Les « blocs techniques » du système logiciel
- Unité « exécutable »
- Aspects technologiques
- Exemples : Web app, SPA, Batch, BDD



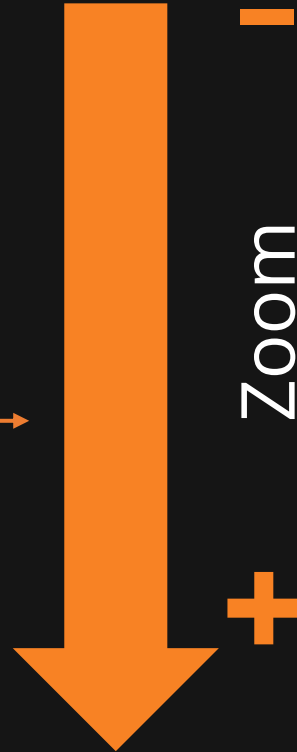
# / LE C4 MODEL

## 4 Niveaux d'abstraction / Diagrammes

Context

Containers

Components →



- Découpage d'un container
- Groupe cohérent de fonctionnalités
- Exemples : Ensemble de classes, package ?  
Namespace ? JAR ? DLL ?

# / LE C4 MODEL

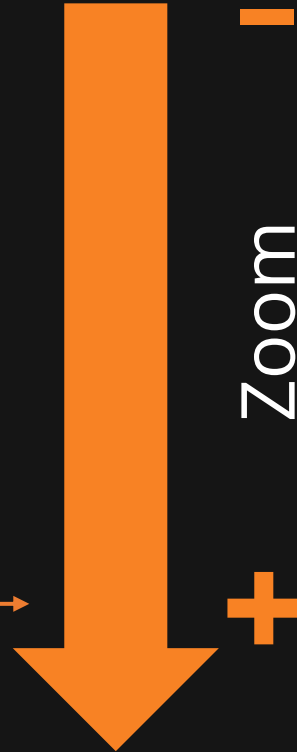
## 4 Niveaux d'abstraction / Diagrammes

Context

Containers

Components

Code



Zoom



- Implémentation des composants
- Focus sur les zones clés
- Exemple : Diagramme de classe UML, Entité / Relation

# / LE C4 MODEL

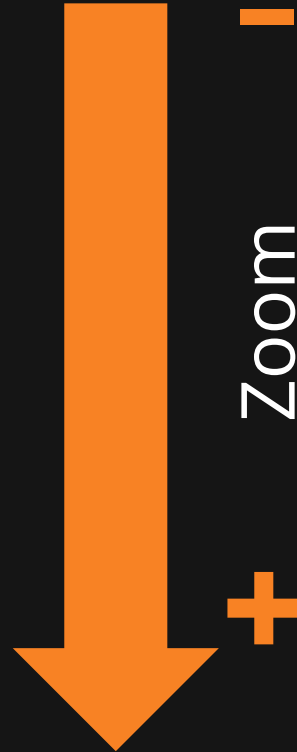
4 Niveaux d'abstraction / Diagrammes

Context

Containers

Components

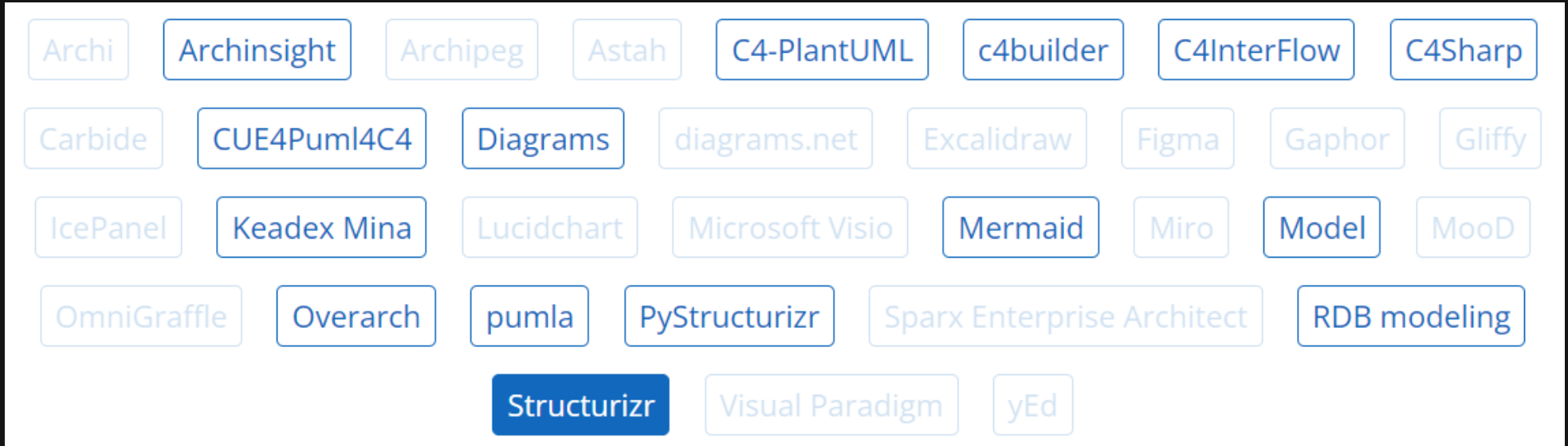
Code



Concepts indépendants

- d'un outil
- d'une notation

# / DIAGRAM AS CODE



<https://c4model.com/#Tooling>

# / DIAGRAM AS CODE

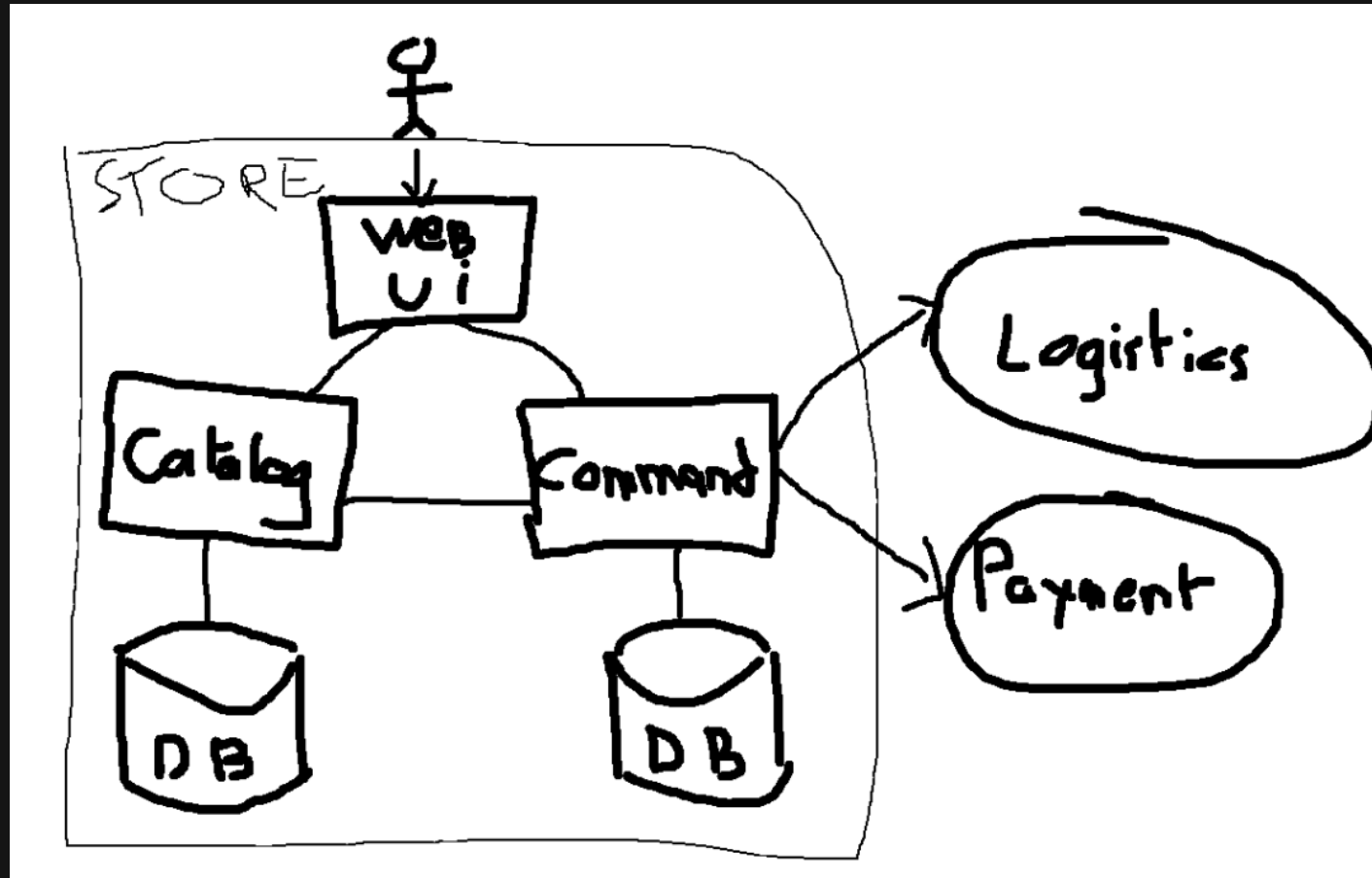


Structurizr

by Simon Brown

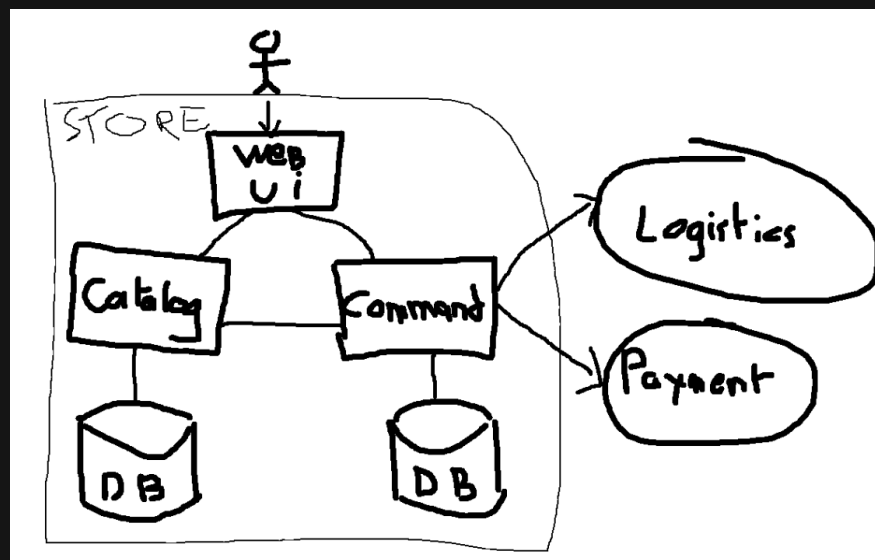
- Open source (Lite, On-Premise, CLI) et Cloud (SaaS)
- As code (modèle et vue), éléments réutilisables, Structurizr DSL
- Support de diagrammes complémentaires (landscape, dynamic, deployment)
- Fonctions avancées : Visualisations / rendus alternatifs, thèmes, doc as code, ...

# / CAS PRATIQUE

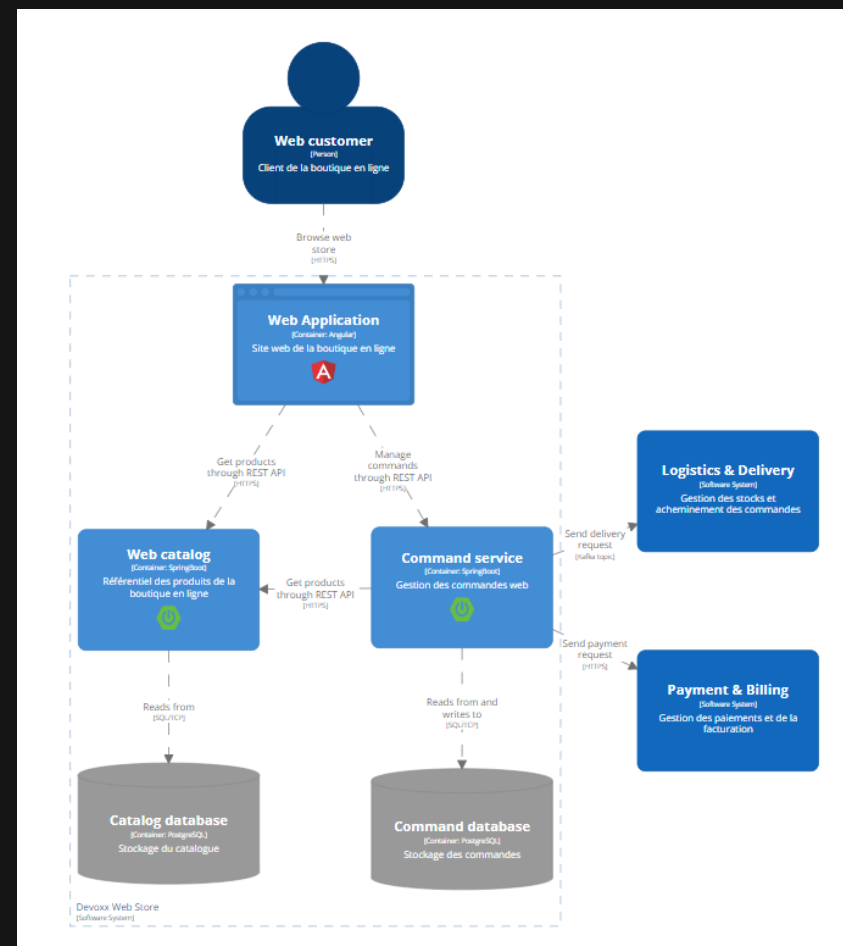


DevOxx Store – BOI-BOITE 2.0

# / CAS PRATIQUE



Devovx Store – BOI-BOITE 2.0



Devovx Store – C4 model (containers)



# / À EMPORTER



## On aime pour...

- Application multi « containers » / domaine applicatif / système d'information
- Equipes larges / multi-compétences
- Contexte en transformation



## Moins adapté pour...

- Monolithe isolé, contexte « simple »
- Besoin de modélisation riche / avancée (réseau, technique, processus, MDA...)



## Tips

- [C4] Lire la FAQ (<https://c4model.com/#FAQ>)
- [C4] Commencer petit (une appli, niveaux Context et Container, Structurizr Lite ou C4-PlantUML)
- [C4] Combiner C4 model avec d'autres types de modélisation si nécessaire
- [Structurizr] Utiliser la doc as code complémentaire plutôt que surcharger les descriptions
- [Structurizr] Découper le workspace pour garder une bonne maintenabilité
- [Structurizr] Utiliser les thèmes / styles pour simplifier la compréhension
- [Structurizr] Consulter les exemples sur Github (<https://github.com/structurizr/examples/tree/main/dsl>)
- [Structurizr] Utilisez la recherche de la doc Structurizr

# QUESTIONS RÉPONSES

DEVOXX FRANCE 2024



Jérôme Gauthier



jerga



jergaut



jerome.gauthier@soprasteria.com

