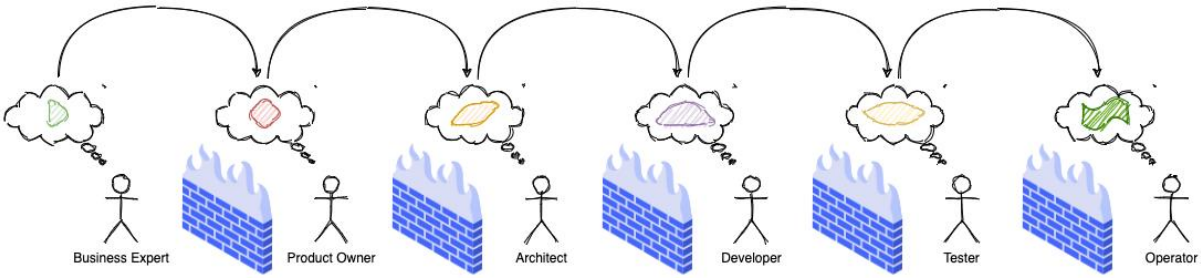
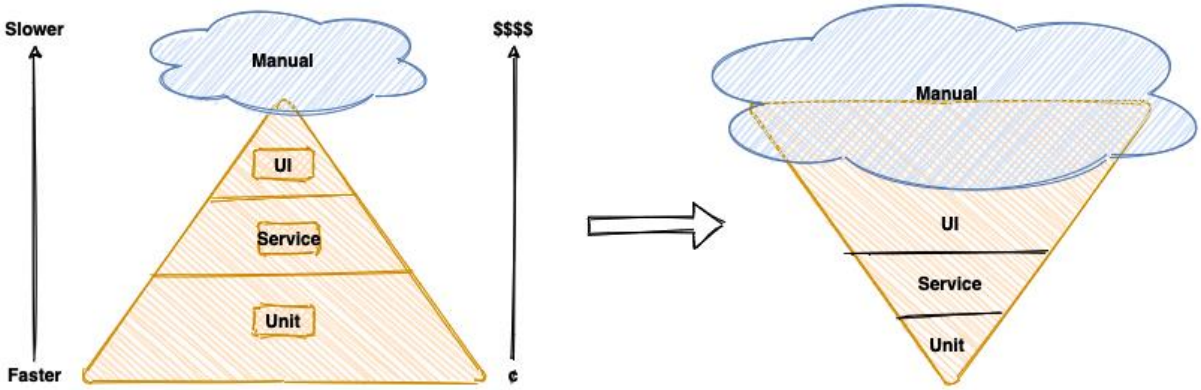
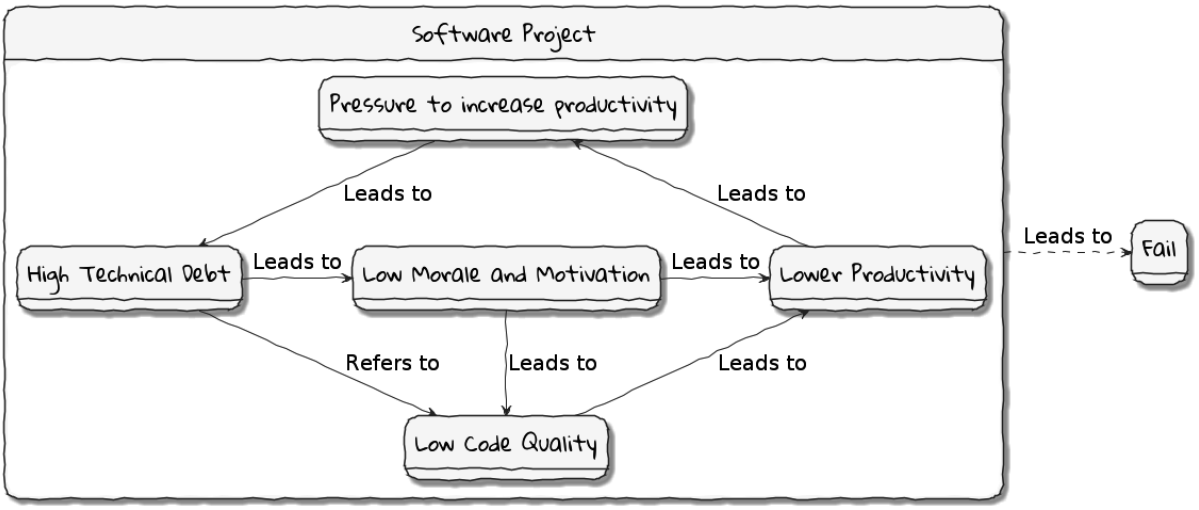
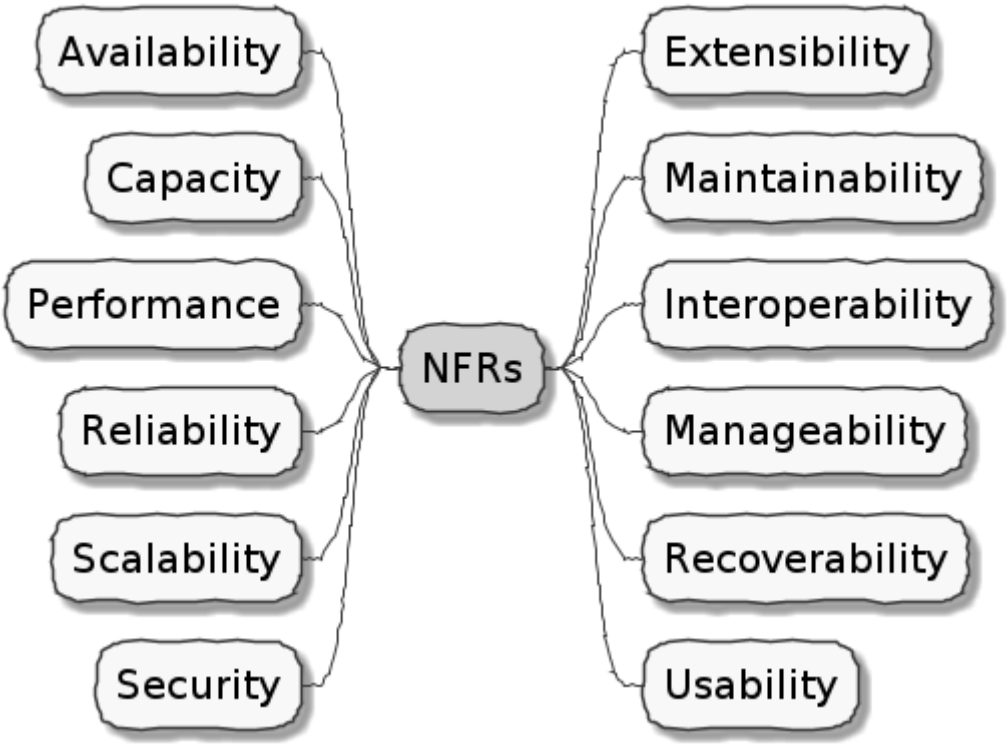
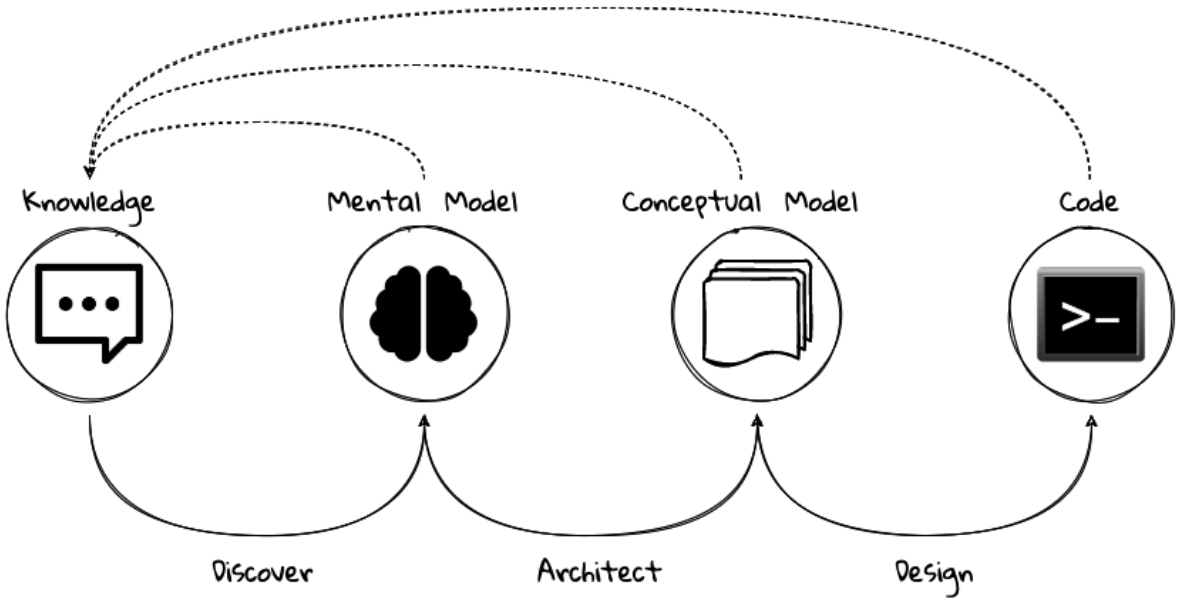
Chapter 1: The Rationale for Domain-Driven Design





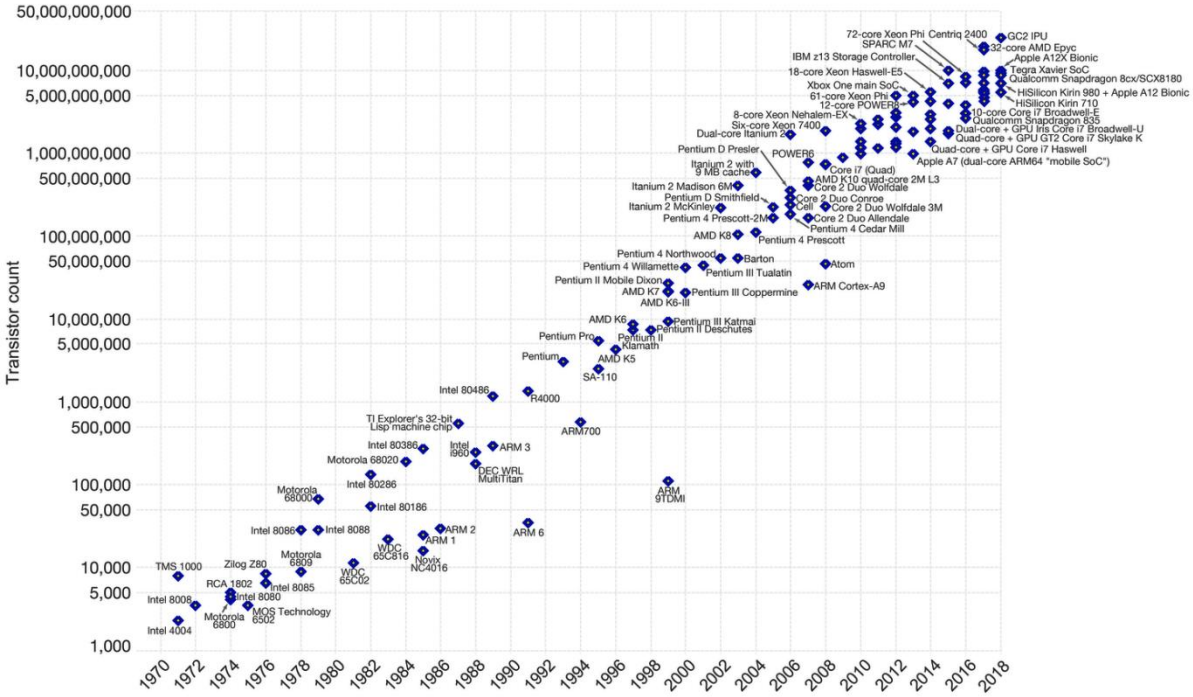



Availability
Capacity
Performance
Reliability
Scalability
Security
NFRs
Extensibility
Maintainability
Interoperability
Manageability
Recoverability
Usability

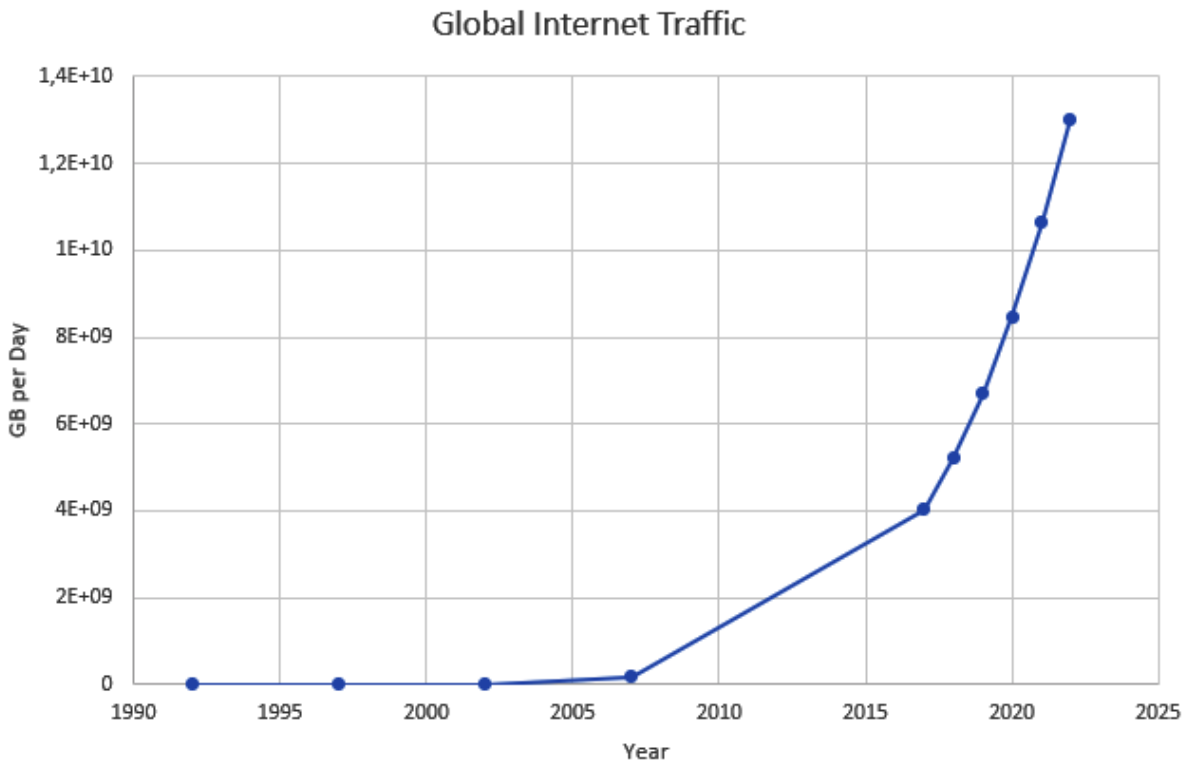

Knowledge
Mental Model
Conceptual Model
Code
Discover
Architect
Design

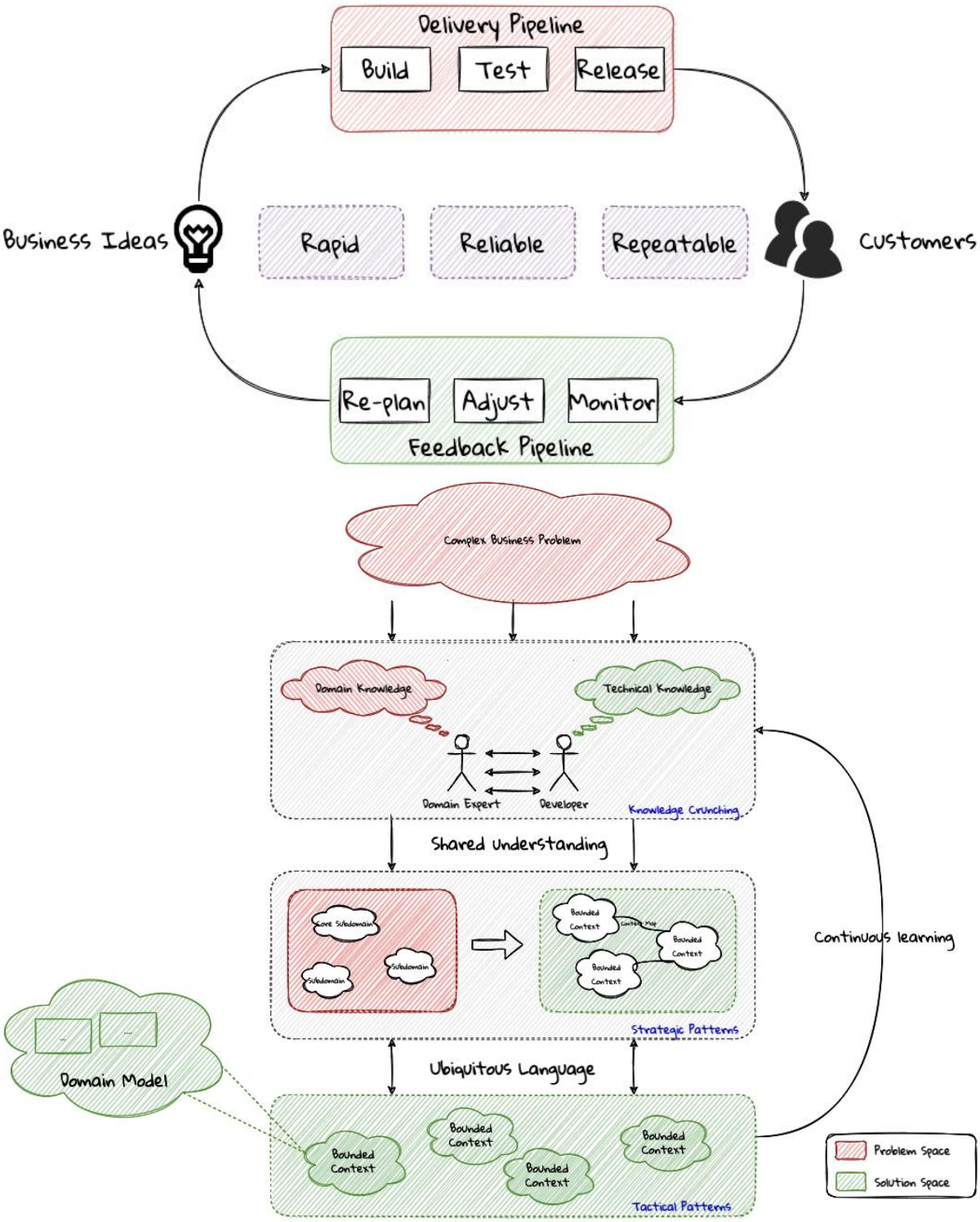
Moore's Law – The number of transistors on integrated circuit chips (1971-2018)

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are linked to Moore's law.



Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)
The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.




Delivery Pipeline
Build
Test
Release
Business Ideas
Rapid
Reliable
Repeatable
Customers
Re-plan
Adjust
Monitor
Feedback Pipeline
Complex Business Problem
Domain Knowledge
Technical Knowledge
Domain Expert
Developer
Knowledge Crunching
Shared Understanding
Core Subdomain
Subdomain
Subdomain
Bounded Context
Context Map
Bounded Context
Bounded Context
Strategic Patterns
Continuous learning
Ubiquitous Language
Domain Model
Bounded Context
Bounded Context
Bounded Context
Bounded Context
Tactical Patterns
Problem Space
Solution Space

Noun [edit]

domain (*plural* **domains**)

1. A geographic area owned or controlled by a single person or organization. [quotations ▼]

   *The king ruled his **domain** harshly.*

2. A field or sphere of activity, influence or expertise.

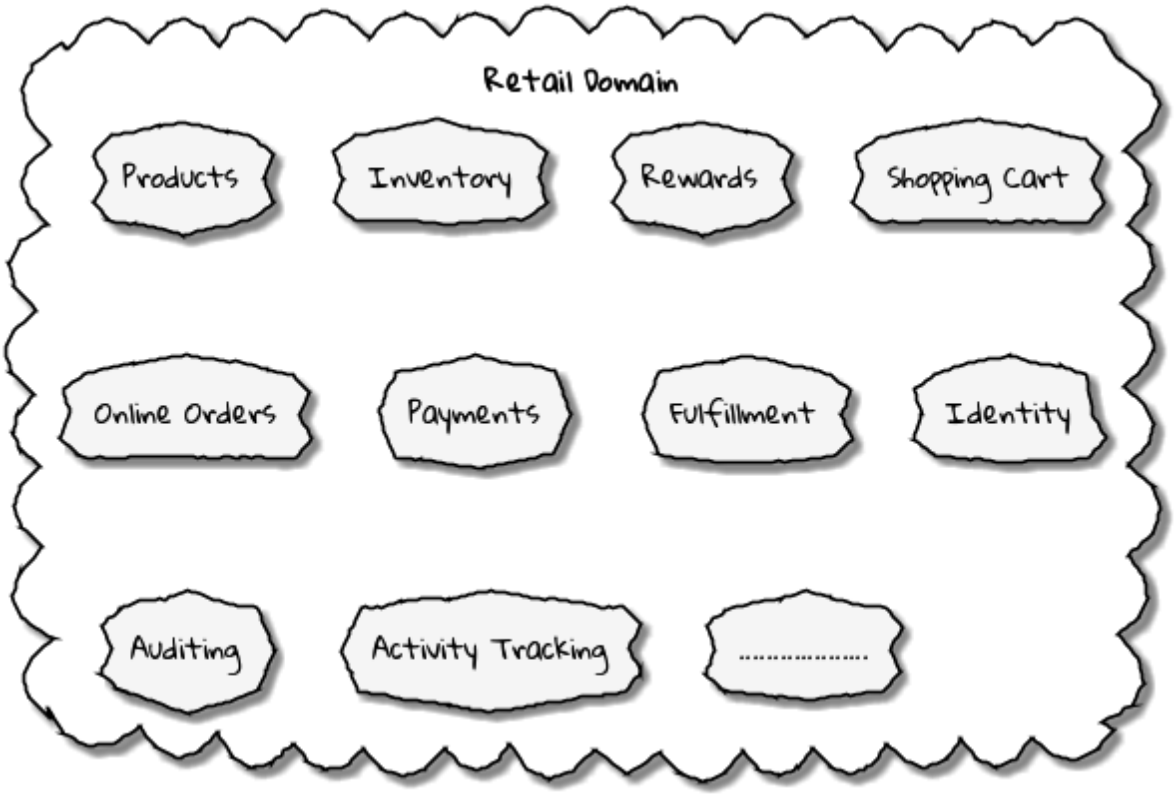
   *Dealing with complaints isn't really my **domain**: get in touch with customer services.*

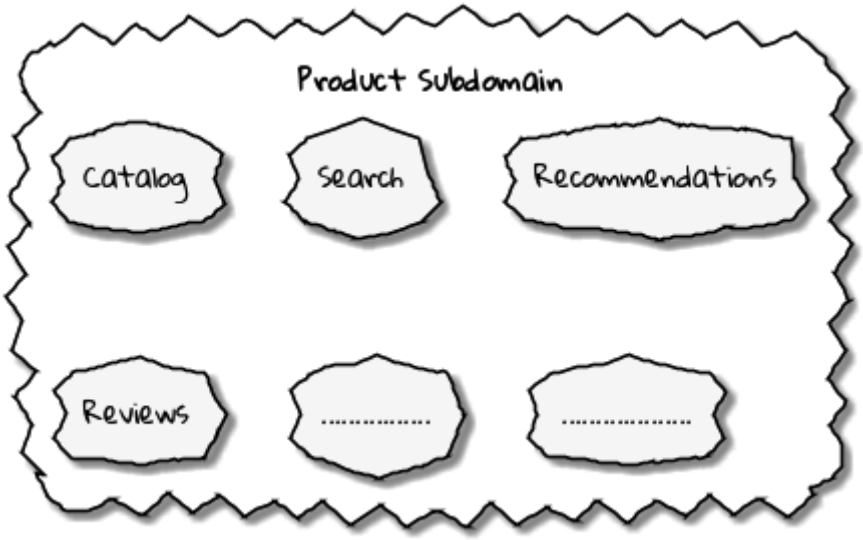
   *His **domain** is English history.*

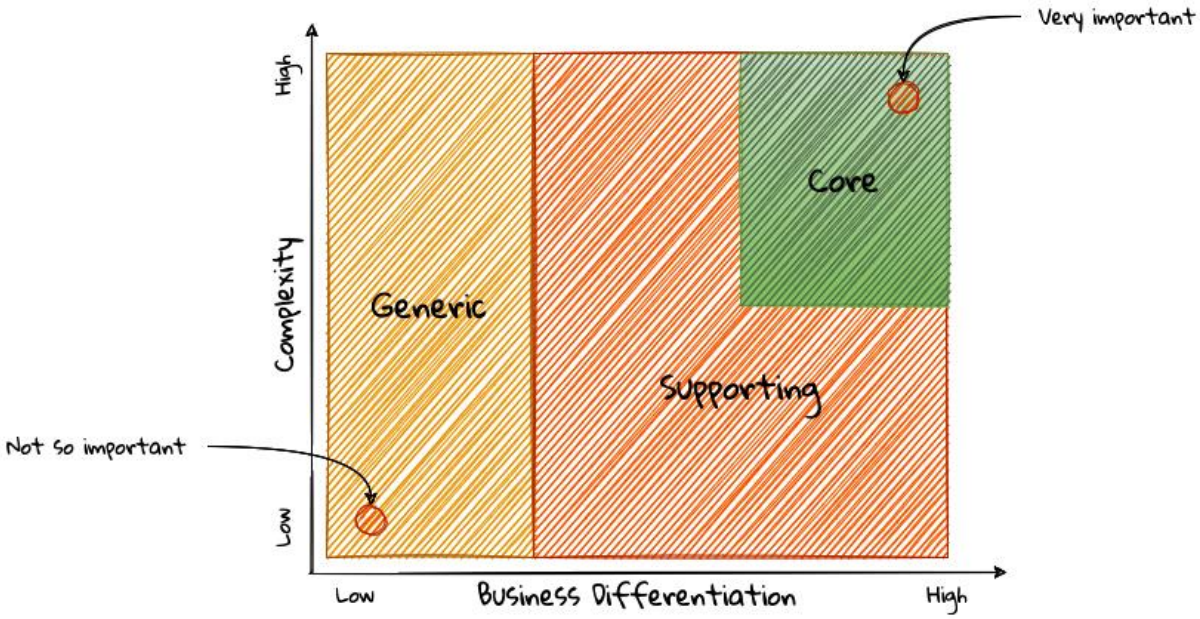
3. A group of related items, topics, or subjects. [quotations ▼]
4. (*mathematics*) The set of all possible mathematical entities (points) where a given function is defined.
5. (*mathematics*, *set theory*) The set of input (argument) values for which a function is defined.
6. (*mathematics*) A ring with no zero divisors; that is, in which no product of nonzero elements is zero.

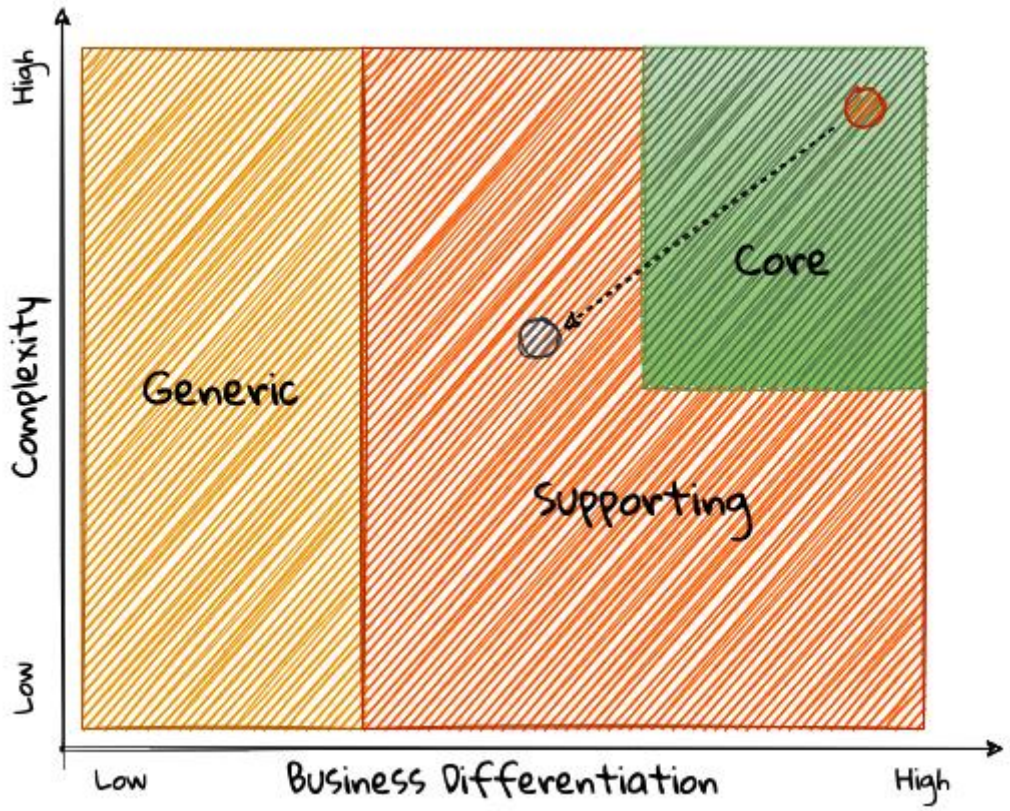
   Hyponym: integral domain

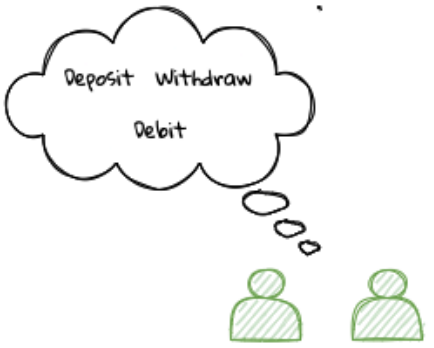
7. (*mathematics*, *topology*, *mathematical analysis*) An open and connected set in some topology. For example, the interval (0,1) as a subset of the real numbers.
8. (*computing*, *Internet*) Any DNS domain name, particularly one which has been delegated and has become representative of the delegated domain name and its subdomains. [quotations ▼]
9. (*computing*, *Internet*) A collection of DNS or DNS-like domain names consisting of a delegated domain name and all its subdomains.
10. (*computing*) A collection of information having to do with a domain, the computers named in the domain, and the network on which the computers named in the domain reside.
11. (*computing*) The collection of computers identified by a domain's domain names.
12. (*physics*) A small region of a magnetic material with a consistent magnetization direction.
13. (*computing*) Such a region used as a data storage element in a bubble memory.
14. (*data processing*) A form of technical metadata that represent the type of a data item, its characteristics, name, and usage. [quotations ▼]
15. (*taxonomy*) The highest rank in the classification of organisms, above kingdom; in the three-domain system, one of the taxa *Bacteria*, *Archaea*, or *Eukaryota*.
16. (*biochemistry*) A folded section of a protein molecule that has a discrete function; the equivalent section of a chromosome




Product Subdomain
Catalog
Search
Recommendations
Reviews
.....
.....


Very important
High
Complexity
Low
Core
Generic
Supporting
Not so important
Low
Business Differentiation
High





Domain Experts

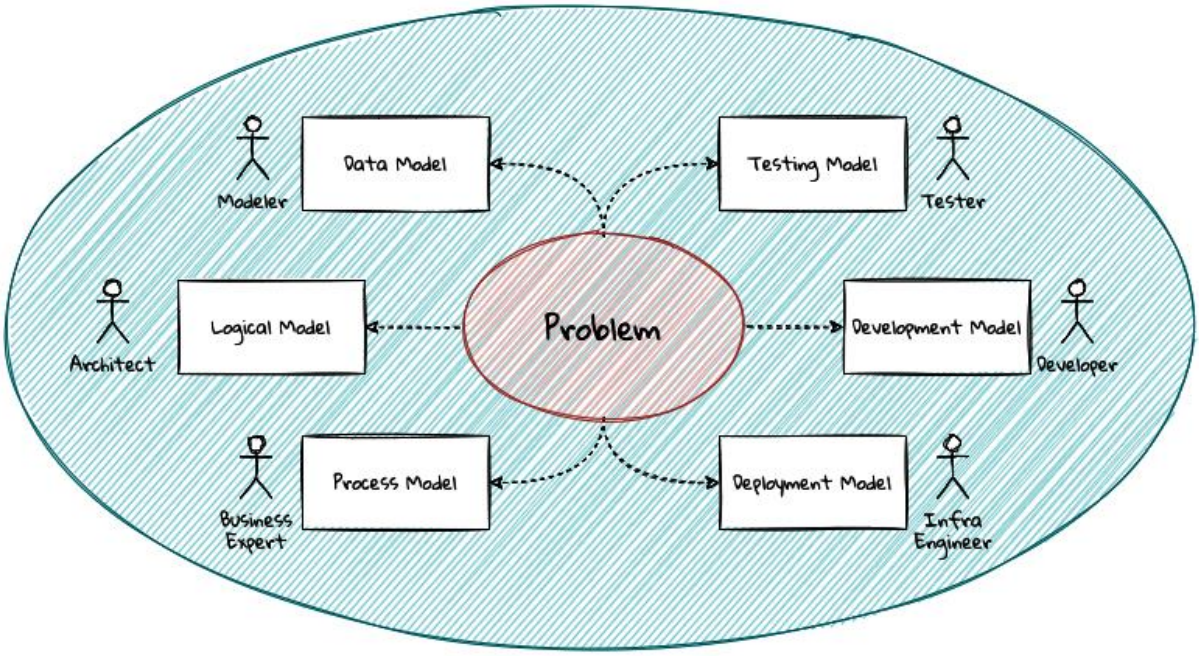
- Premium checking account feature
- Increase liquidity
- Increase customer base

Technical Experts

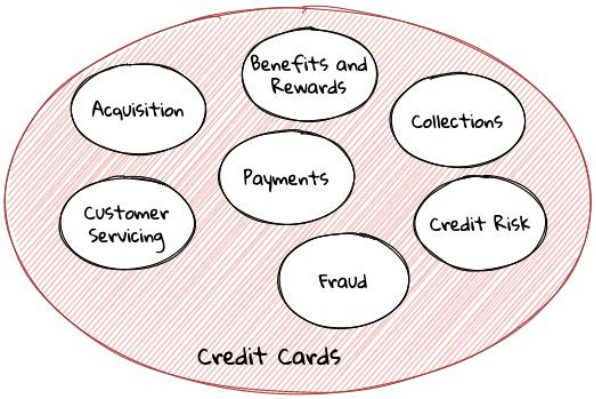
- Concurrency control
- ACID transactions
- Frameworks and design patterns

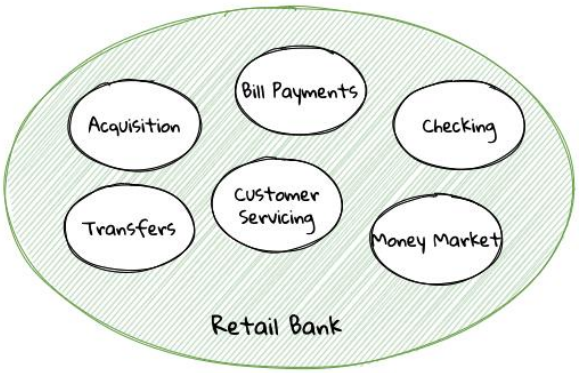
Problem domain

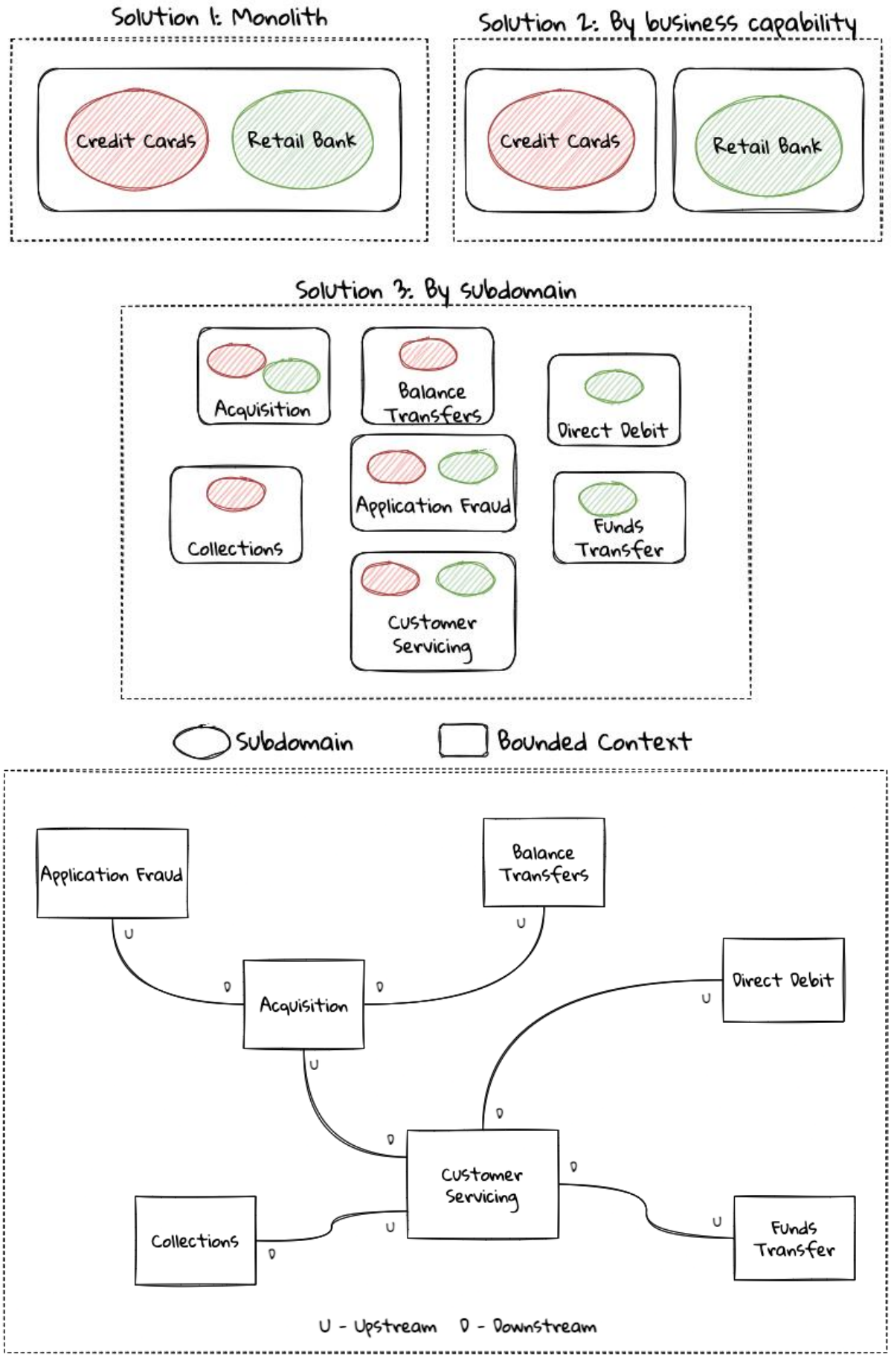
Solution domain

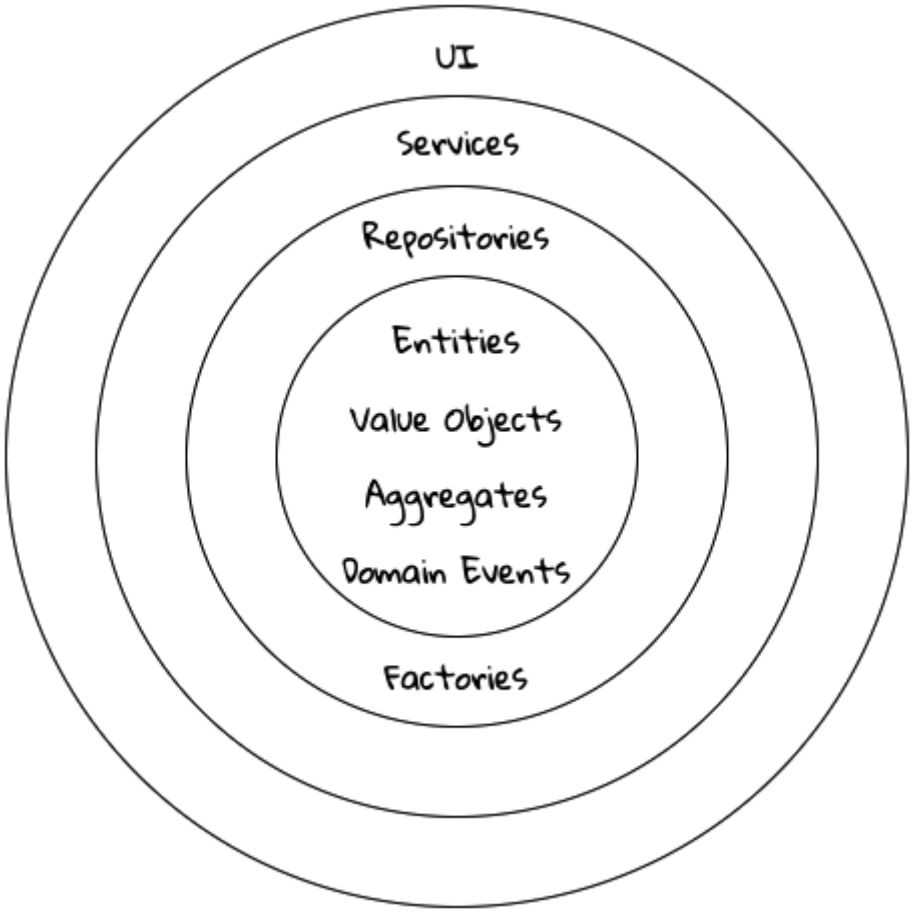

Modeler
Data Model
Testing Model
Tester
Architect
Logical Model
Problem
Development Model
Developer
Business Expert
Process Model
Deployment Model
Infra Engineer

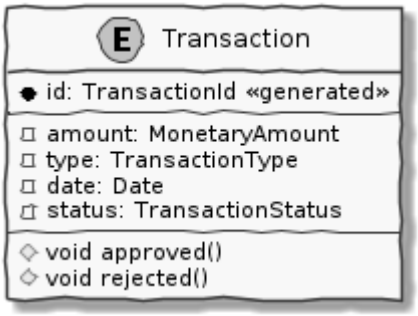

Problem domain
Ubiquitous Language
Solution domain

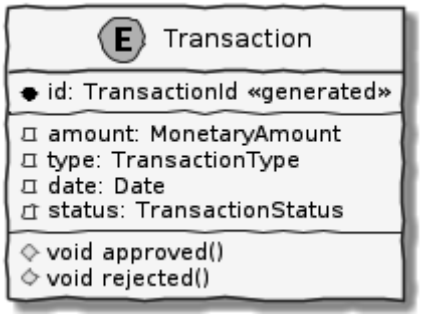

Acquisition
Benefits and Rewards
Collections
Payments
Customer Servicing
Credit Risk
Fraud
Credit Cards

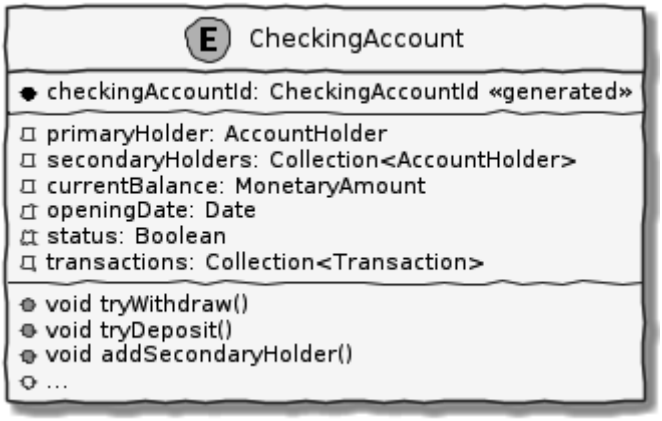

Acquisition
Bill Payments
Checking
Transfers
Customer Servicing
Money Market
Retail Bank

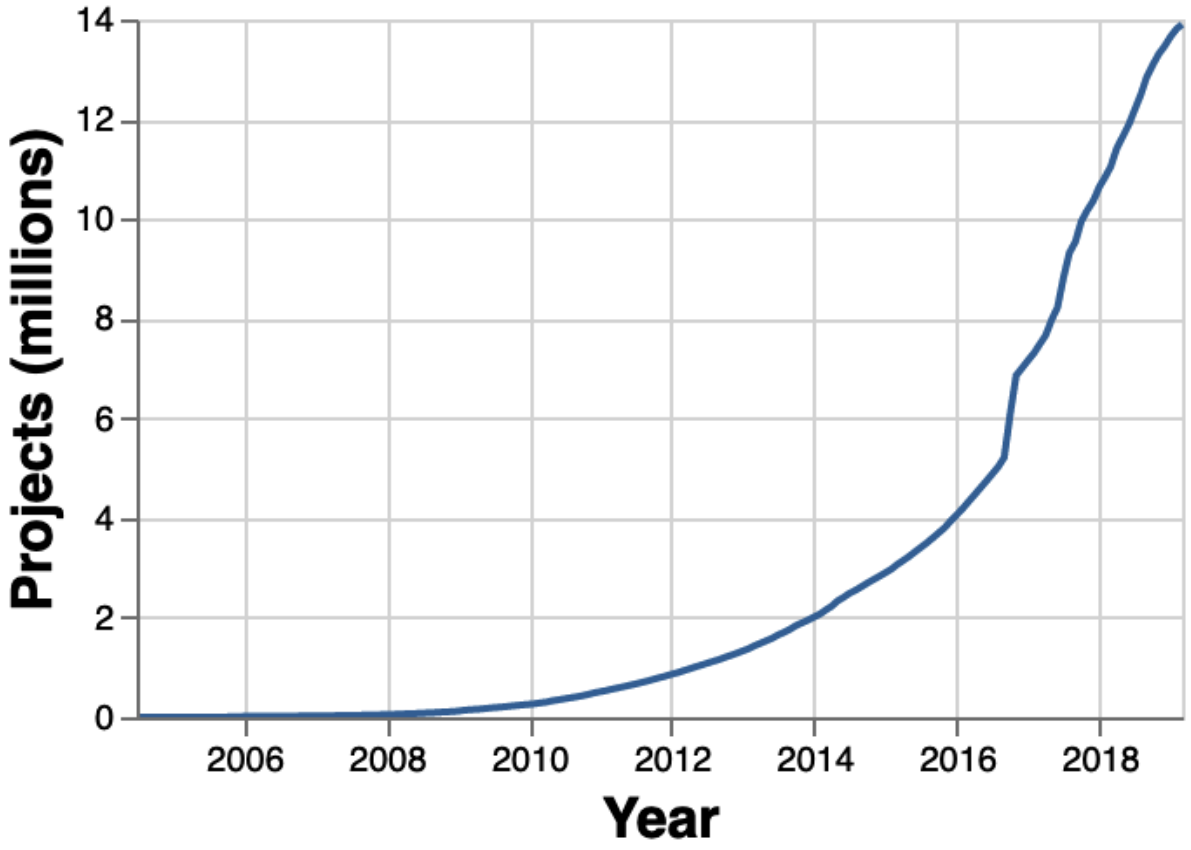

Solution 1: Monolith
Credit Cards
Retail Bank
Solution 2: By business capability
Credit Cards
Retail Bank
Solution 3: By subdomain
Acquisition
Balance Transfers
Direct Debit
Application Fraud
Collections
Funds Transfer
Customer Servicing
Subdomain
Bounded Context
Application Fraud
Balance Transfers
Acquisition
Direct Debit
Customer Servicing
Collections
Funds Transfer
U - Upstream D - Downstream

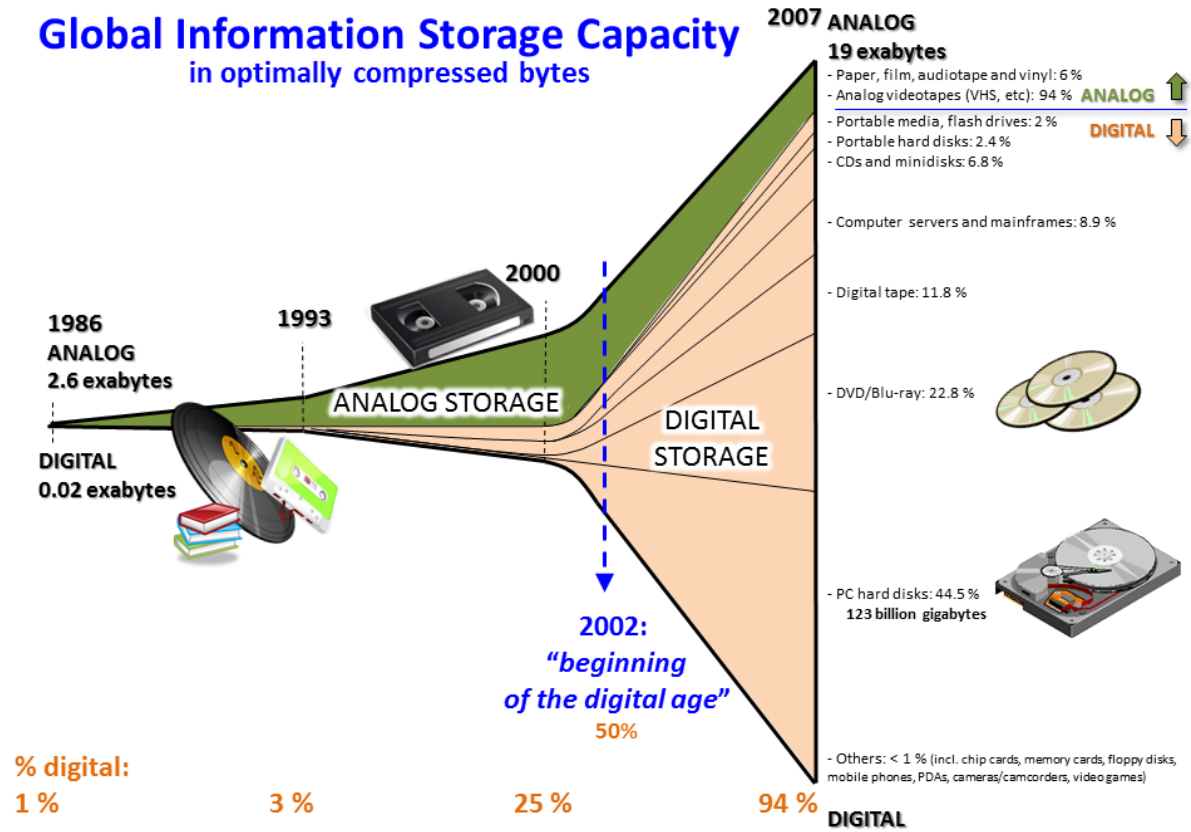

UI
Services
Repositories
Entities
Value Objects
Aggregates
Domain Events
Factories


E Transaction
id: TransactionId «generated»
amount: MonetaryAmount
type: TransactionType
date: Date
status: TransactionStatus
void approved()
void rejected()


E Transaction
id: TransactionId «generated»
amount: MonetaryAmount
type: TransactionType
date: Date
status: TransactionStatus
void approved()
void rejected()

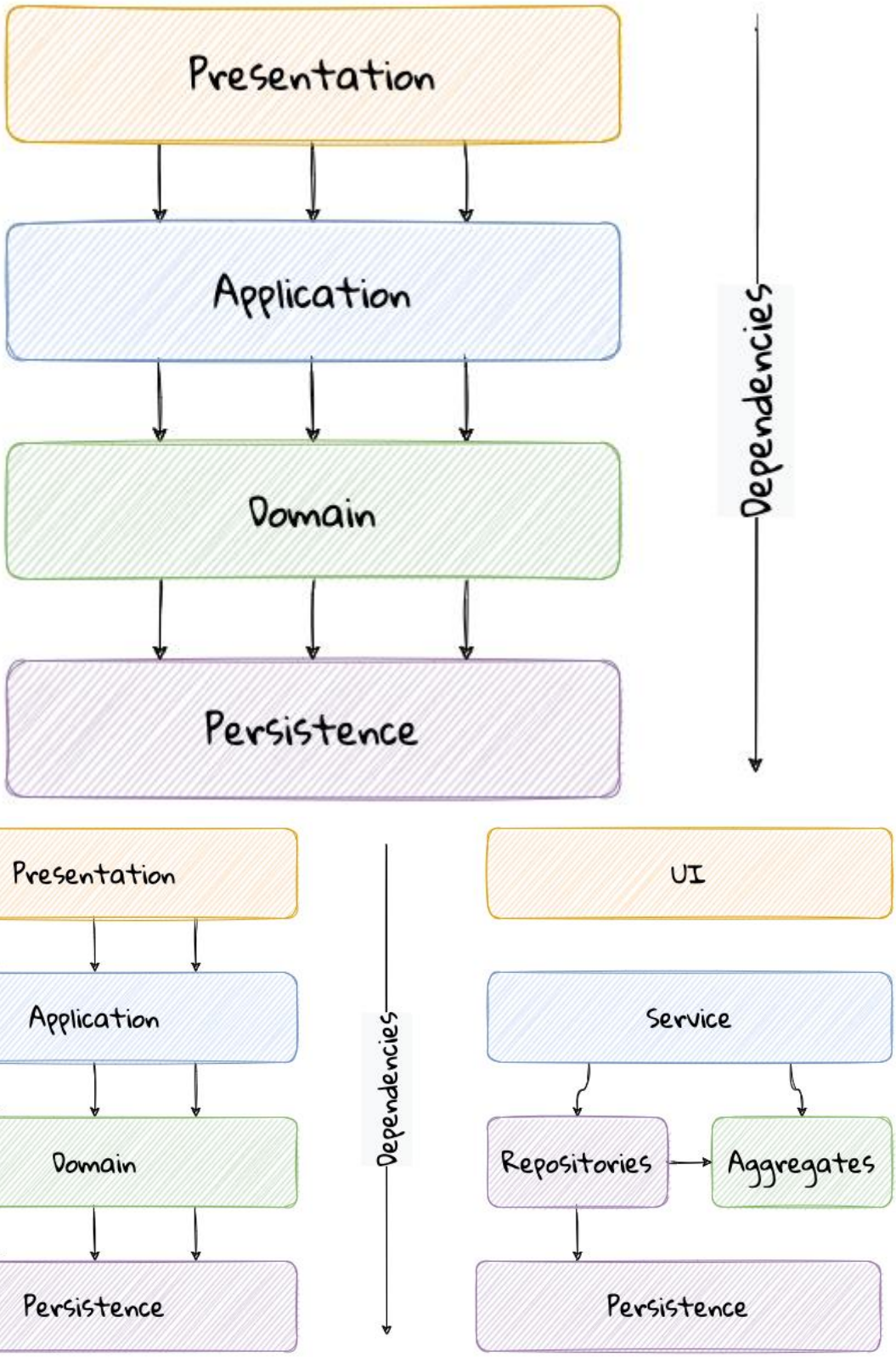

E CheckingAccount
checkingAccountId: CheckingAccountId «generated»
primaryHolder: AccountHolder
secondaryHolders: Collection<AccountHolder>
currentBalance: MonetaryAmount
openingDate: Date
status: Boolean
transactions: Collection<Transaction>
void tryWithdraw()
void tryDeposit()
void addSecondaryHolder()
...

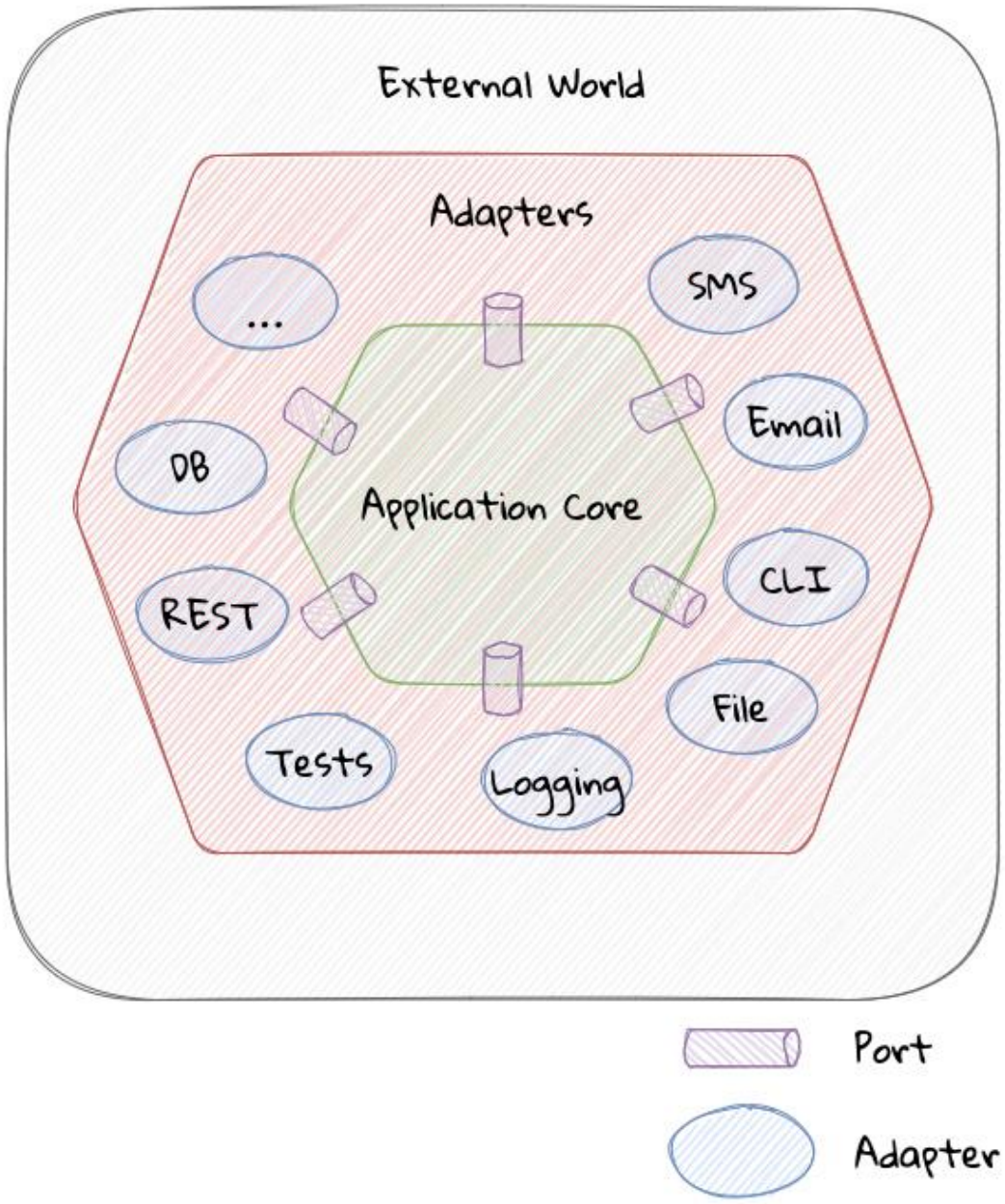


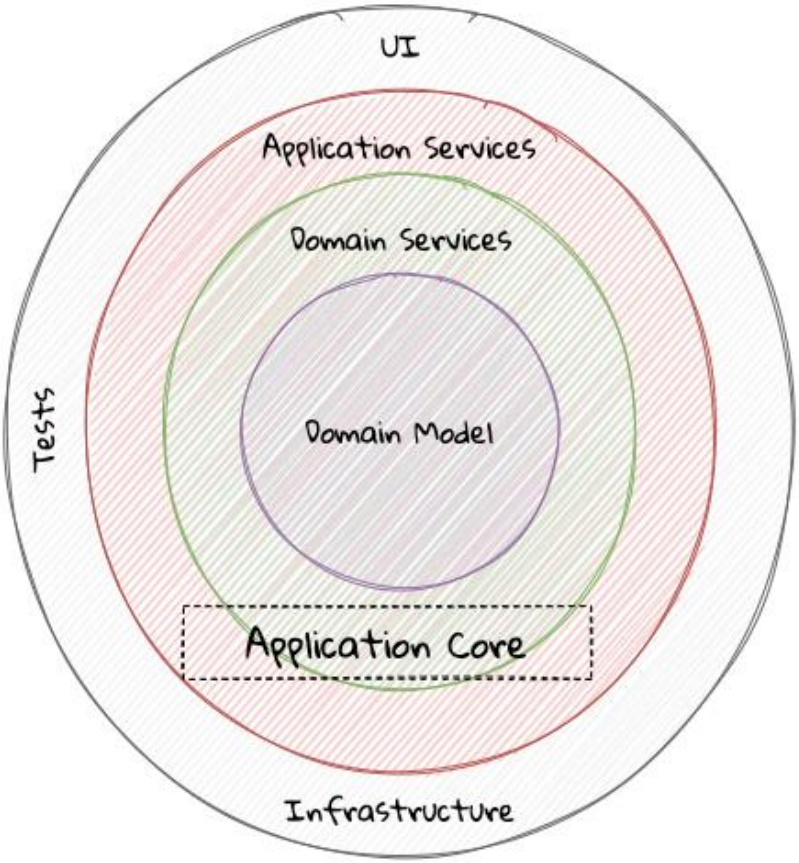


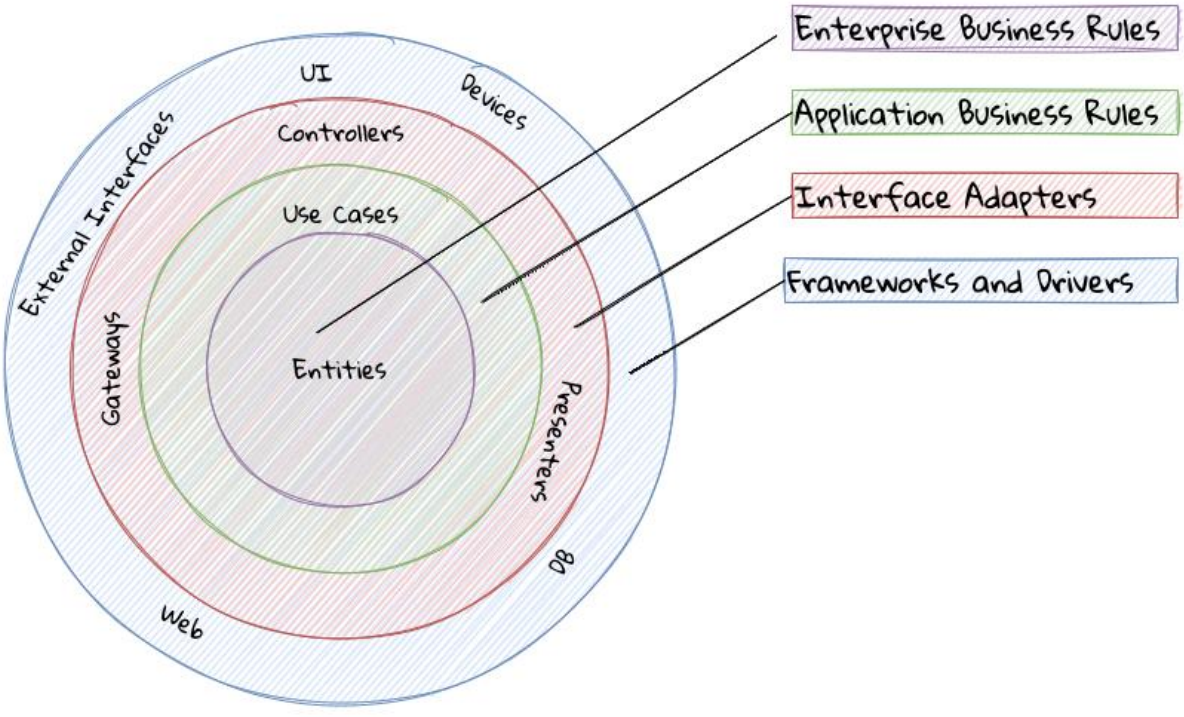
Source: Hilbert, M., & López, P. (2011). The World's Technological Capacity to Store, Communicate, and Compute Information. *Science*, 332(6025), 60 –65. http://www.martinhilbert.net/WorldInfoCapacity.html

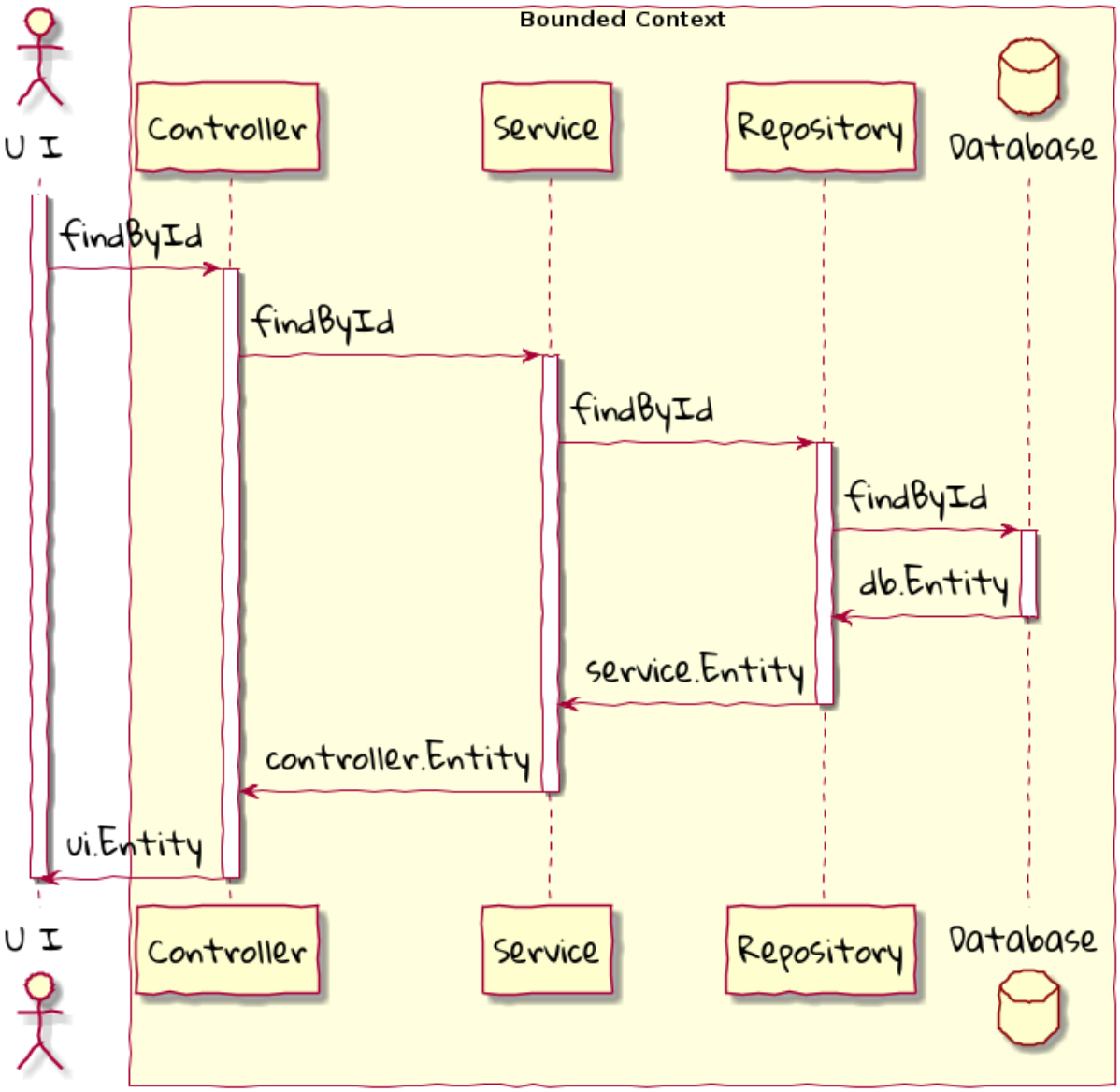
Chapter 2: The Mechanics of Domain-Driven Design

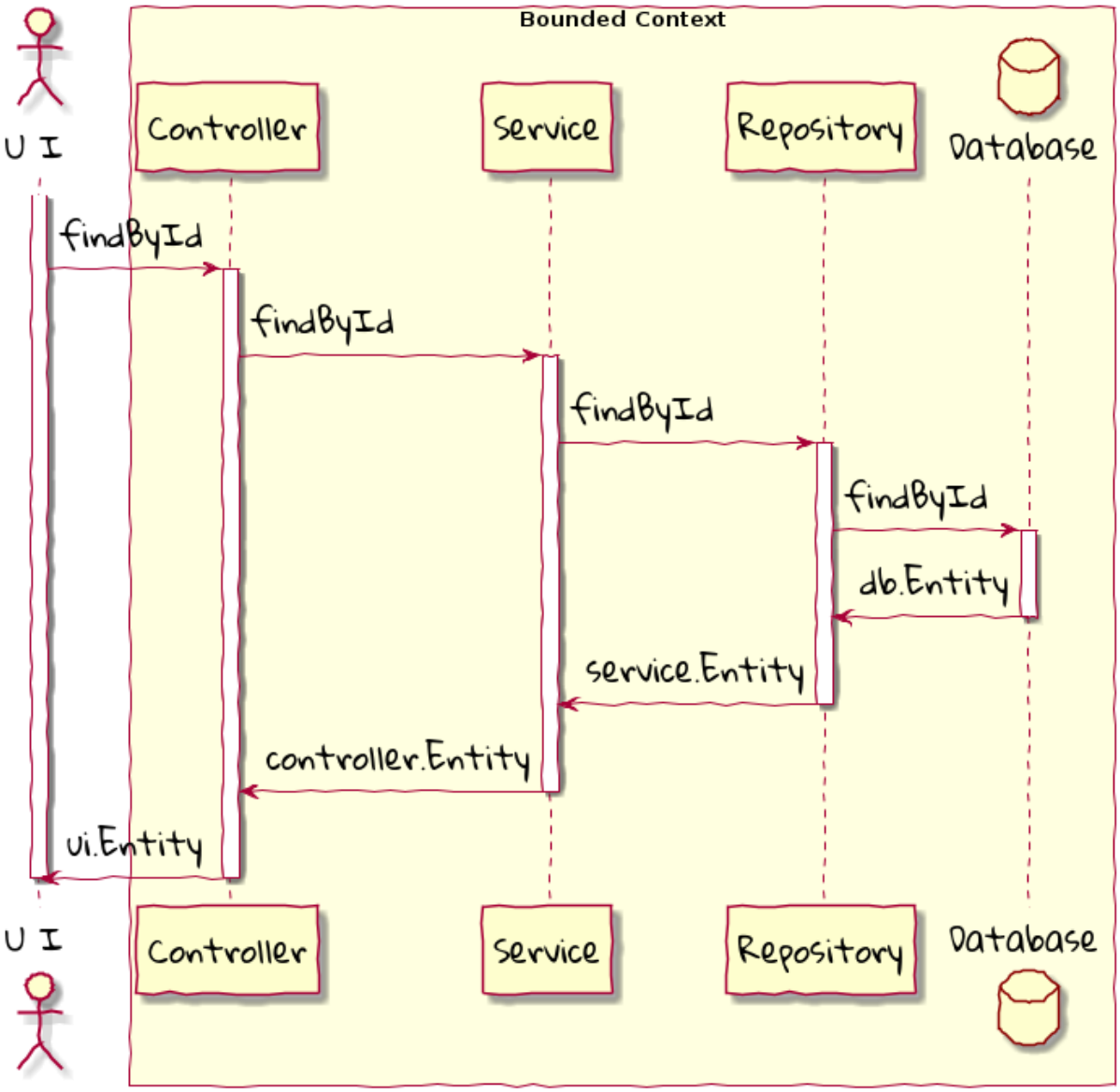


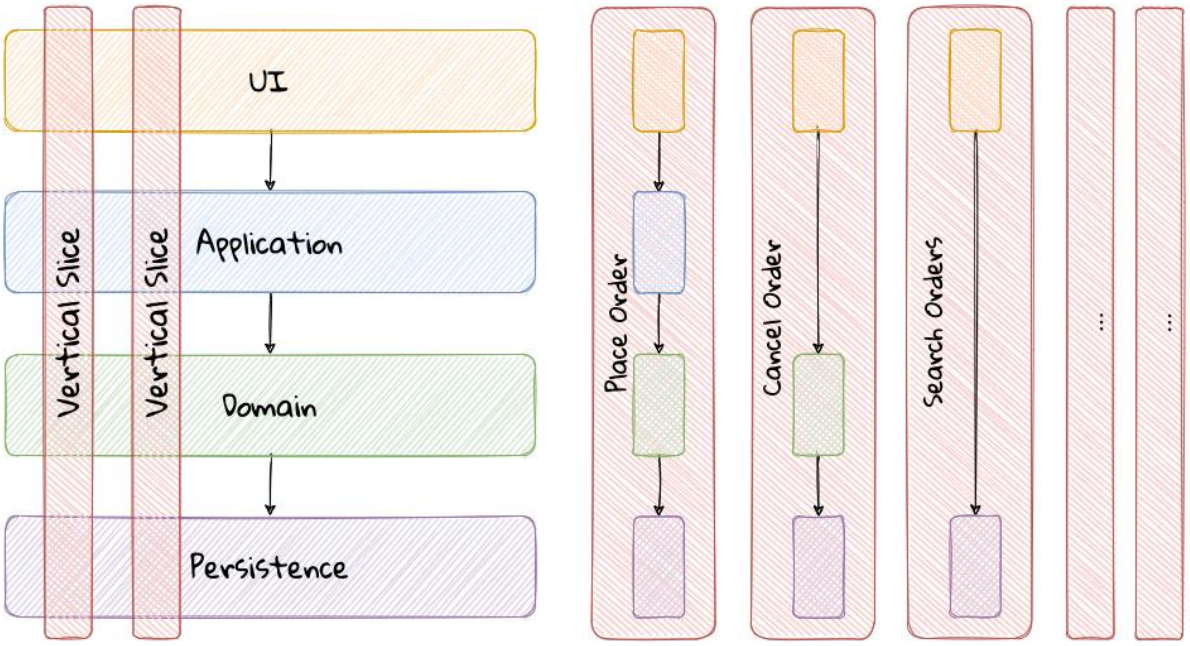

External World
Adapters
...
SMS
Email
DB
Application Core
CLI
REST
File
Tests
Logging
Port
Adapter

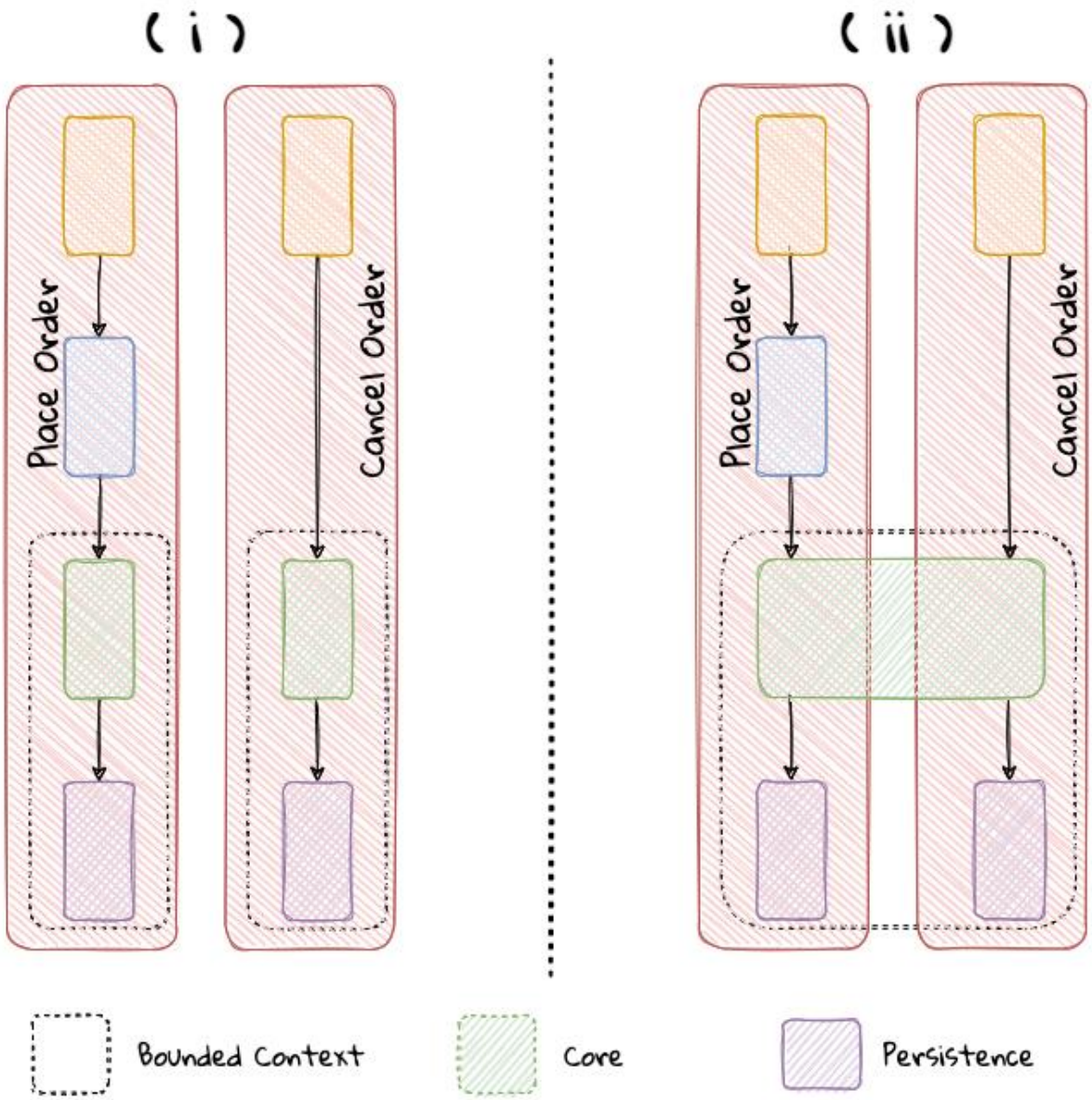

UI
Application Services
Domain Services
Domain Model
Application Core
Tests
Infrastructure
Web Service
DB
....

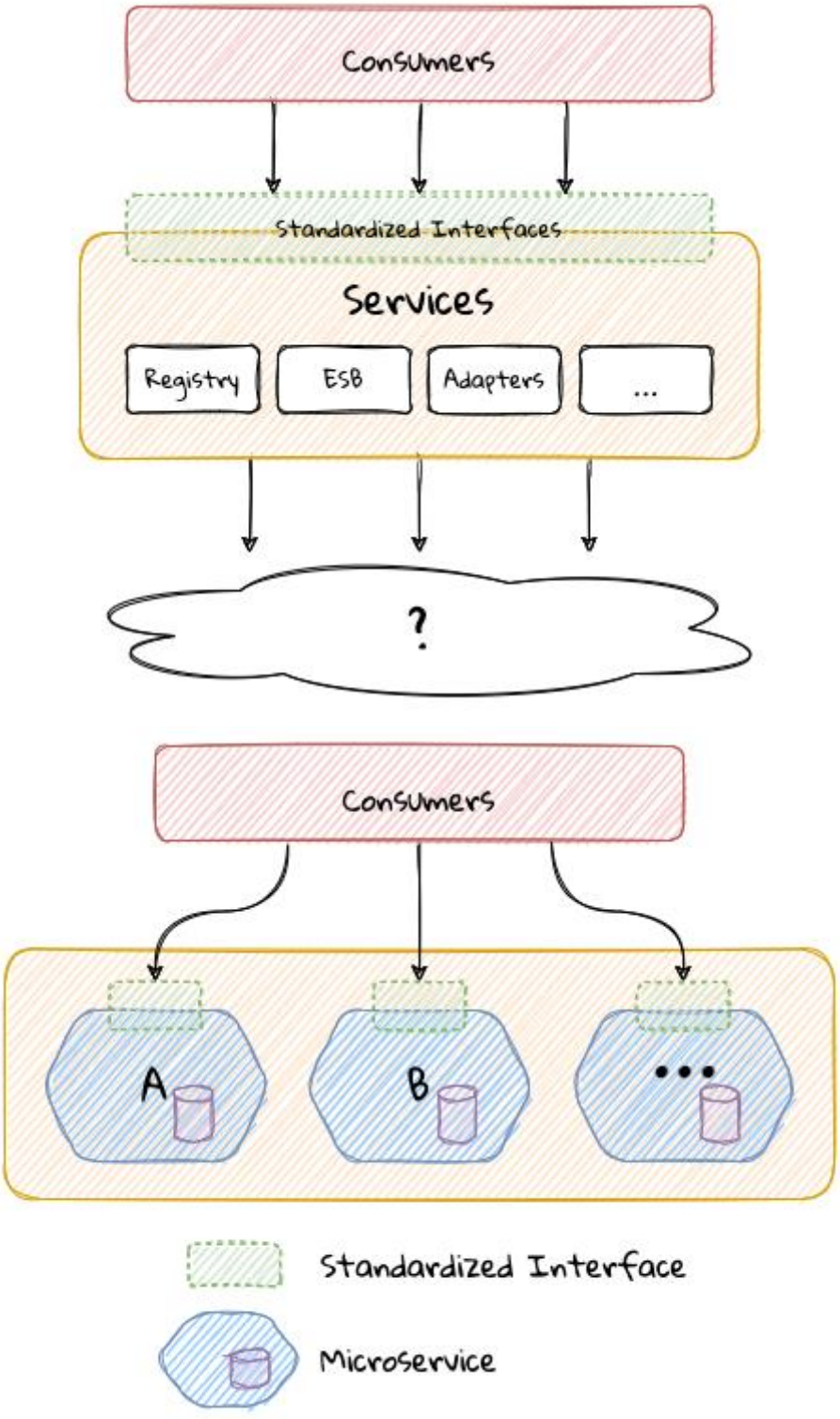

UI
Devices
Controllers
Use Cases
Entities
External Interfaces
Gateways
Presenters
DB
Web
Enterprise Business Rules
Application Business Rules
Interface Adapters
Frameworks and Drivers

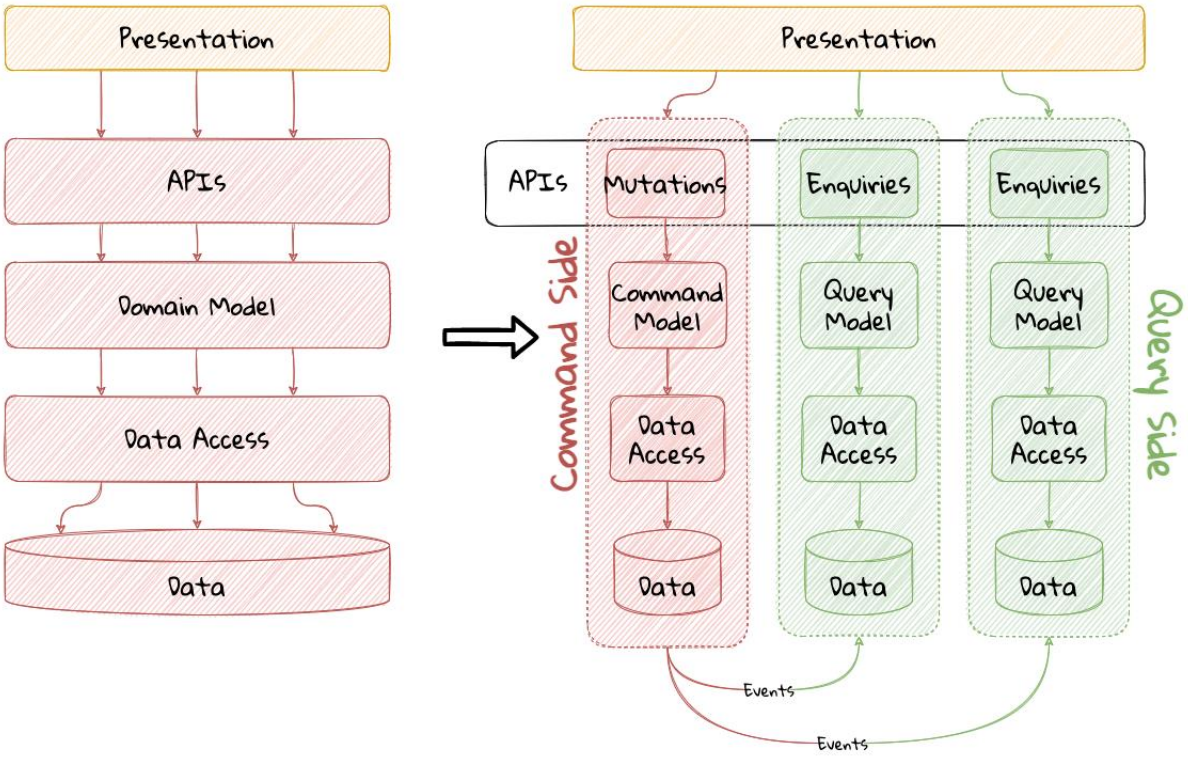

Bounded Context
U I
Controller
Service
Repository
Database
findById
findById
findById
findById
db.Entity
service.Entity
controller.Entity
ui.Entity
U I
Controller
Service
Repository
Database


Bounded Context
U I
Controller
Service
Repository
Database
findById
findById
findById
findById
db.Entity
service.Entity
controller.Entity
ui.Entity
U I
Controller
Service
Repository
Database

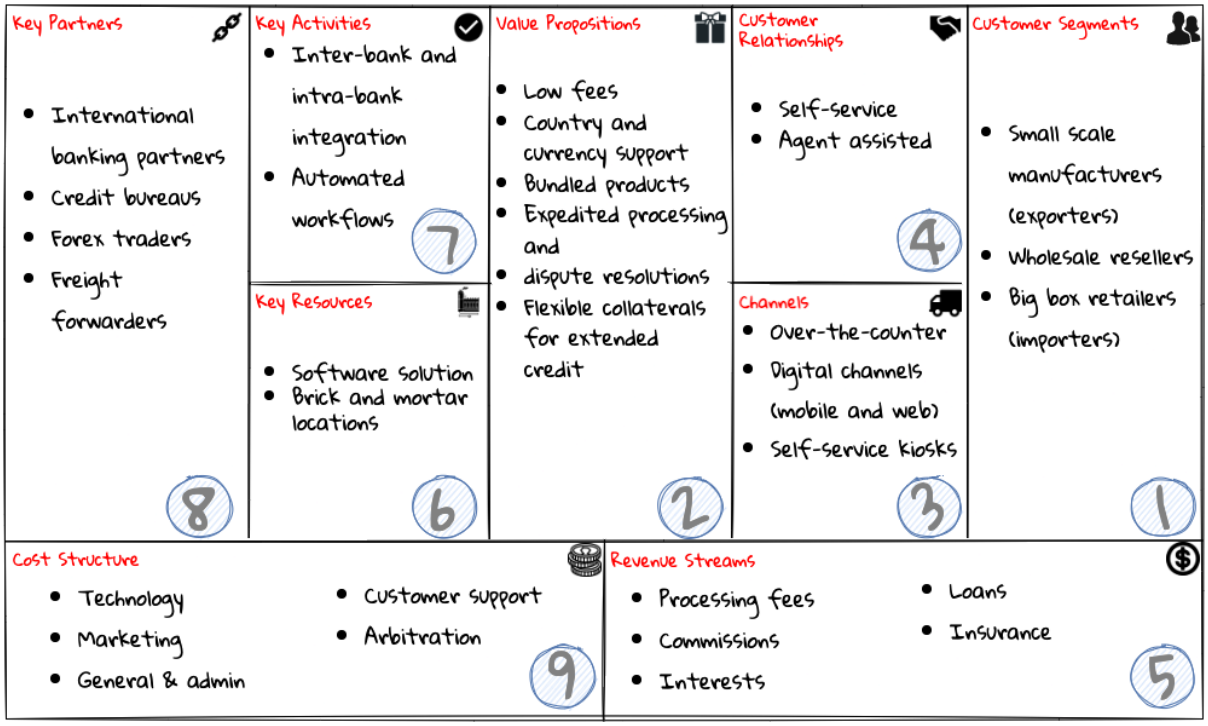

UI
Application
Domain
Persistence
Vertical Slice
Vertical Slice
Place Order
Cancel Order
Search Orders
...
...

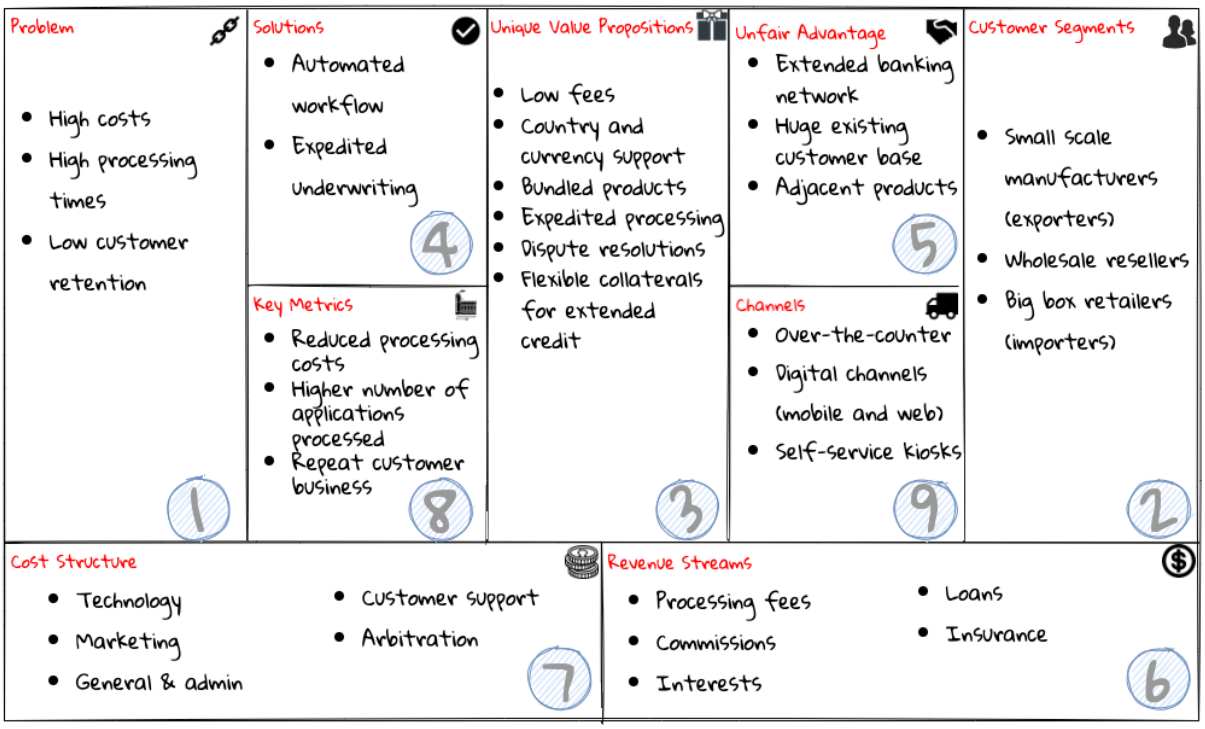

(i)
(ii)
Place Order
Cancel Order
Place Order
Cancel Order
Bounded Context
Core
Persistence

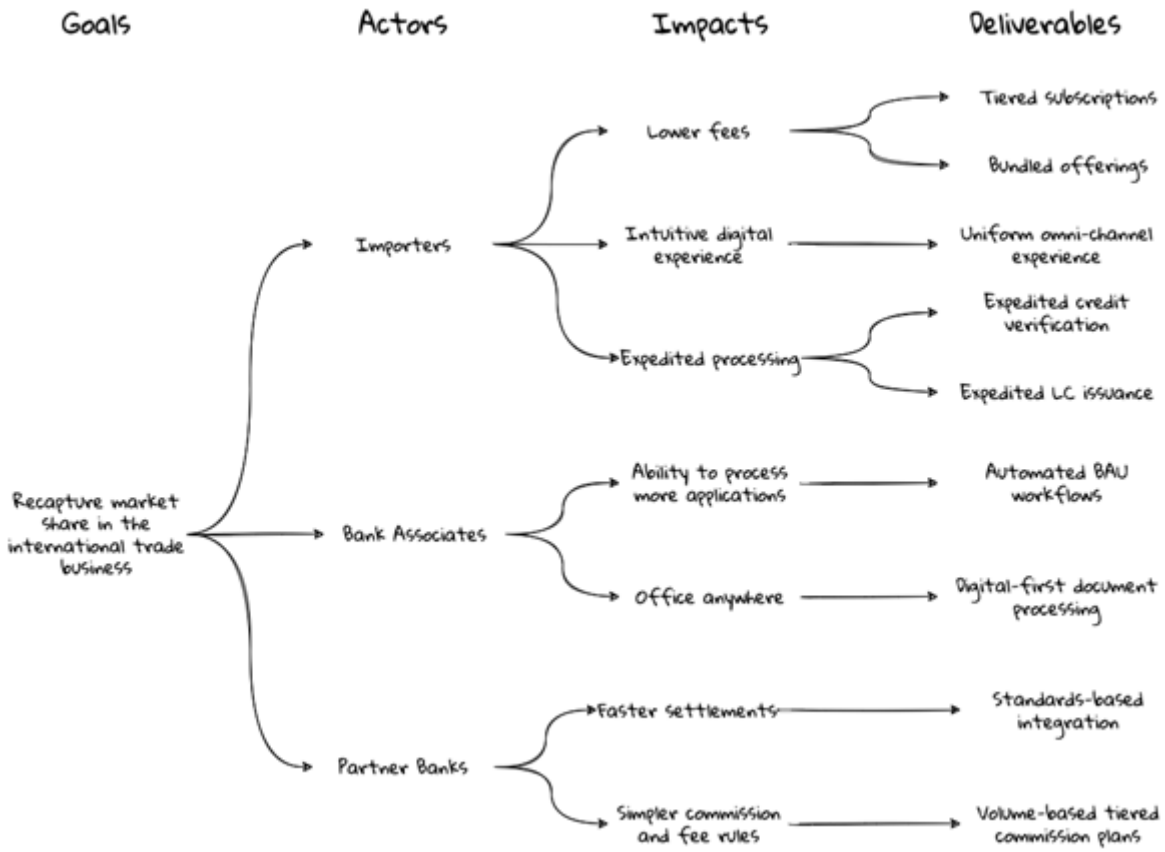

Consumers
Standardized Interfaces
Services
Registry
ESB
Adapters
...
?
Consumers
A
B
...
Standardized Interface
Microservice


Presentation
APIs
Domain Model
Data Access
Data
Presentation
APIs
Mutations
Enquiries
Enquiries
Command Side
Command Model
Query Model
Query Model
Query Side
Data Access
Data Access
Data Access
Data
Data
Data
Events
Events

Chapter 3: Where and How Does DDD Fit?







Wardley Mapping Canvas KP Bank International Trade Business

1. Purpose
What is your purpose? Why does this organization or project exist?

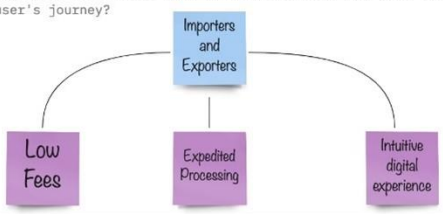
Recapture market share in the international trade business

2. Scope
What is it that you are mapping? What does it include? What does it not include?

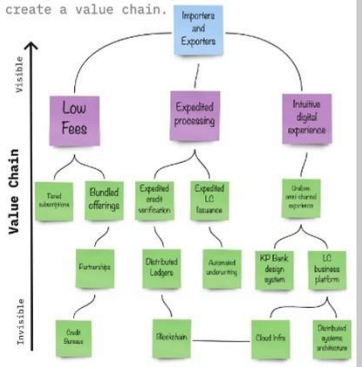
The LC issuance and application process

3. Users
Who uses or interacts with the thing you are mapping?

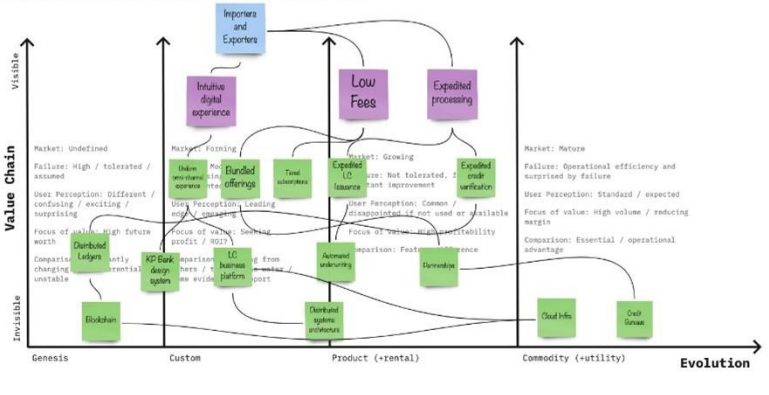
4. User Needs
Copy your users over. What do they need from you? What is each user's journey?

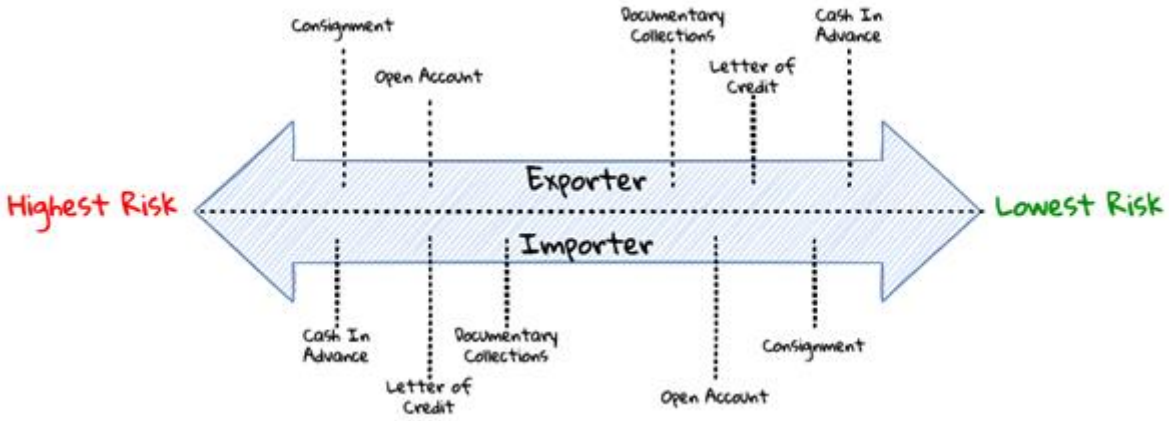


5. Value Chain
Copy the users over, with their needs underneath. What sorts of things do you need to be doing to fulfill those needs? Arrange them according to dependence — create a value chain.

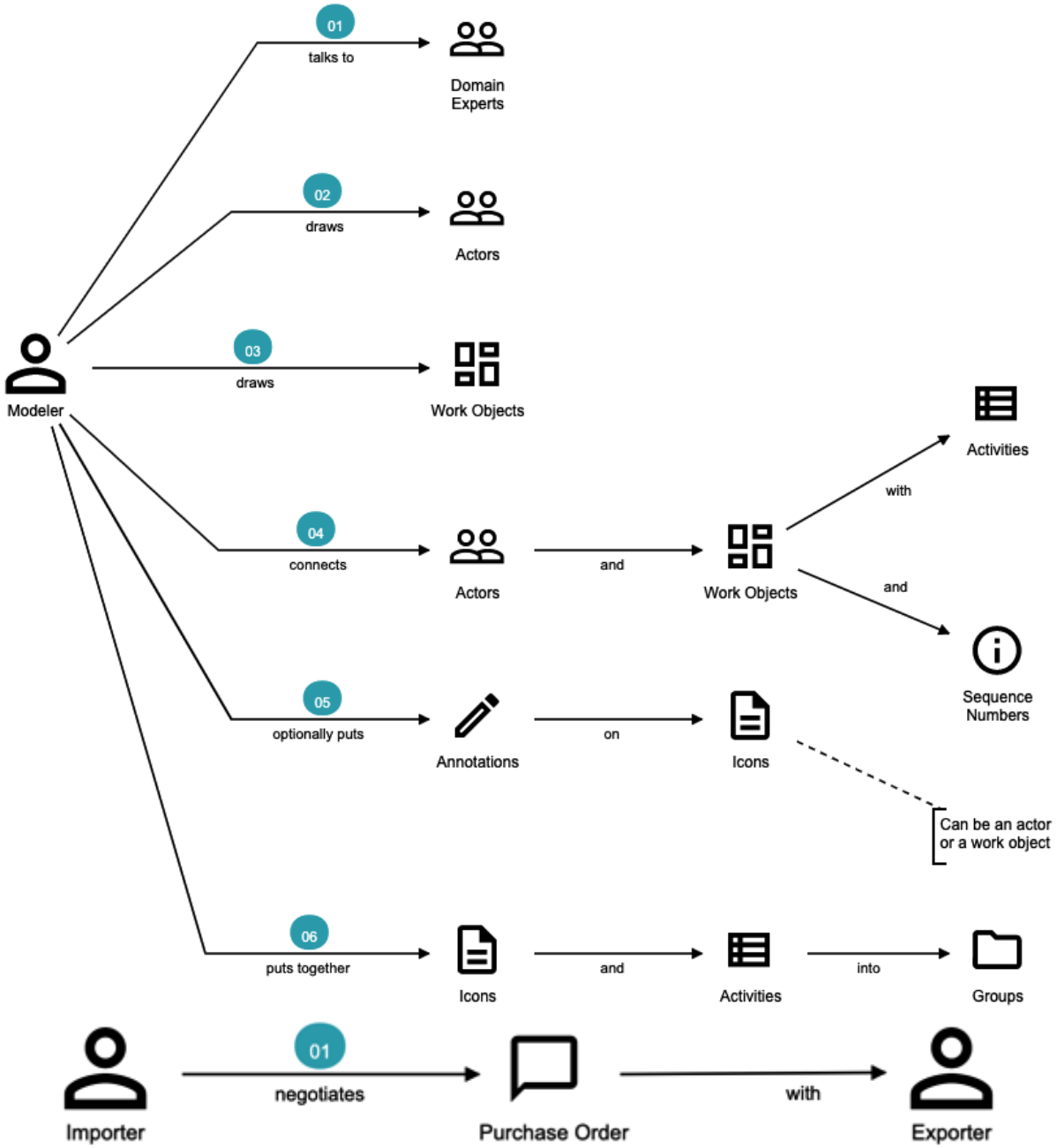


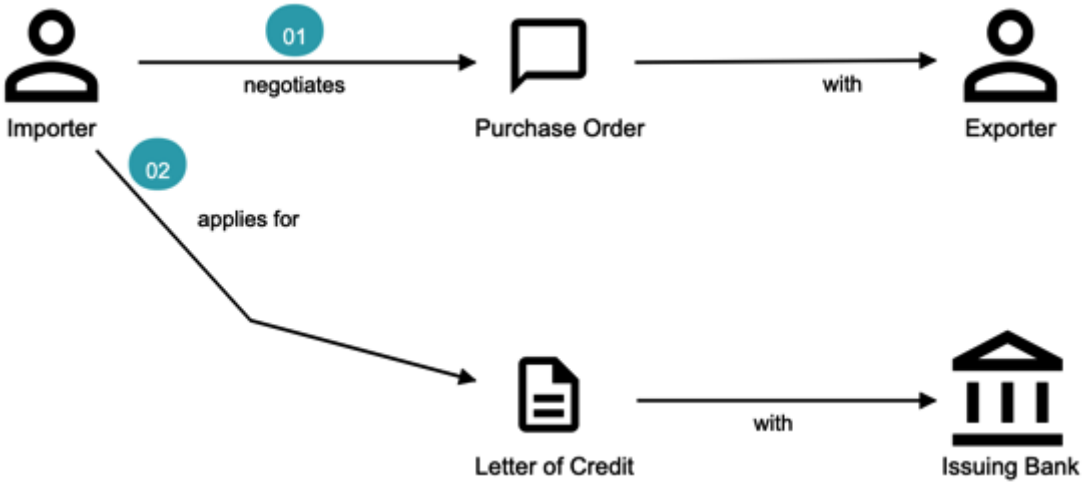
6. Map
Copy the value chain over. Use the evolutionary characteristics to decide where to place each component along the horizontal axis (Evolution).

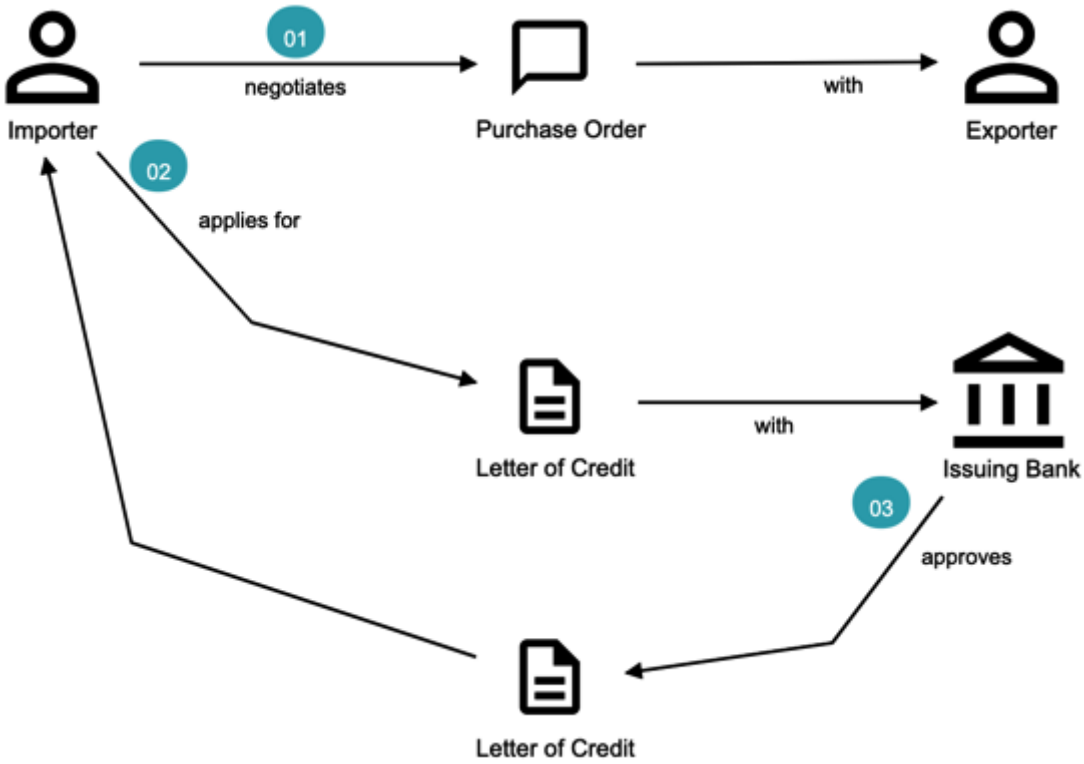


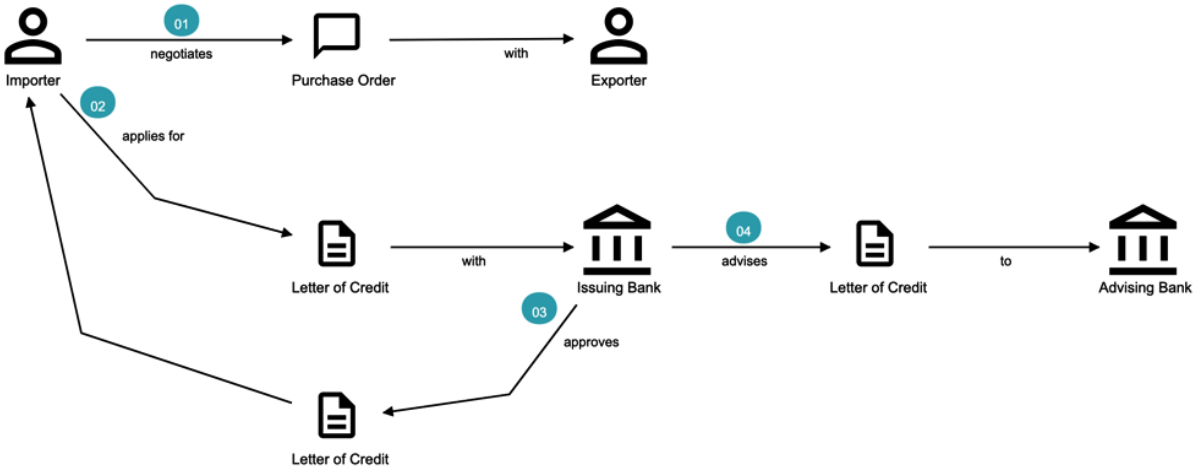

Consignment
Open Account
Documentary Collections
Letter of Credit
Cash In Advance
Exporter
Highest Risk
Lowest Risk
Importer
Cash In Advance
Letter of Credit
Documentary Collections
Open Account
Consignment

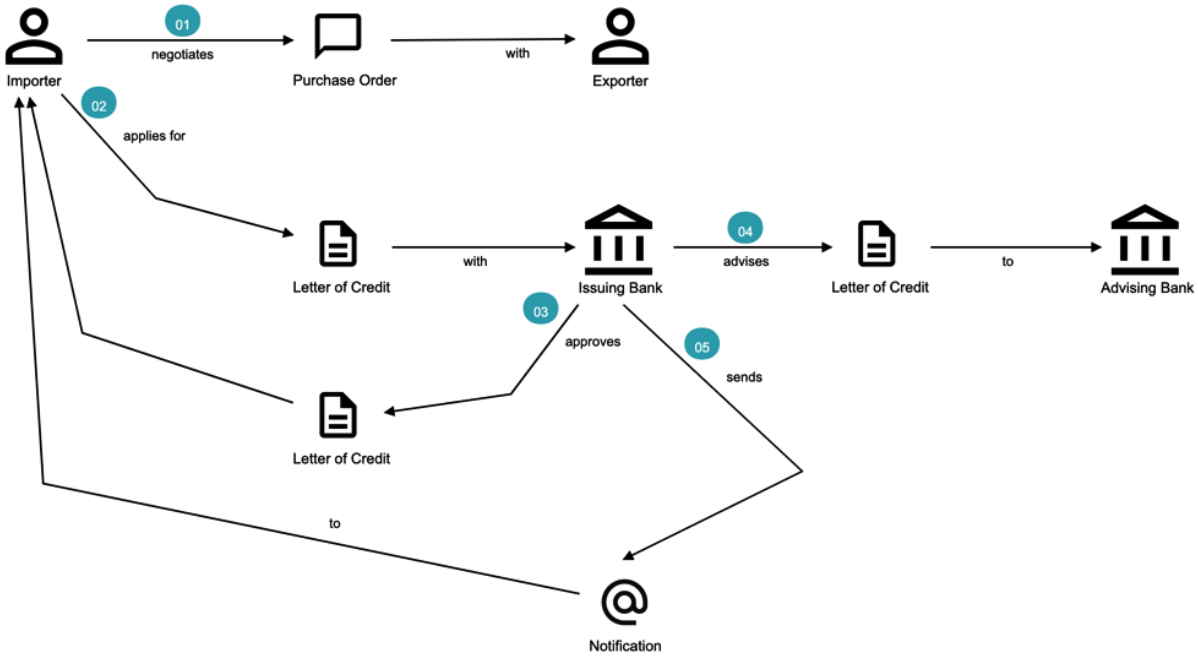
Chapter 4: Domain Analysis and Modeling Using EventStorming

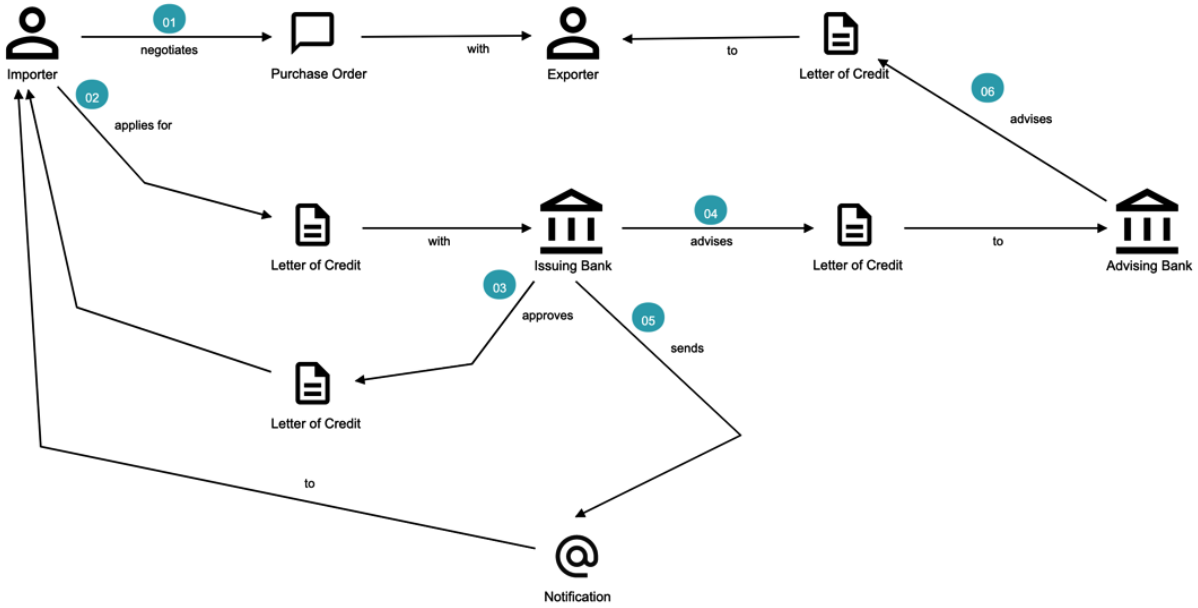


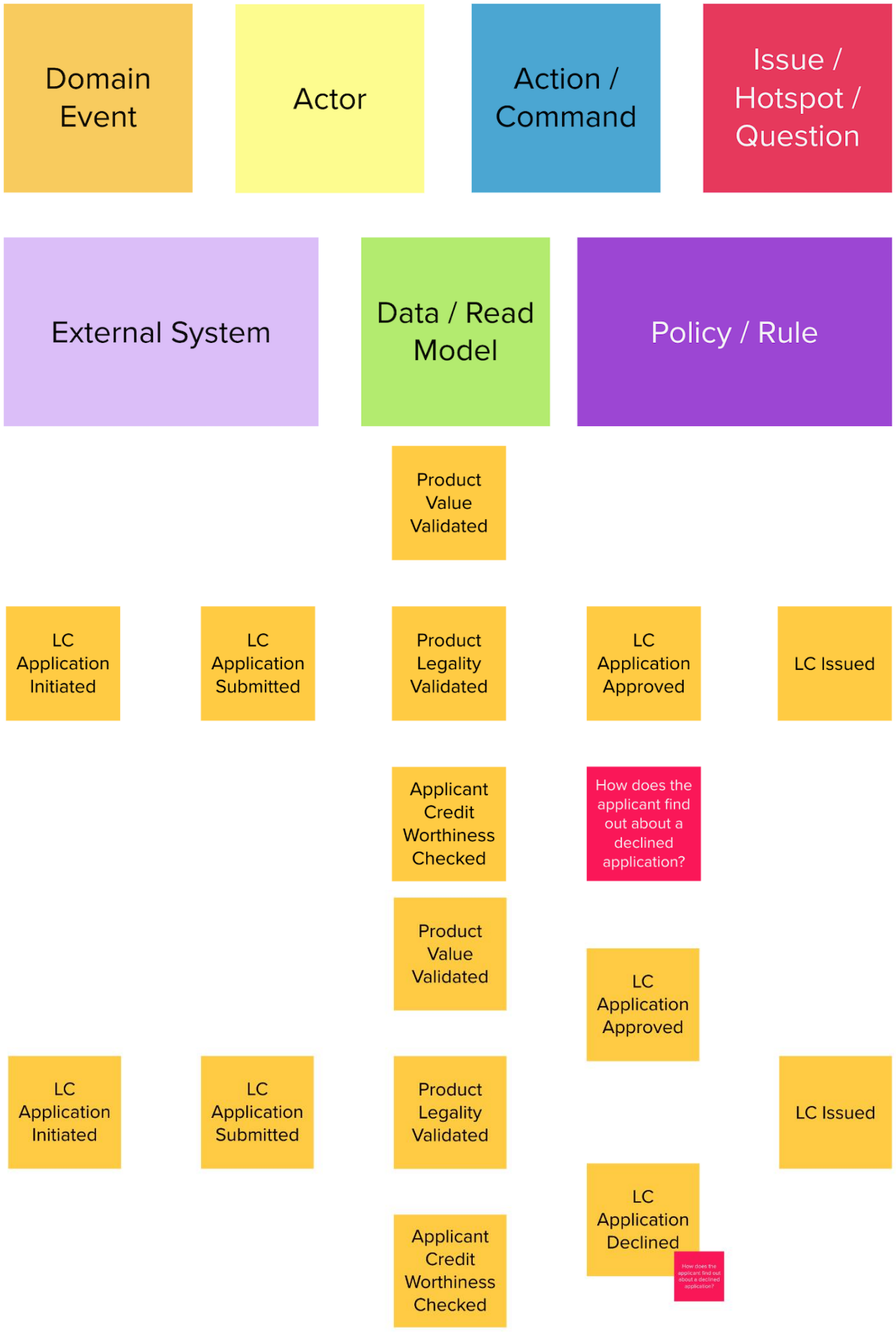

01
negotiates
Importer
Purchase Order
with
Exporter
02
applies for
Letter of Credit
with
Issuing Bank

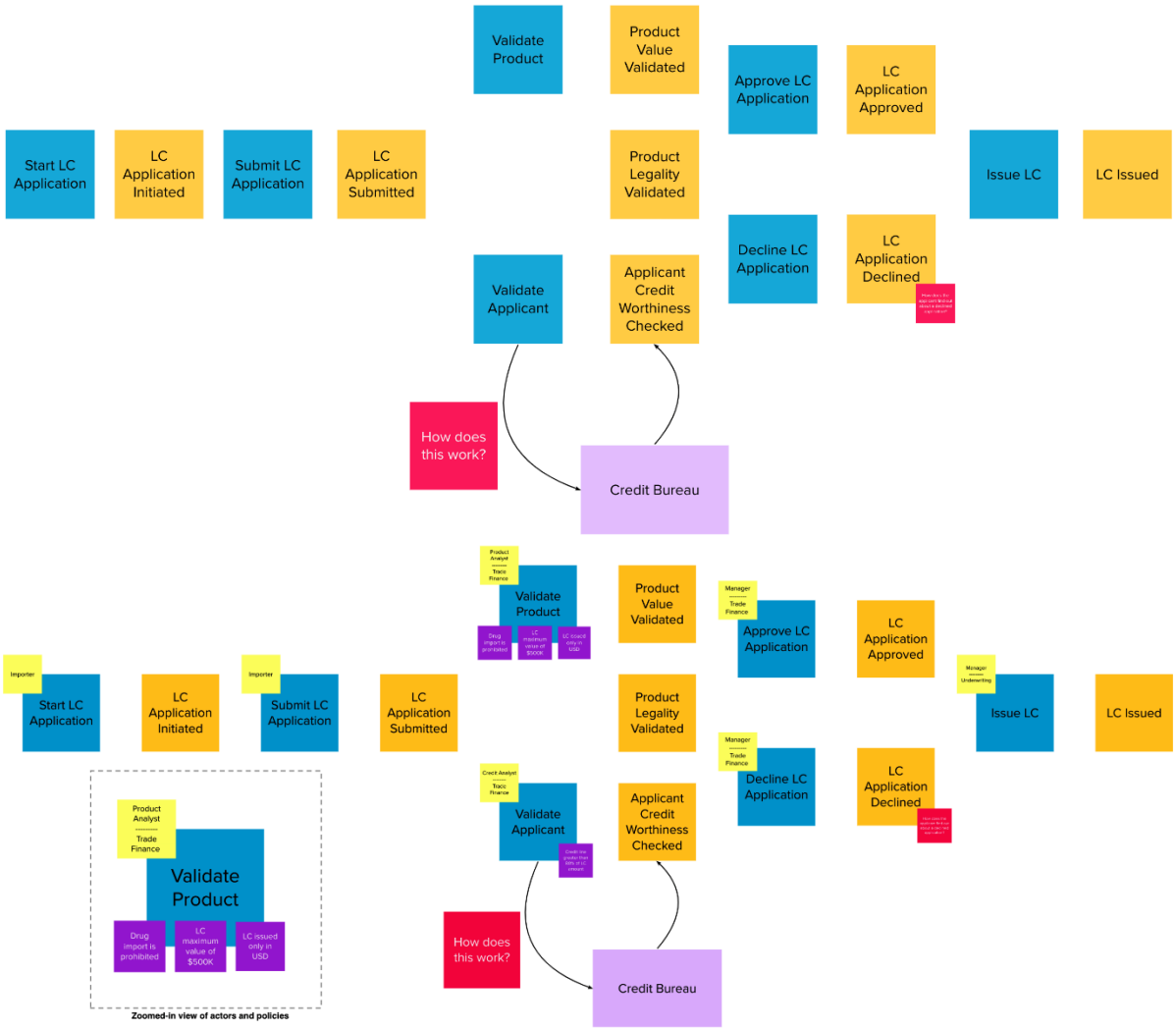

01
negotiates
Importer
Purchase Order
with
Exporter
02
applies for
Letter of Credit
with
Issuing Bank
03
approves
Letter of Credit


01
negotiates
Importer
Purchase Order
with
Exporter
02
applies for
Letter of Credit
with
Issuing Bank
04
advises
Letter of Credit
to
Advising Bank
03
approves
Letter of Credit

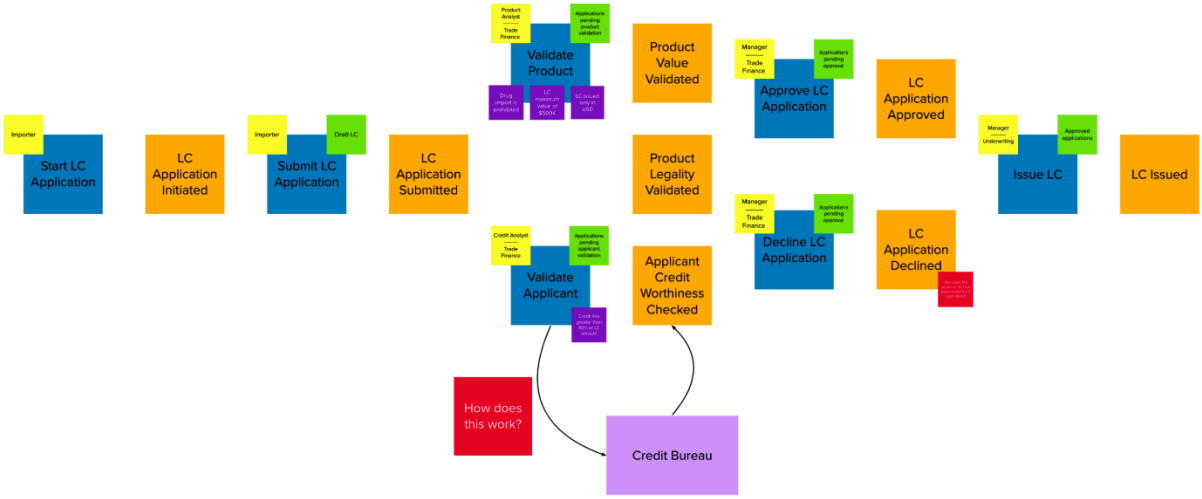

Importer
01
negotiates
Purchase Order
with
Exporter
02
applies for
Letter of Credit
with
Issuing Bank
04
advises
Letter of Credit
to
Advising Bank
03
approves
05
sends
Letter of Credit
to
Notification

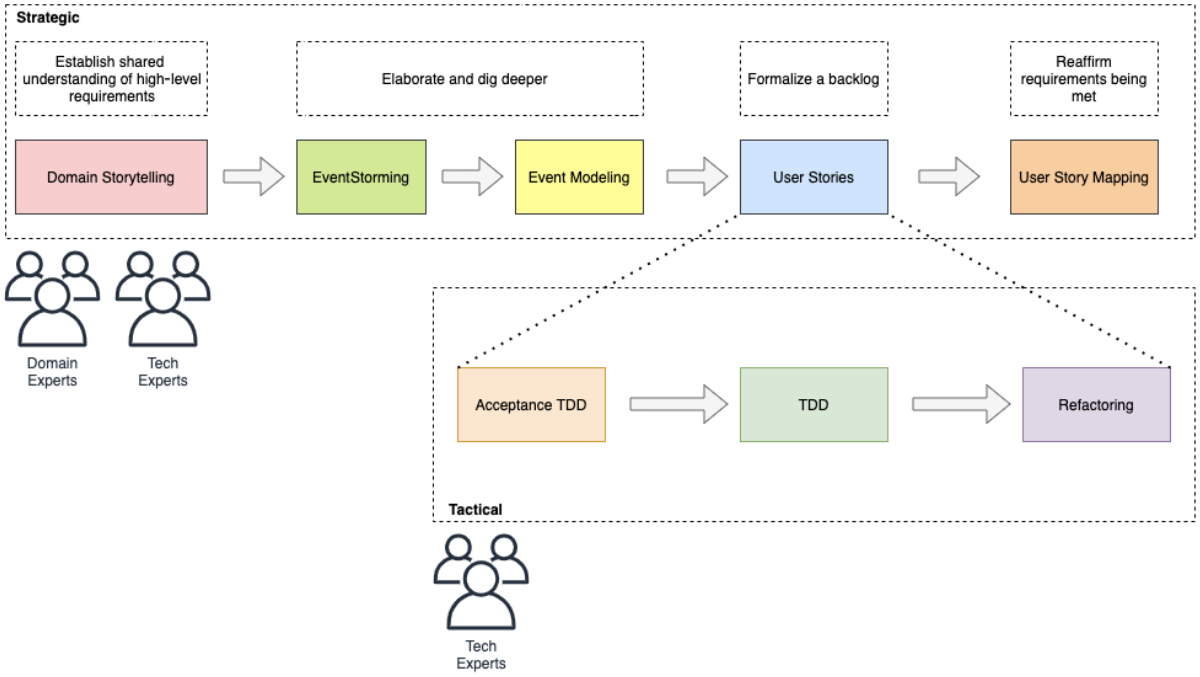

Importer
01
negotiates
Purchase Order
with
Exporter
to
Letter of Credit
06
advises
02
applies for
Letter of Credit
with
Issuing Bank
04
advises
Letter of Credit
to
Advising Bank
03
approves
05
sends
Letter of Credit
to
Notification


Domain Event
Actor
Action / Command
Issue / Hotspot / Question
External System
Data / Read Model
Policy / Rule
Product Value Validated
LC Application Initiated
LC Application Submitted
Product Legality Validated
LC Application Approved
LC Issued
Applicant Credit Worthiness Checked
How does the applicant find out about a declined application?
Product Value Validated
LC Application Approved
LC Application Initiated
LC Application Submitted
Product Legality Validated
LC Issued
LC Application Declined
Applicant Credit Worthiness Checked
How does the applicant find out about a declined application?

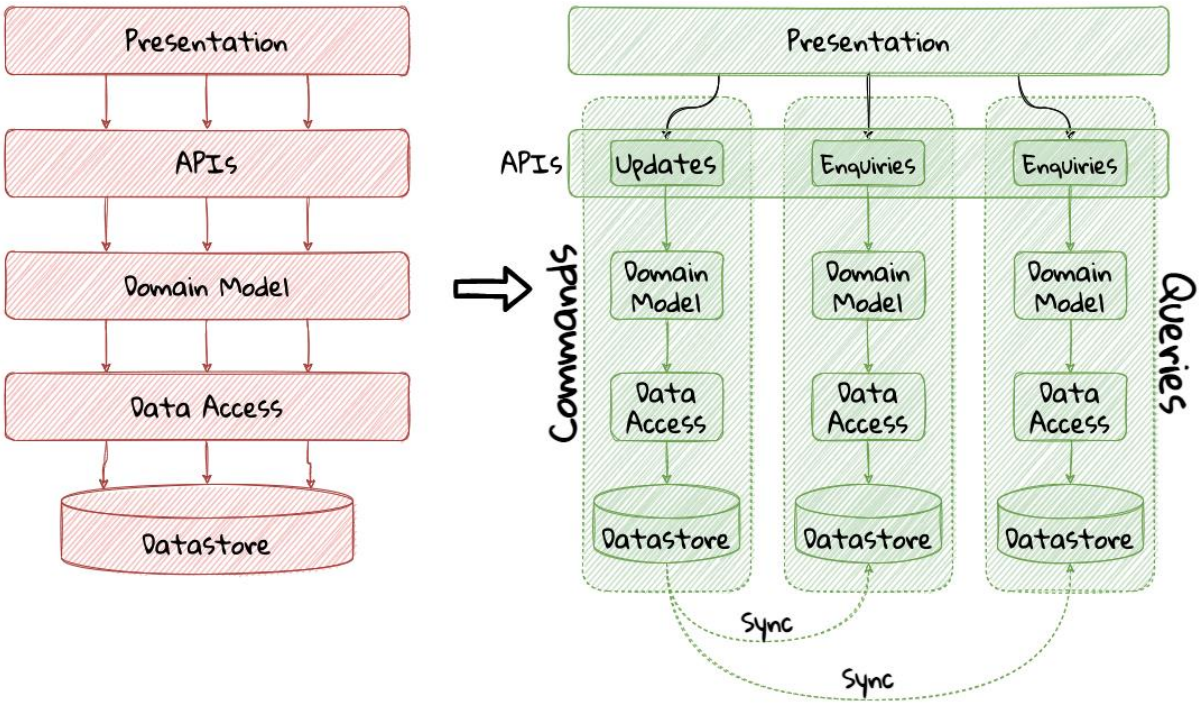


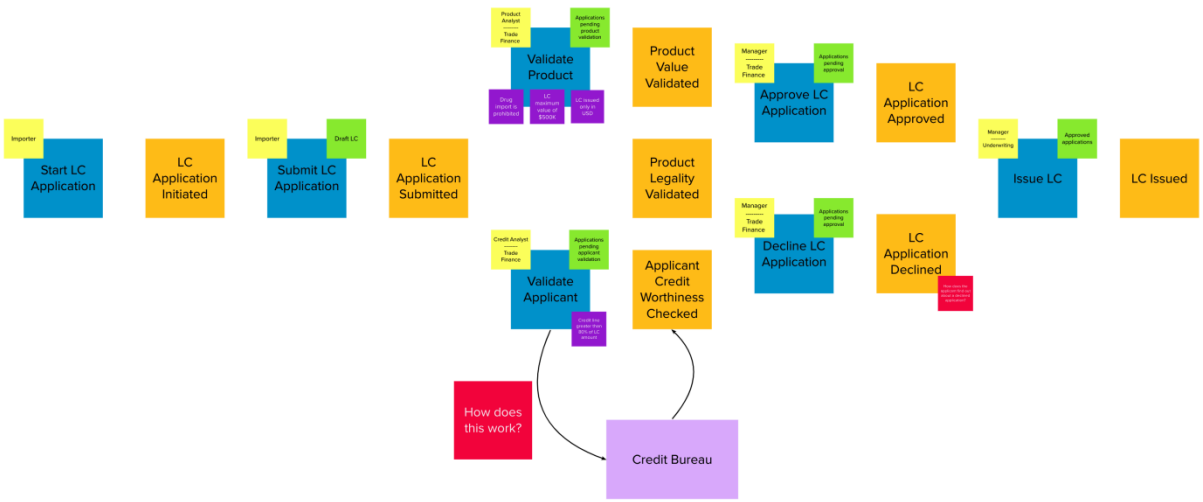
Zoomed-in view of actors and policies

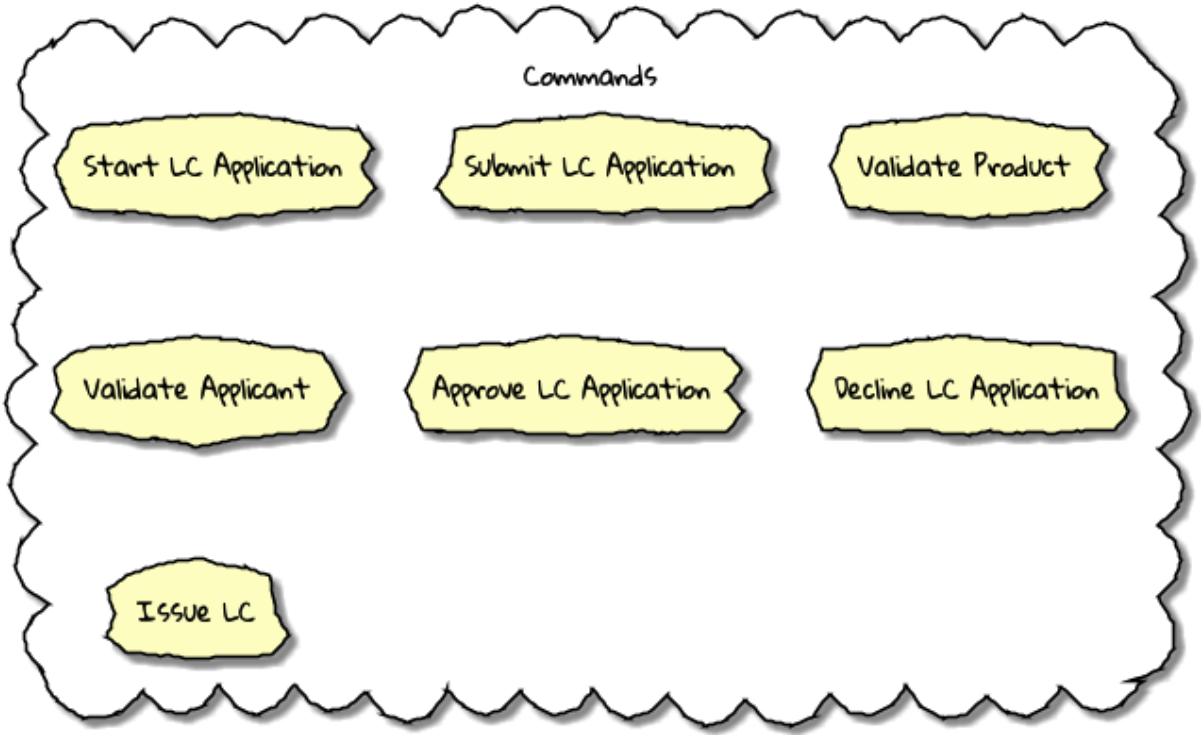

Start LC Application
LC Application Initiated
Submit LC Application
LC Application Submitted
Validate Product
Product Value Validated
Product Legality Validated
Validate Applicant
Applicant Credit Worthiness Checked
How does this work?
Credit Bureau
Approve LC Application
Decline LC Application
LC Application Approved
LC Application Declined
Issue LC
LC Issued

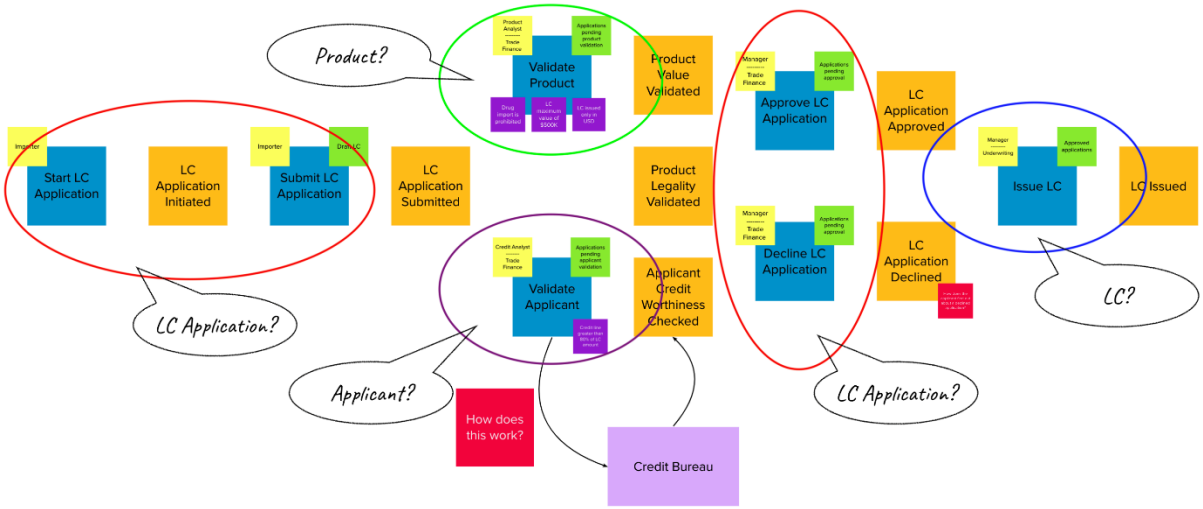

Strategic
Establish shared understanding of high-level requirements
Elaborate and dig deeper
Formalize a backlog
Reaffirm requirements being met
Domain Storytelling
EventStorming
Event Modeling
User Stories
User Story Mapping
Domain Experts
Tech Experts
Acceptance TDD
TDD
Refactoring
Tactical
Tech Experts

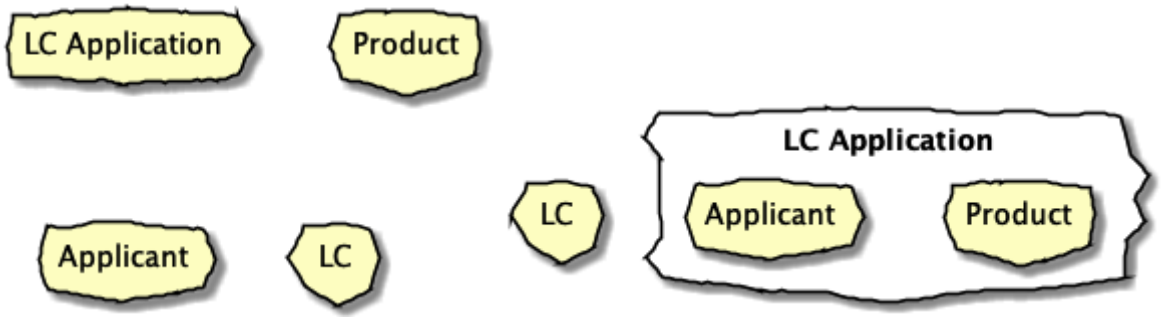
Chapter 5: Implementing Domain Logic

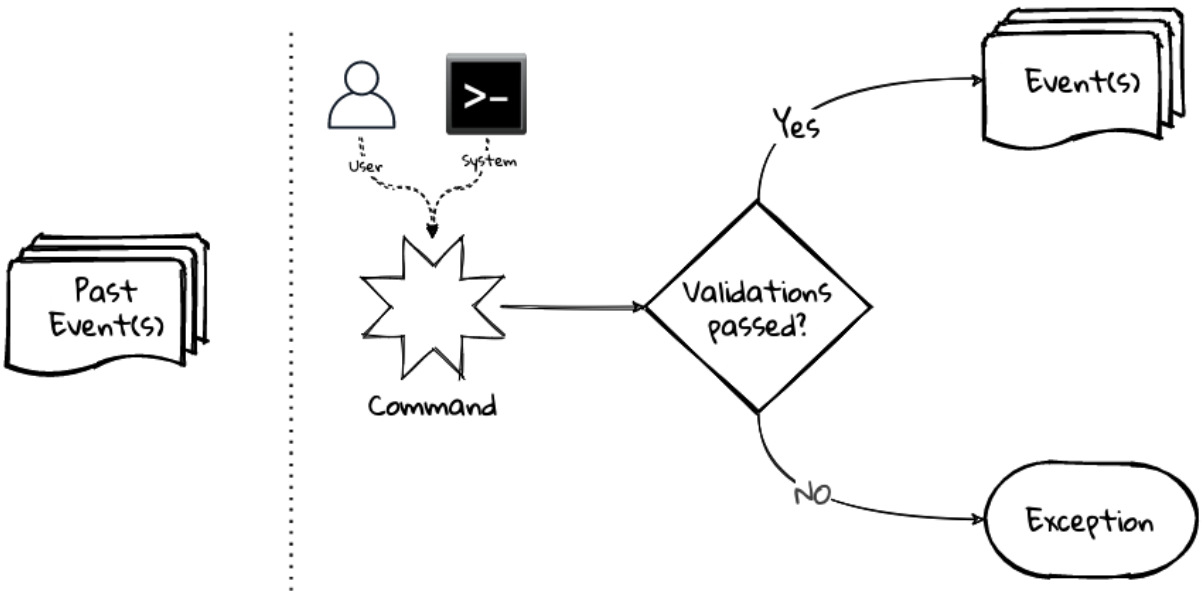


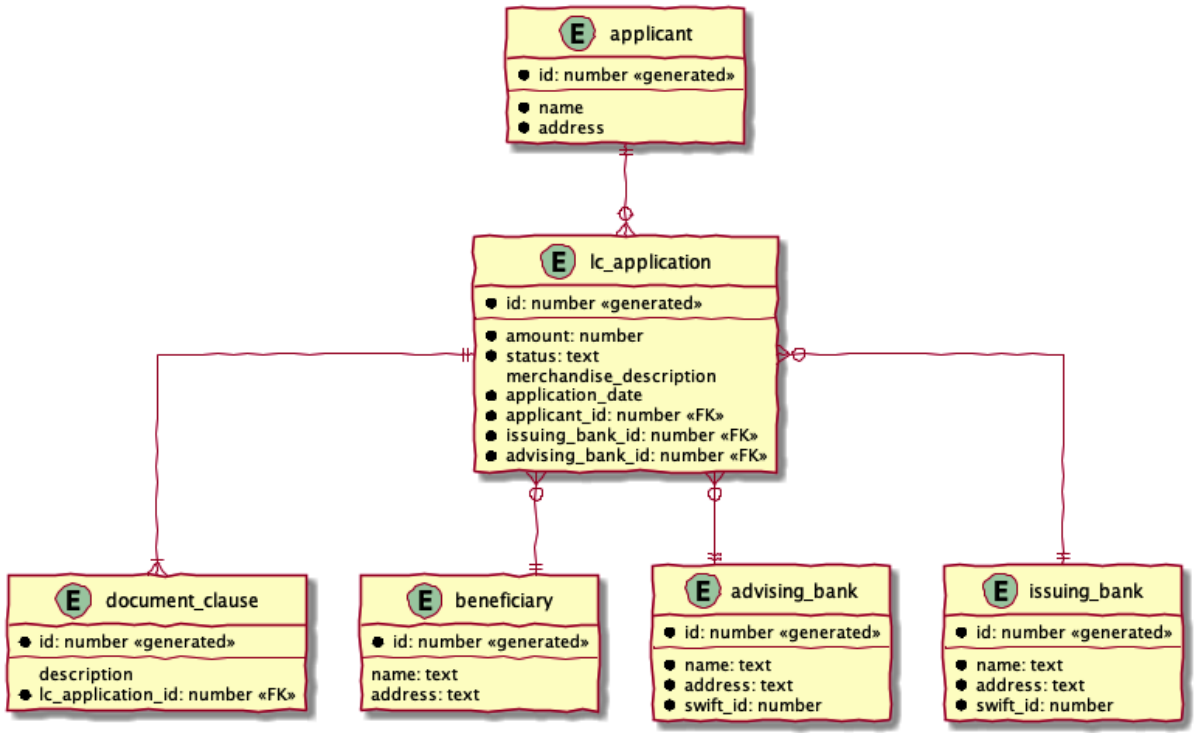


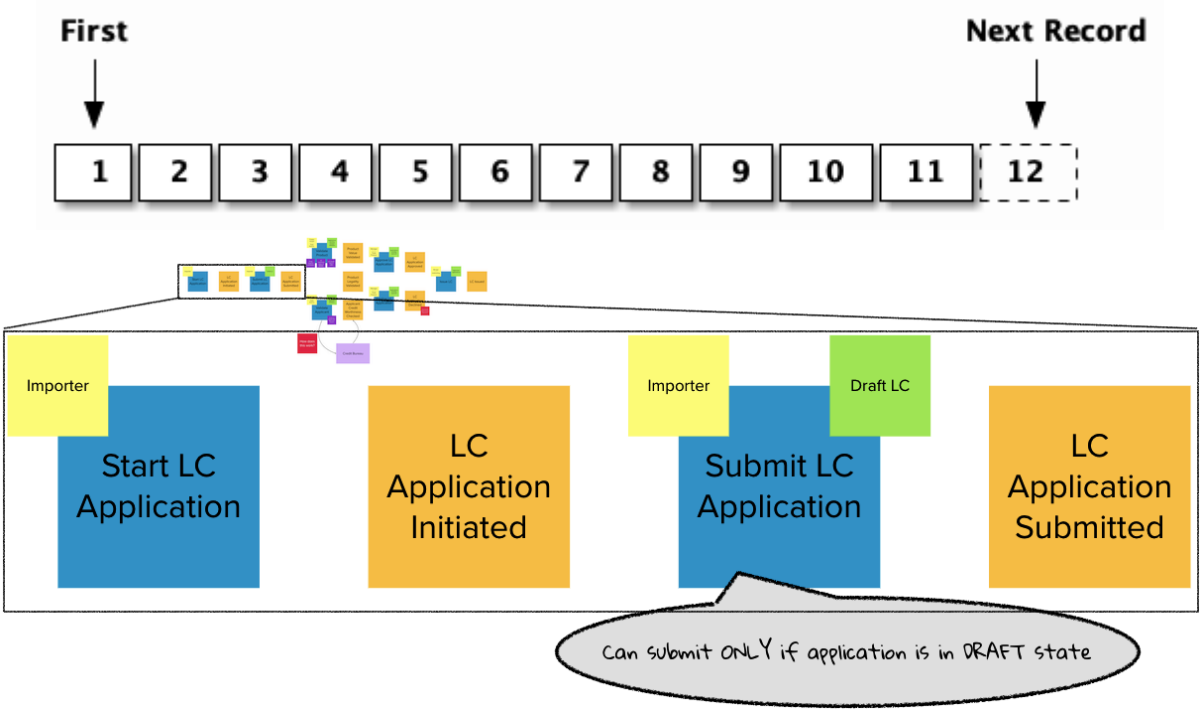

Commands
Start LC Application
Submit LC Application
Validate Product
Validate Applicant
Approve LC Application
Decline LC Application
Issue LC


Product?
Validate Product
Product Value Validated
Start LC Application
LC Application Initiated
Submit LC Application
LC Application Submitted
Product Legality Validated
Approve LC Application
LC Application Approved
Issue LC
LC Issued
LC?
LC Application?
Validate Applicant
Applicant Credit Worthiness Checked
Decline LC Application
LC Application Declined
Applicant?
How does this work?
Credit Bureau
LC Application?

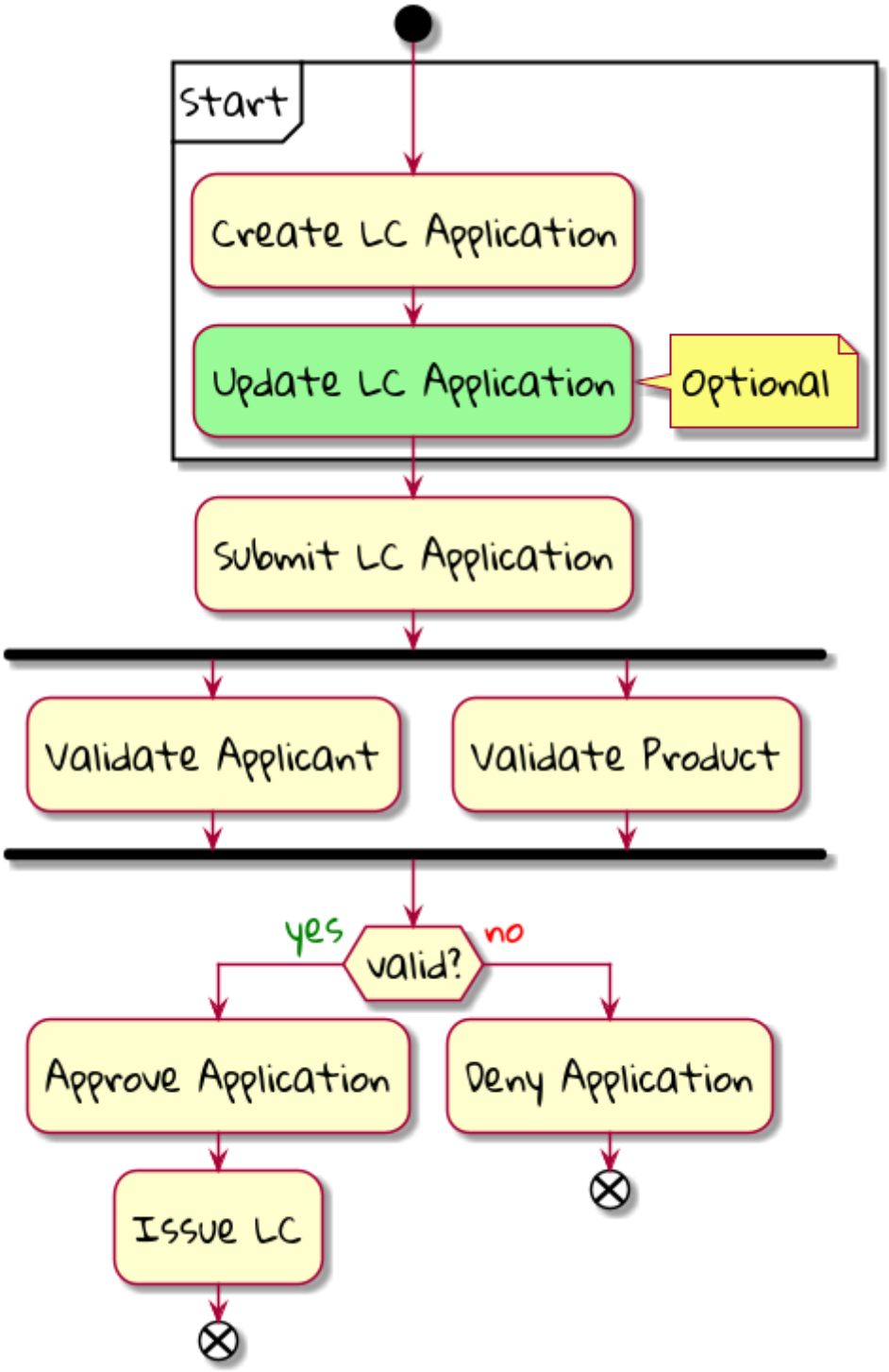

LC Application
Product
Applicant
LC
LC Application
Applicant
Product

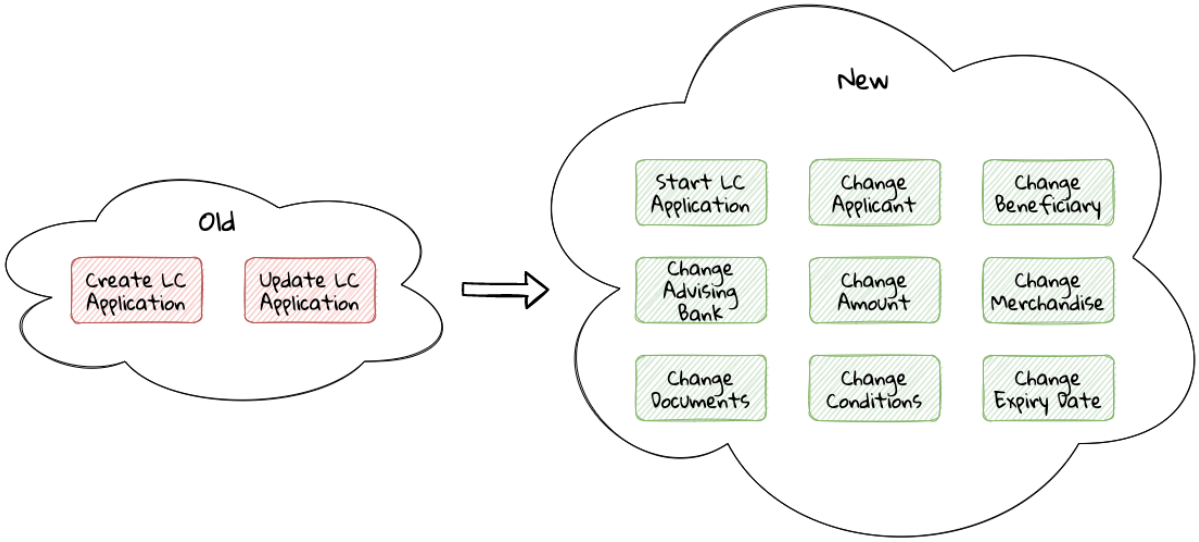

Past
Event(s)
User
System
Command
Validations
passed?
Yes
Event(s)
NO
Exception

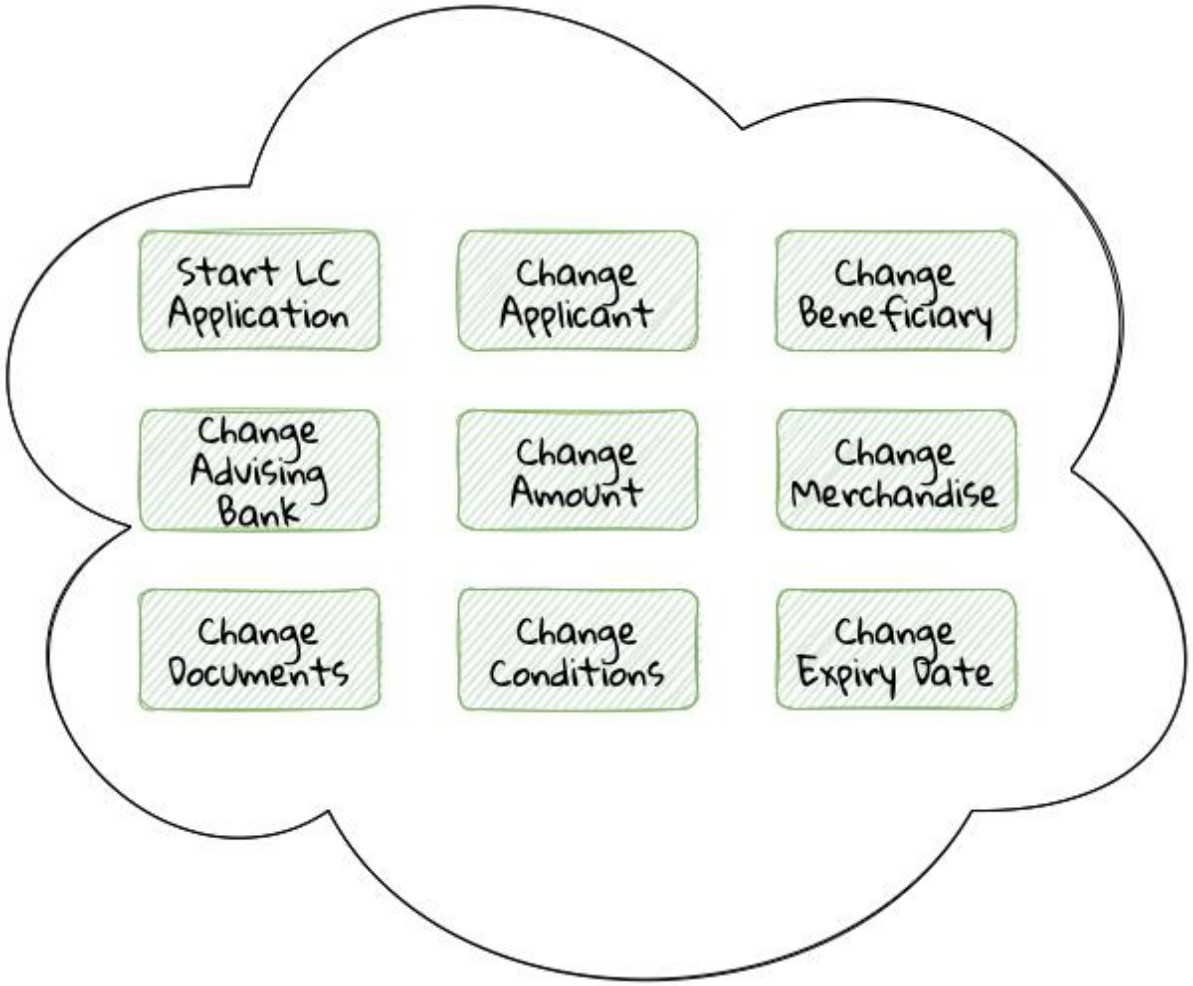

applicant
id: number «generated»
name
address
lc_application
id: number «generated»
amount: number
status: text
merchandise_description
application_date
applicant_id: number «FK»
issuing_bank_id: number «FK»
advising_bank_id: number «FK»
document_clause
id: number «generated»
description
lc_application_id: number «FK»
beneficiary
id: number «generated»
name: text
address: text
advising_bank
id: number «generated»
name: text
address: text
swift_id: number
issuing_bank
id: number «generated»
name: text
address: text
swift_id: number

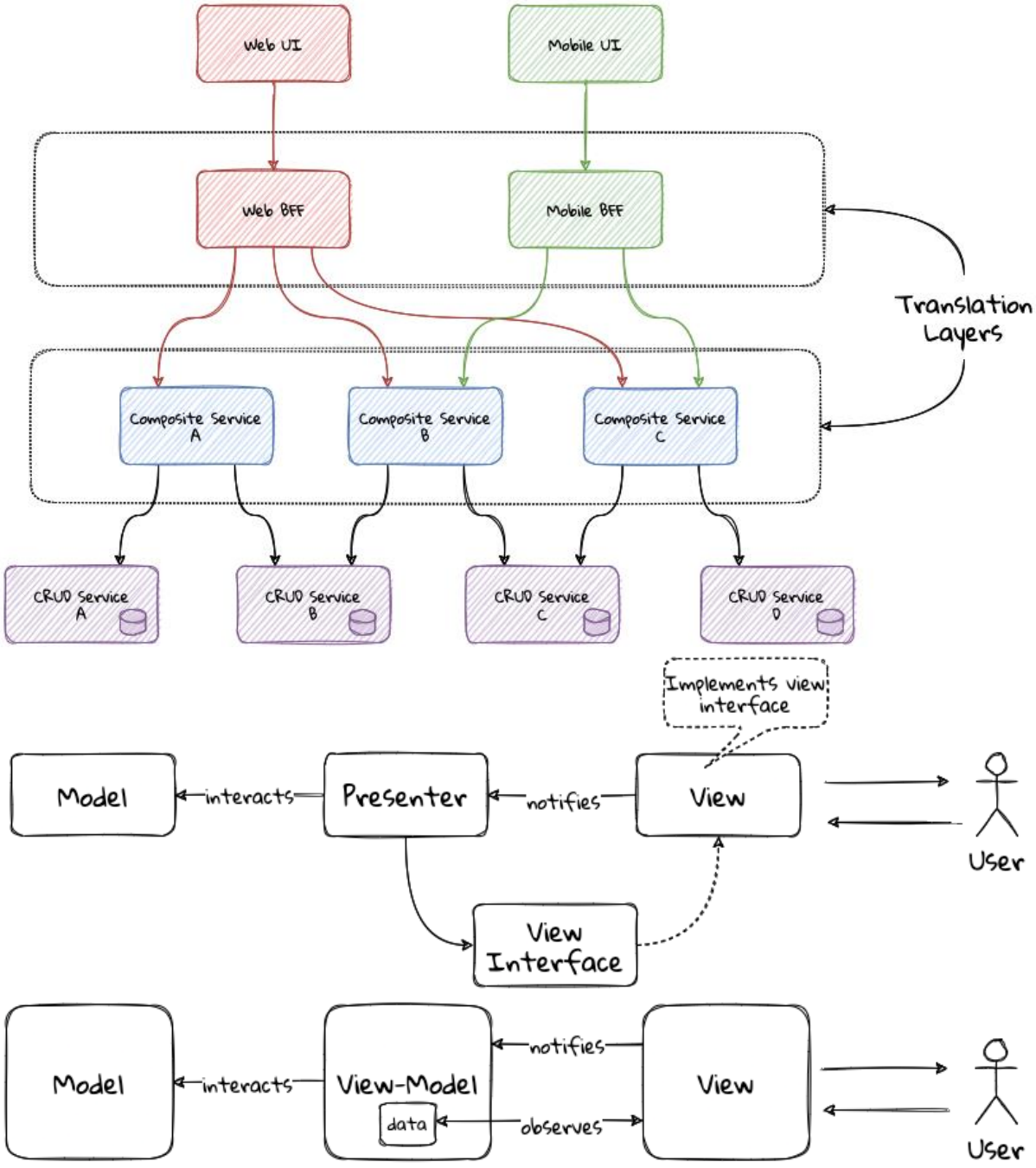

First
Next Record
1
2
3
4
5
6
7
8
9
10
11
12
Importer
Start LC Application
LC Application Initiated
Importer
Draft LC
Submit LC Application
LC Application Submitted
Can submit ONLY if application is in DRAFT state

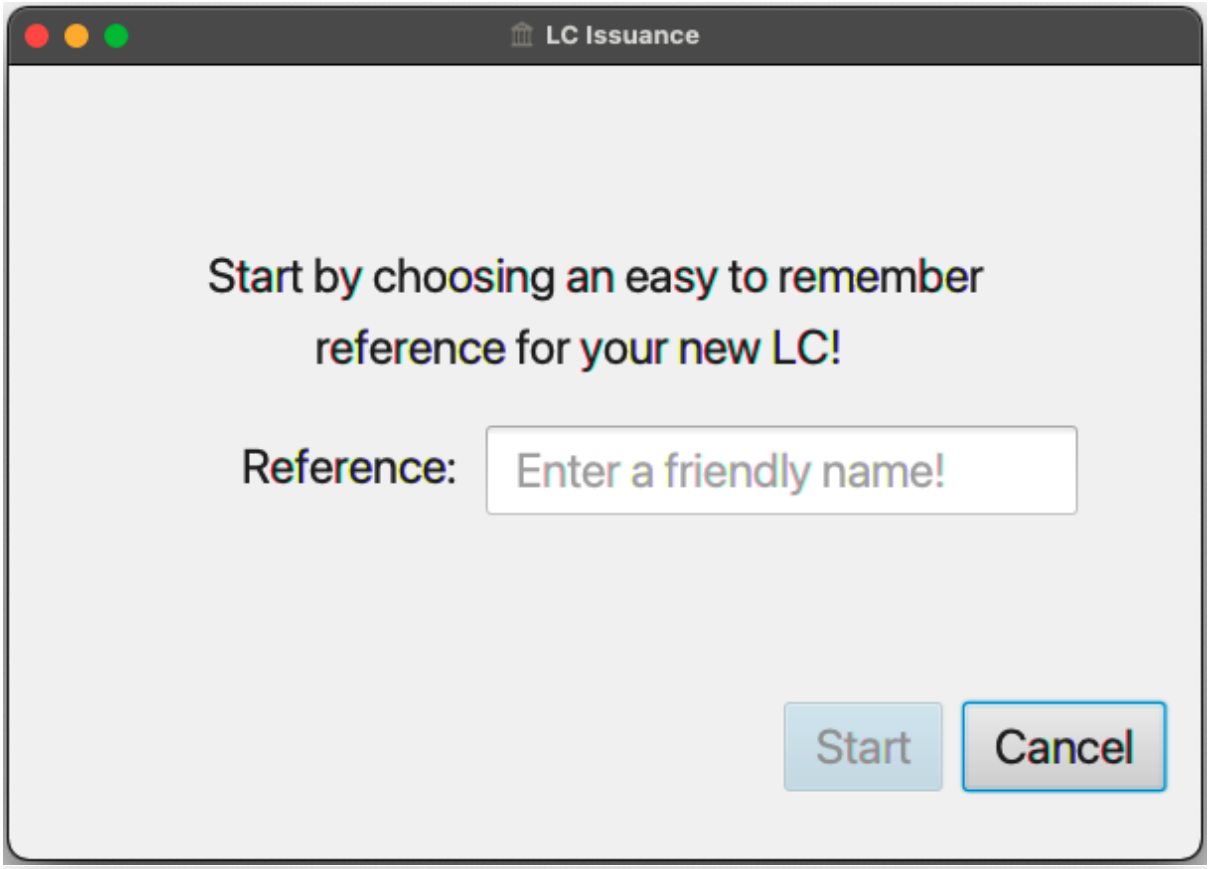
Chapter 6: Implementing the User Interface- Task-Based

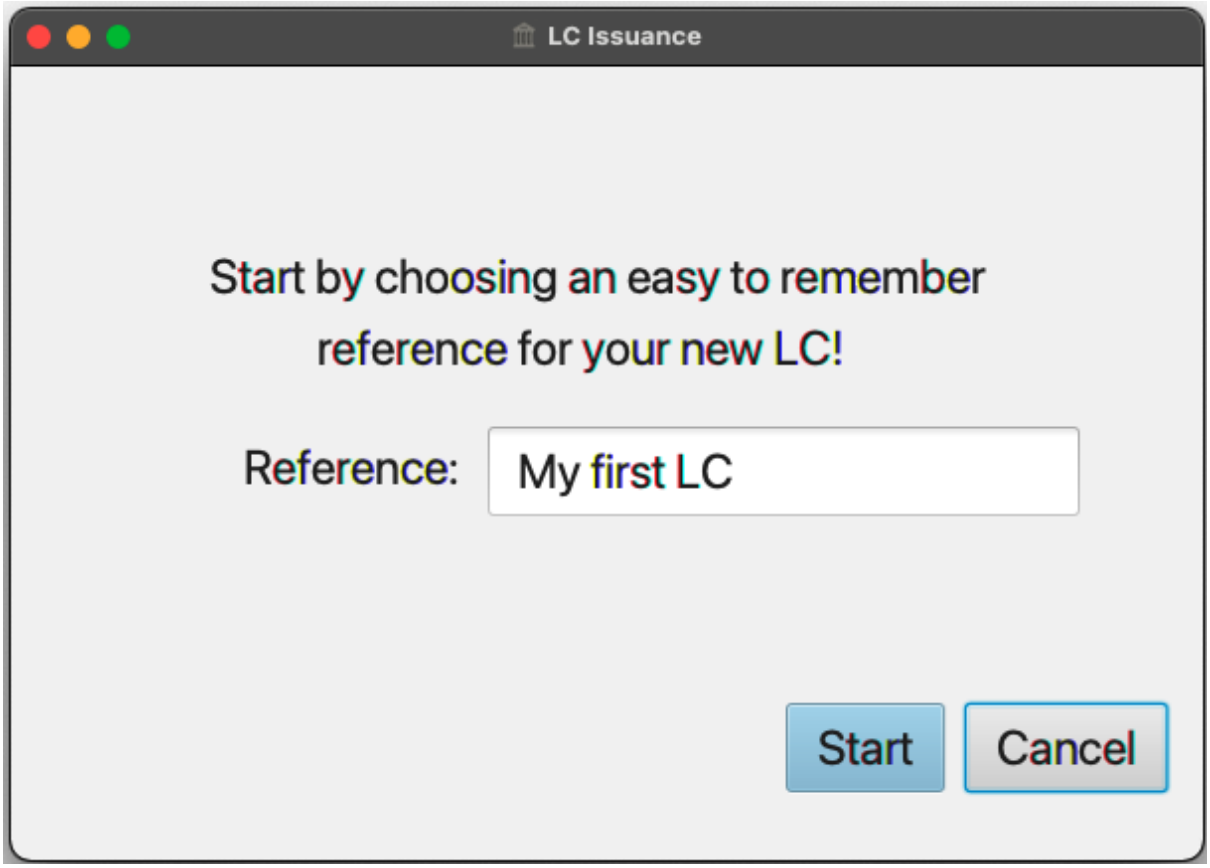



Old
Create LC Application
Update LC Application
New
Start LC Application
Change Applicant
Change Beneficiary
Change Advising Bank
Change Amount
Change Merchandise
Change Documents
Change Conditions
Change Expiry Date

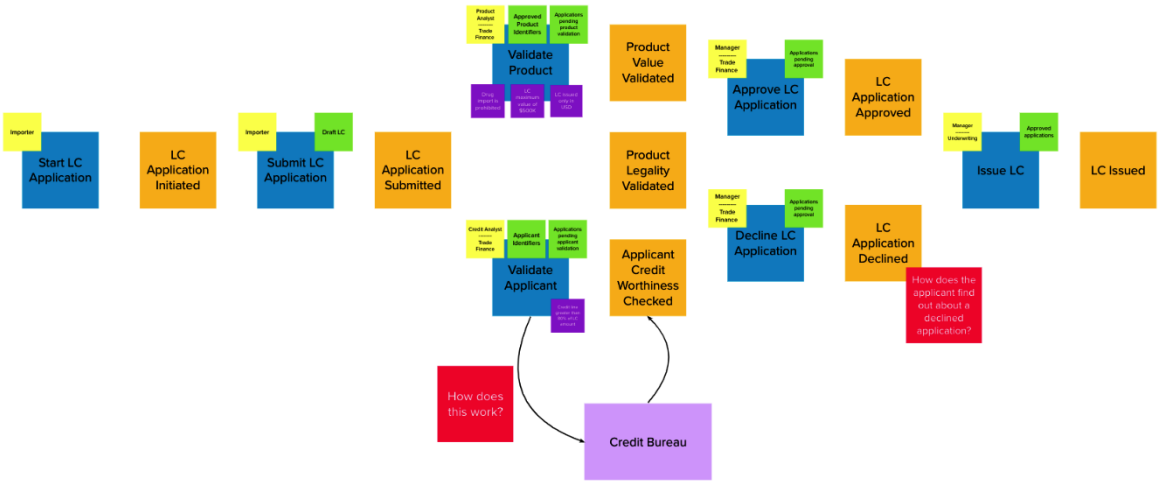

Start LC Application
Change Applicant
Change Beneficiary
Change Advising Bank
Change Amount
Change Merchandise
Change Documents
Change Conditions
Change Expiry Date

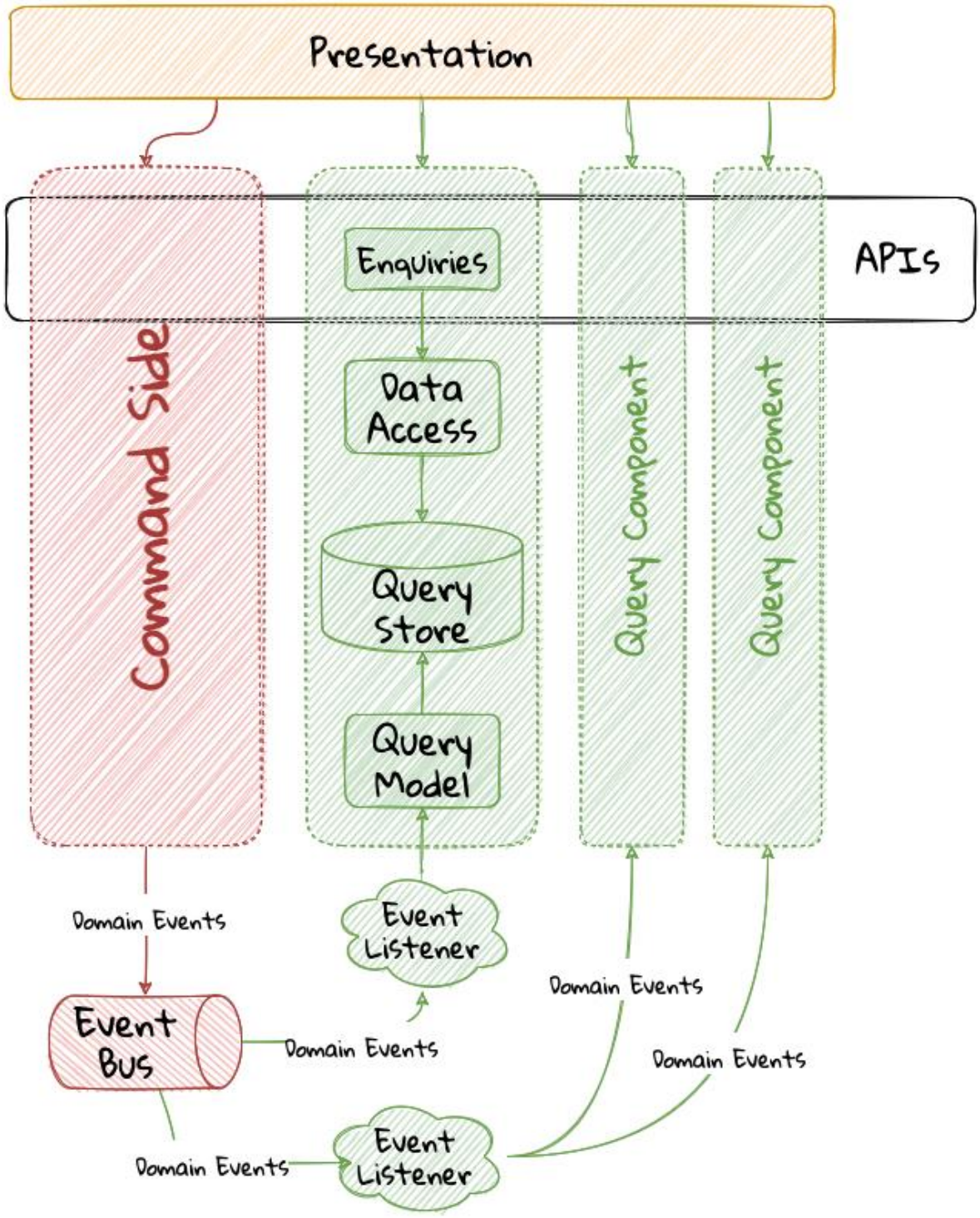

Web UI
Mobile UI
Web BFF
Mobile BFF
Translation Layers
Composite Service A
Composite Service B
Composite Service C
CRUD Service A
CRUD Service B
CRUD Service C
CRUD Service D
Implements view interface
Model
interacts
Presenter
notifies
View
User
View Interface
Model
interacts
View-Model
data
notifies
observes
View
User

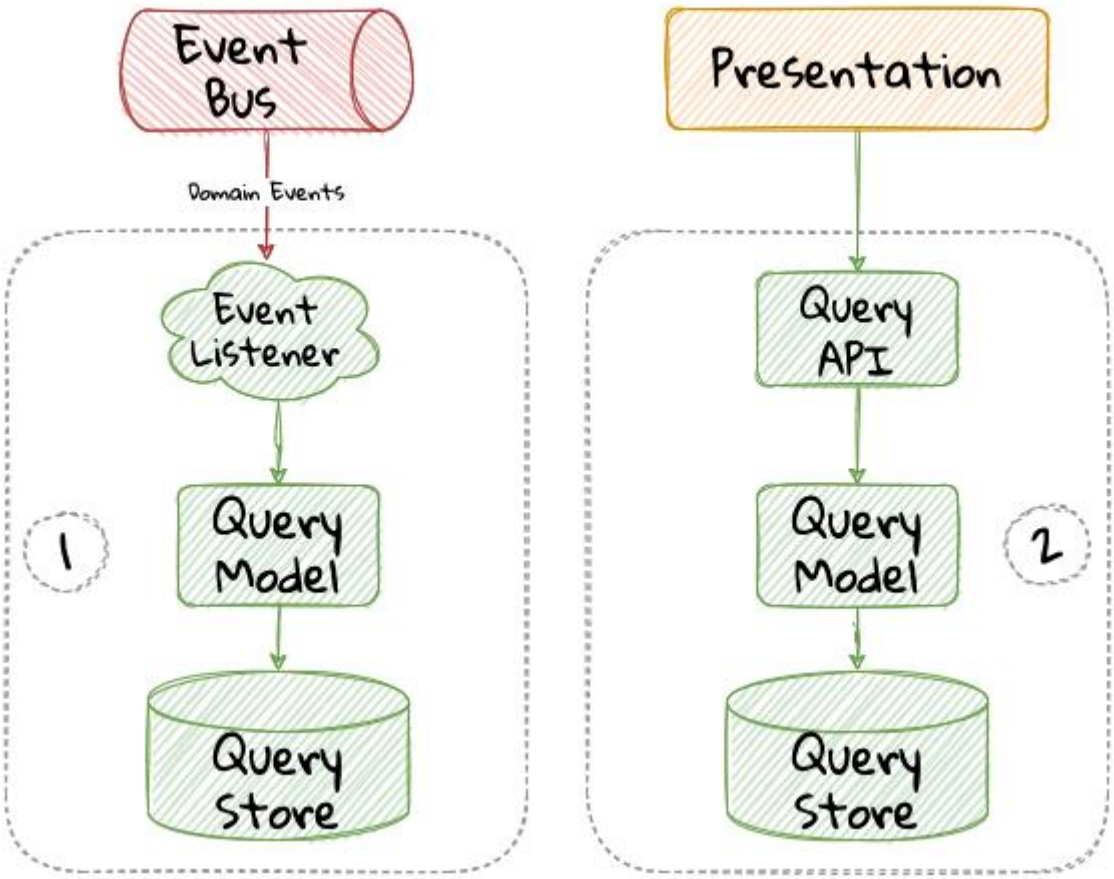

LC Issuance
Start by choosing an easy to remember reference for your new LC!
Reference:
Enter a friendly name!
Start
Cancel

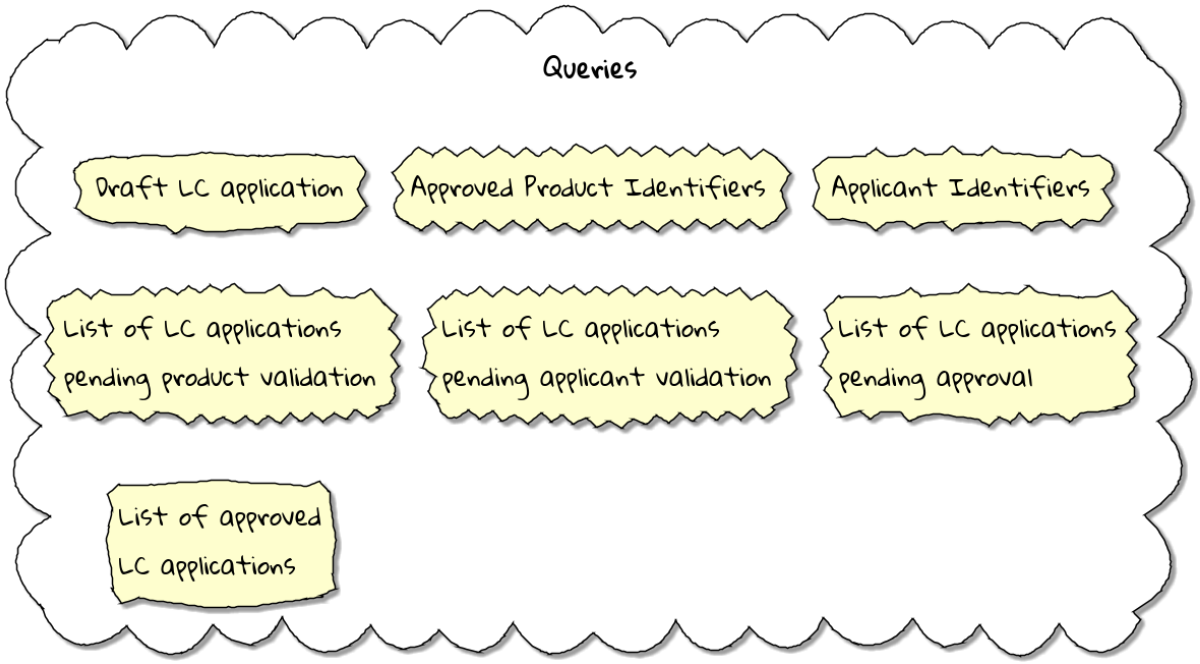

LC Issuance
Start by choosing an easy to remember reference for your new LC!
Reference:
My first LC
Start
Cancel

Chapter 7: Implementing Queries

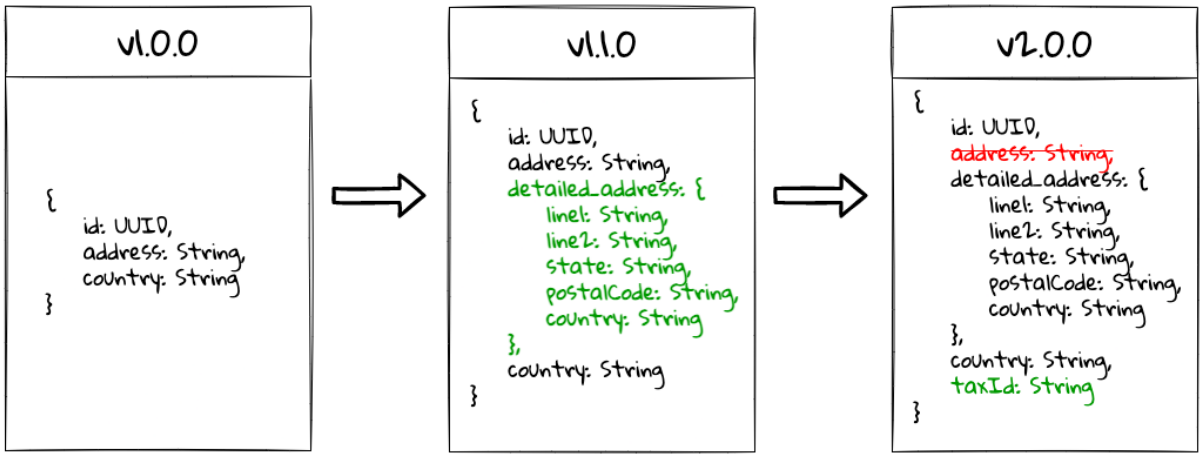



Presentation
APIs
Enquiries
Data Access
Query Store
Query Model
Command Side
Query Component
Query Component
Domain Events
Event Listener
Event Bus
Domain Events
Domain Events
Domain Events
Domain Events
Event Listener

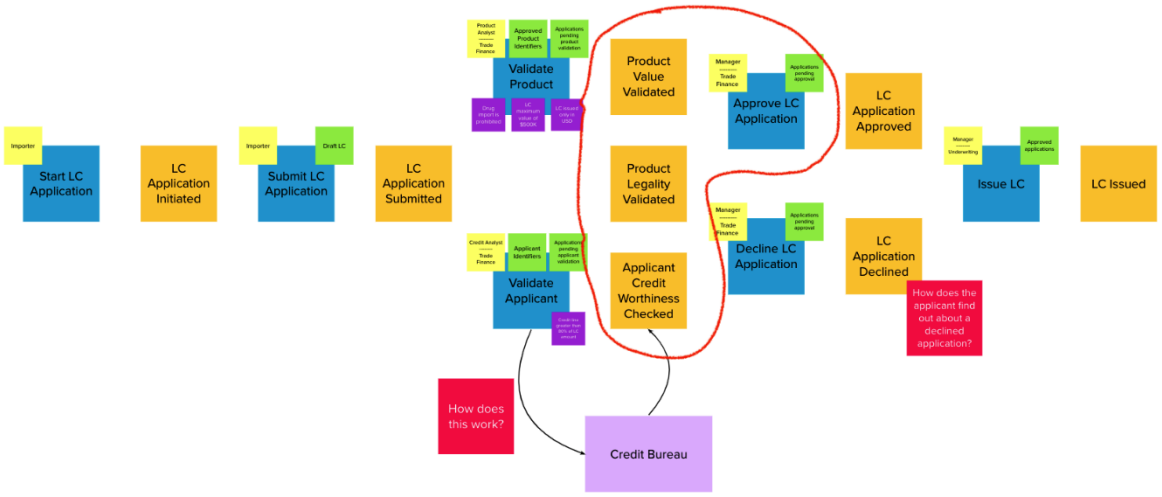

Event Bus
Domain Events
Event Listener
Query Model
Query Store
1
Presentation
Query API
Query Model
Query Store
2

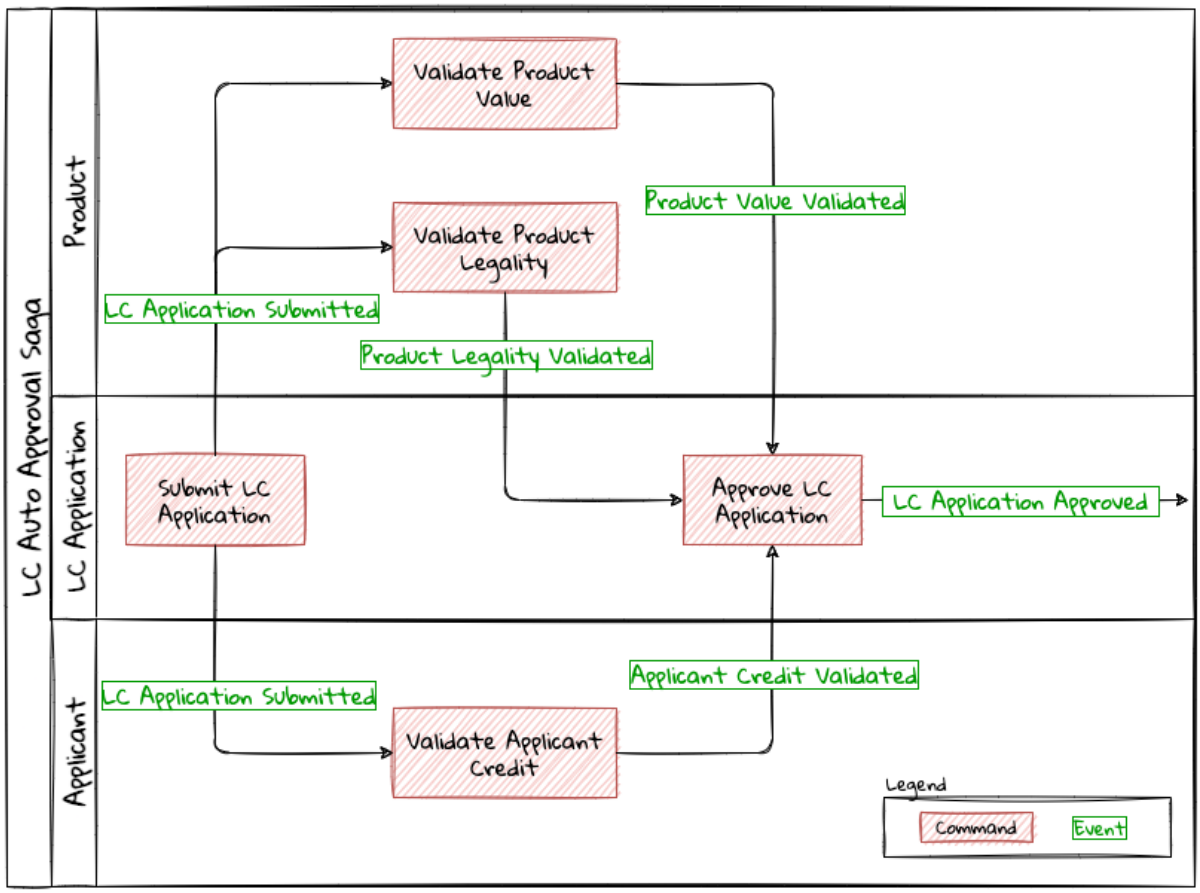

Queries
Draft LC application
Approved Product Identifiers
Applicant Identifiers
List of LC applications pending product validation
List of LC applications pending applicant validation
List of LC applications pending approval
List of approved LC applications

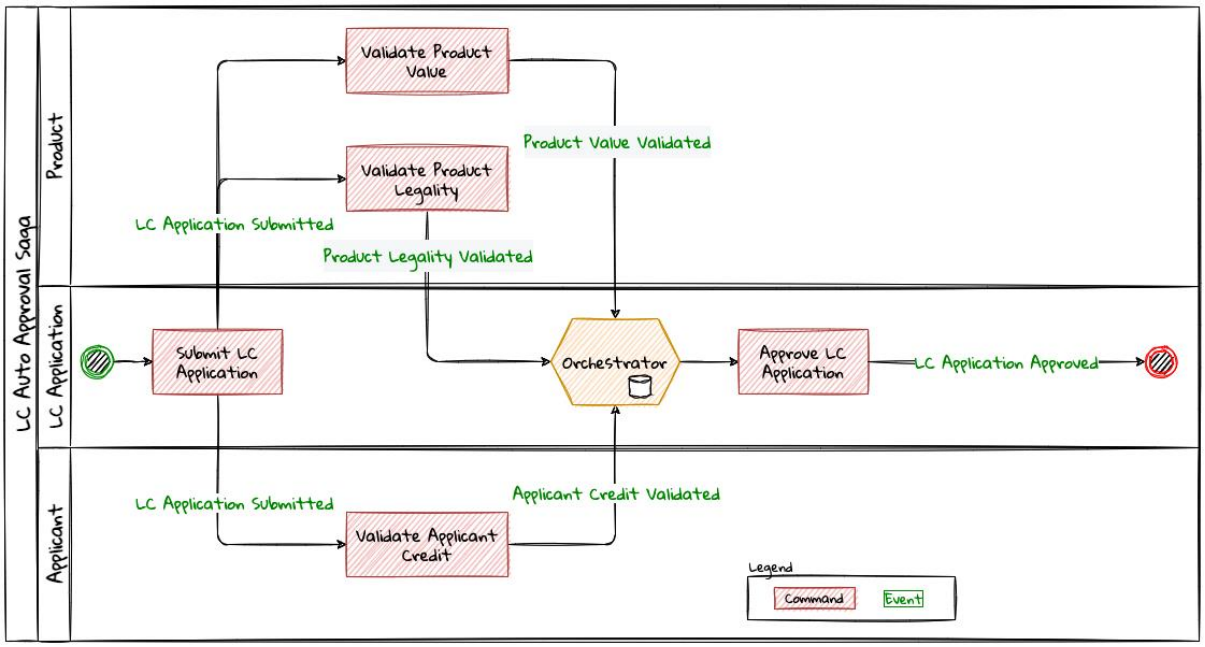
BeneficiaryInformationChangedEvent

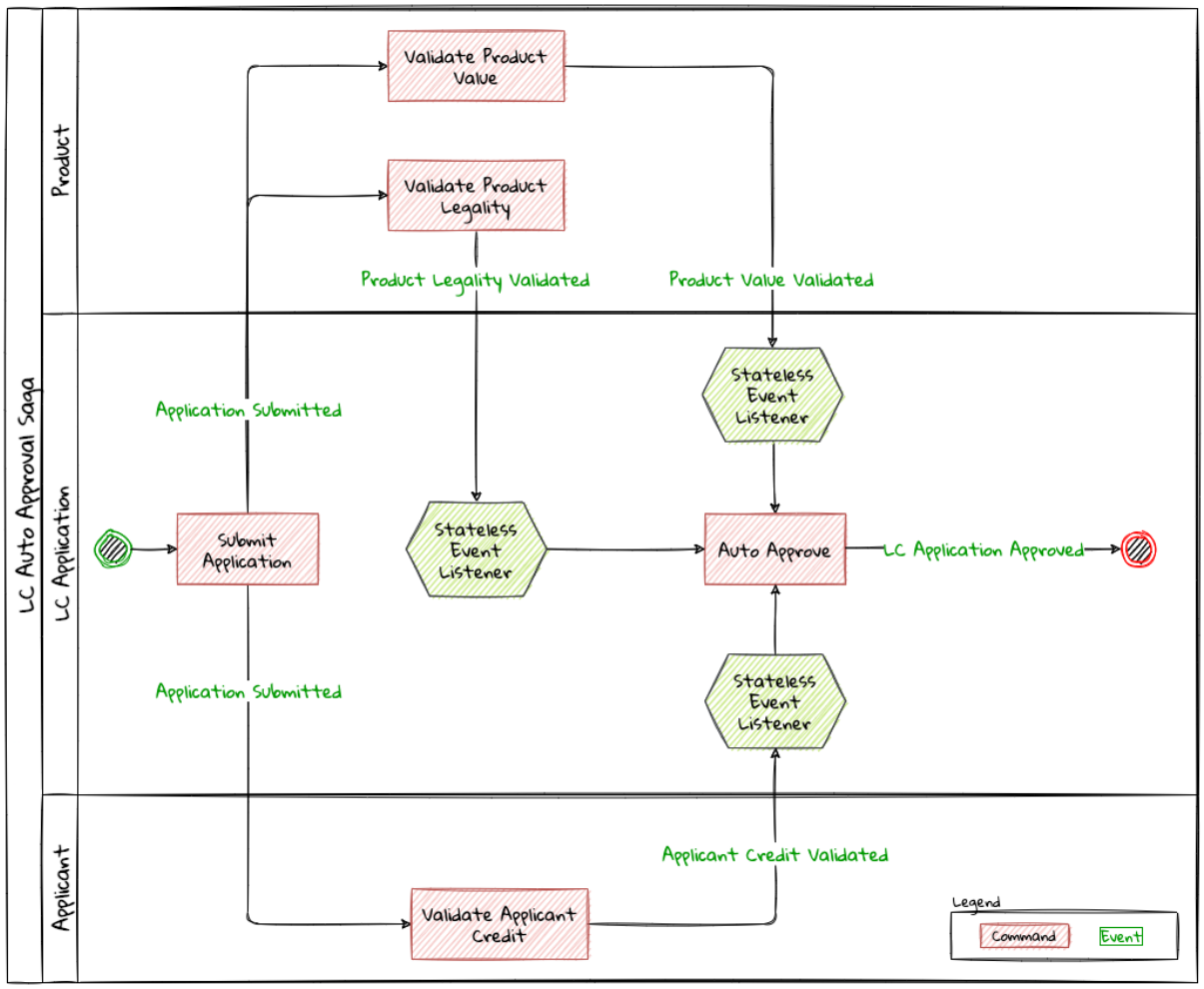


Chapter 8: Implementing Long-Running Flows

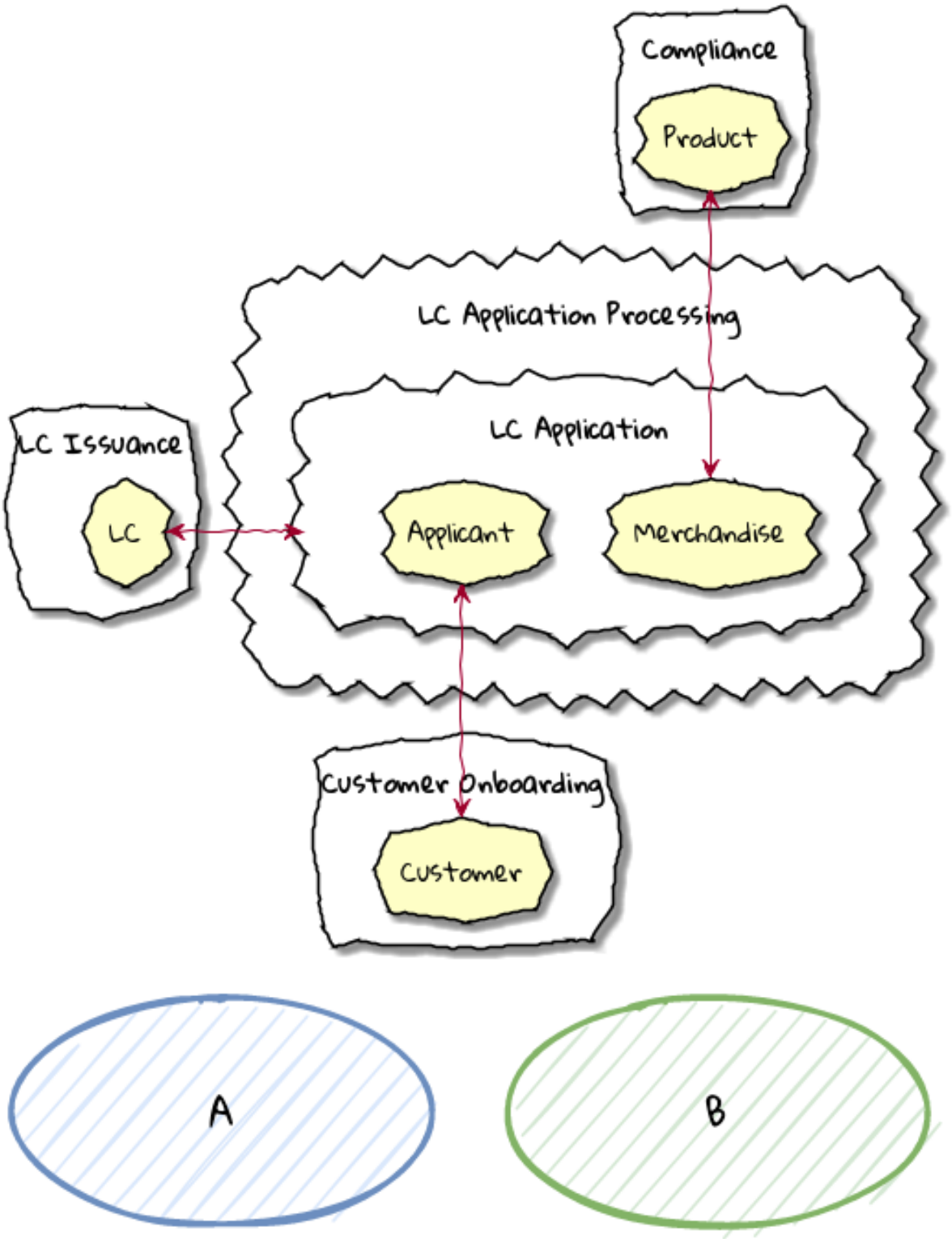


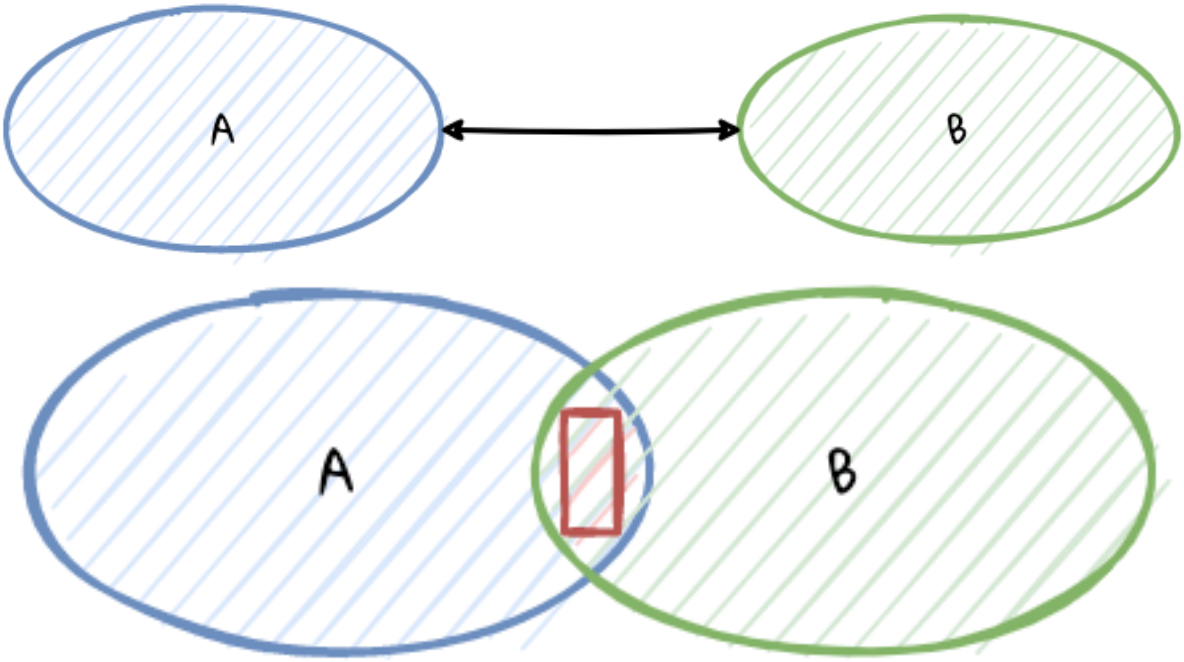



LC Auto Approval Saga
Product
LC Application
Applicant
Validate Product Value
Validate Product Legality
Product Value Validated
LC Application Submitted
Product Legality Validated
Submit LC Application
Orchestrator
Approve LC Application
LC Application Approved
LC Application Submitted
Applicant Credit Validated
Validate Applicant Credit
Legend
Command
Event

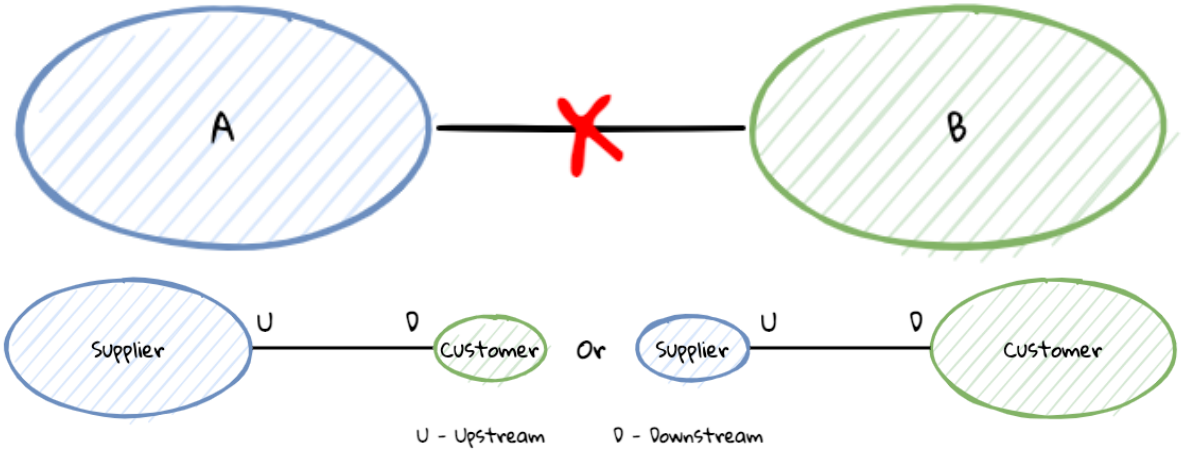

LC Auto Approval Saga
Product
LC Application
Applicant
Validate Product Value
Validate Product Legality
Product Legality Validated
Product Value Validated
Stateless Event Listener
Application Submitted
Submit Application
Stateless Event Listener
Auto Approve
LC Application Approved
Application Submitted
Stateless Event Listener
Applicant Credit Validated
Validate Applicant Credit
Legend
Command
Event

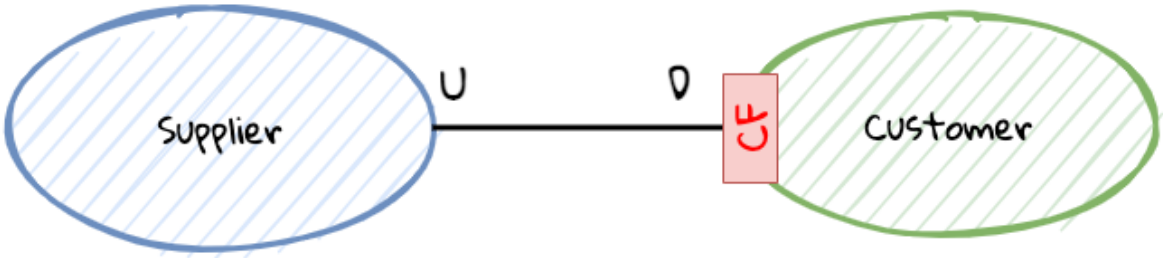
Chapter 9: Integrating with External Systems

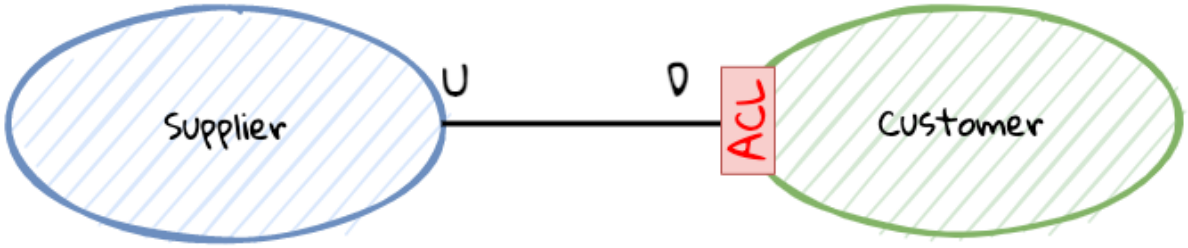




Shared Kernel







Address

USPS

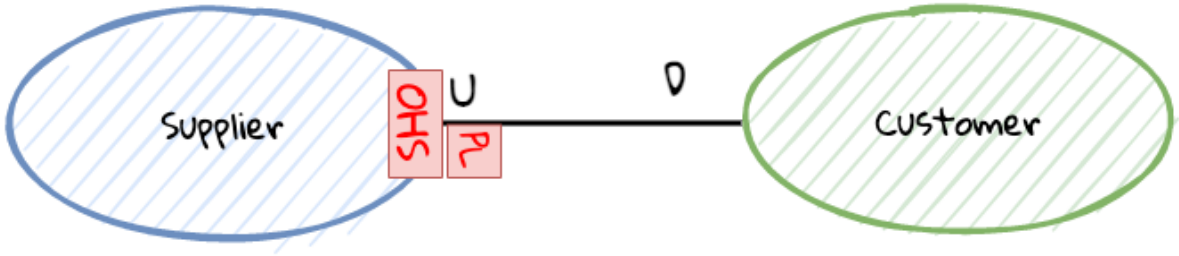
```
{
    line1: string,
    line2: string,
    state: string,
    zip5: number,
    zip4: number
}
```

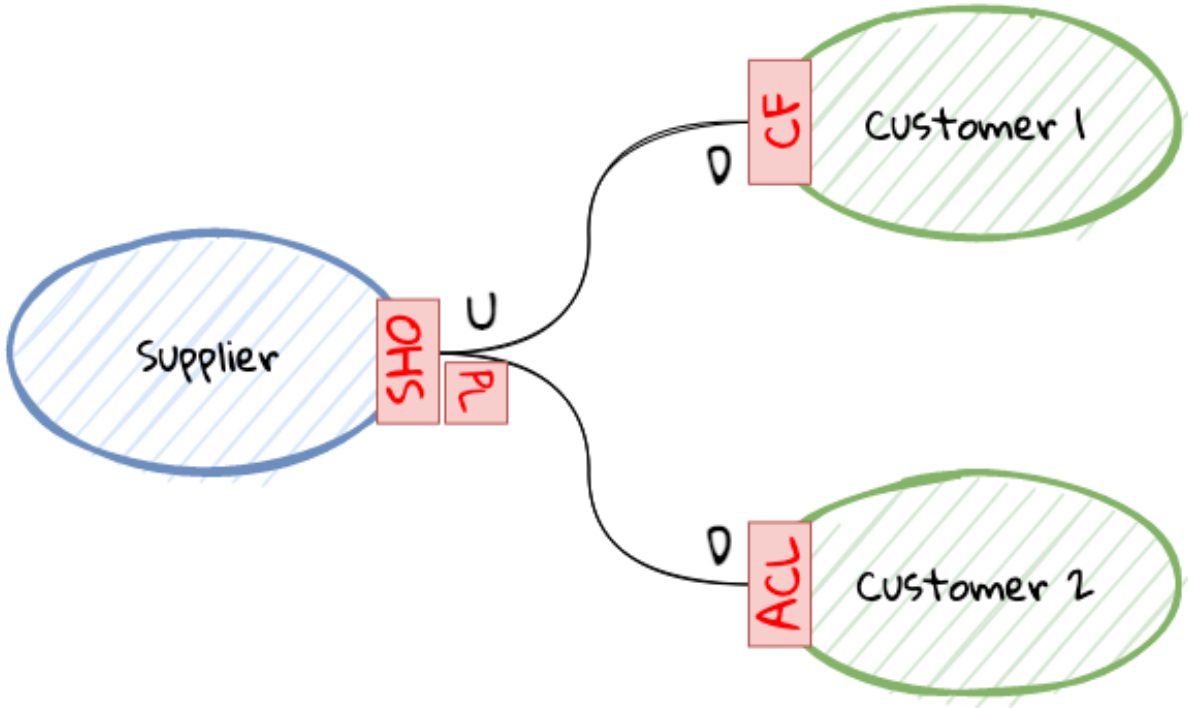
LC Application

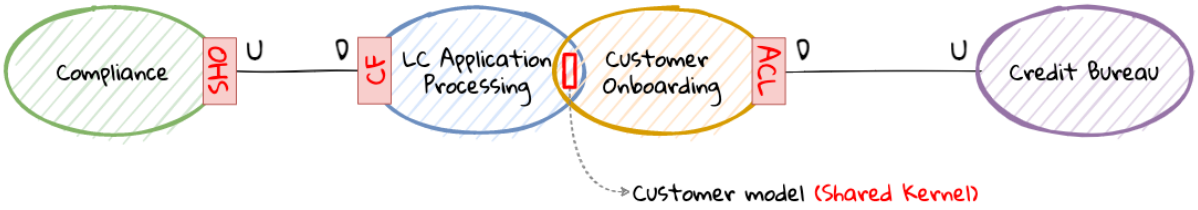
```
{
    line1: string,
    line2: string,
    region: string,
    postalCode: string,
    country: string
}
```

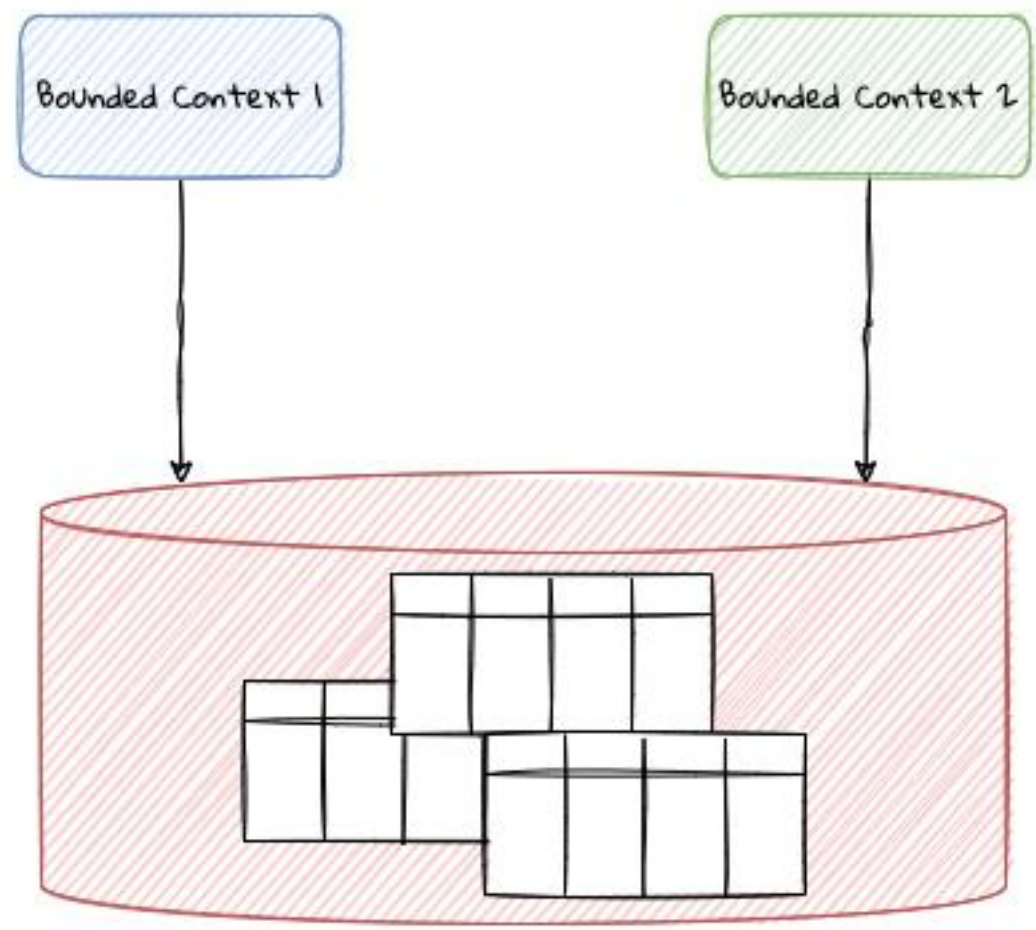
Canada Post

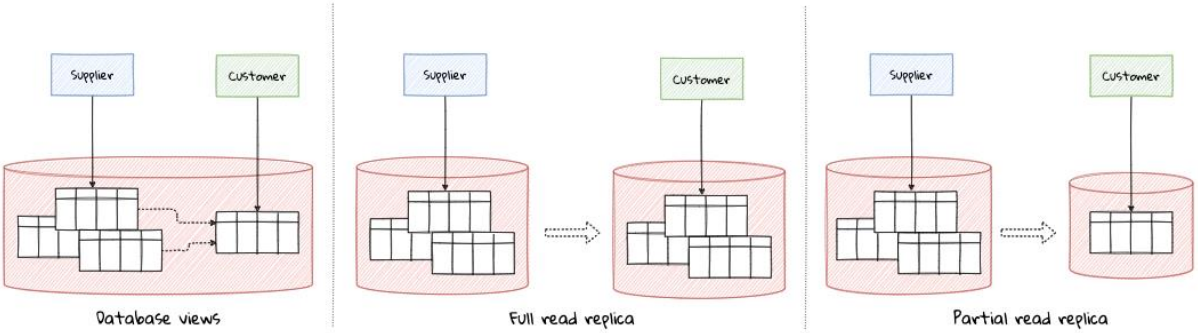
```
{
    line1: string,
    line2: string,
    province: string,
    postalCode: string
}
```

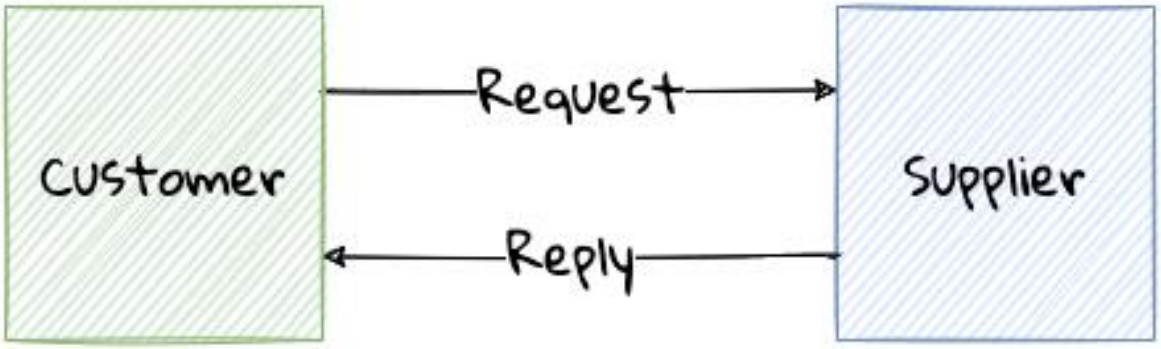


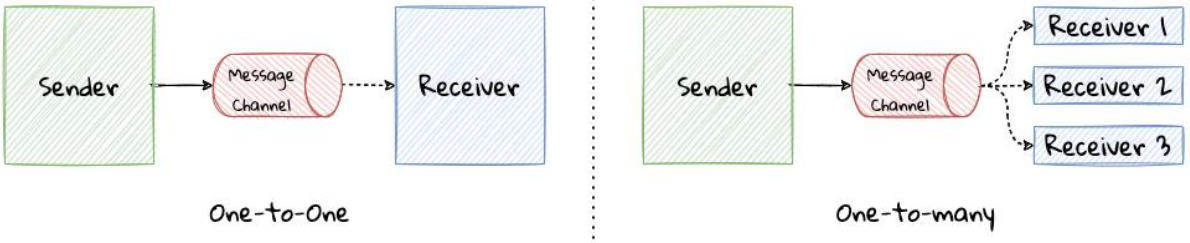



Compliance
OHS
U
D
CF
LC Application Processing
Customer Onboarding
ACL
D
U
Credit Bureau
Customer model (Shared Kernel)


Bounded Context 1
Bounded Context 2


Supplier
Customer
Database views
Supplier
Customer
Full read replica
Supplier
Customer
Partial read replica

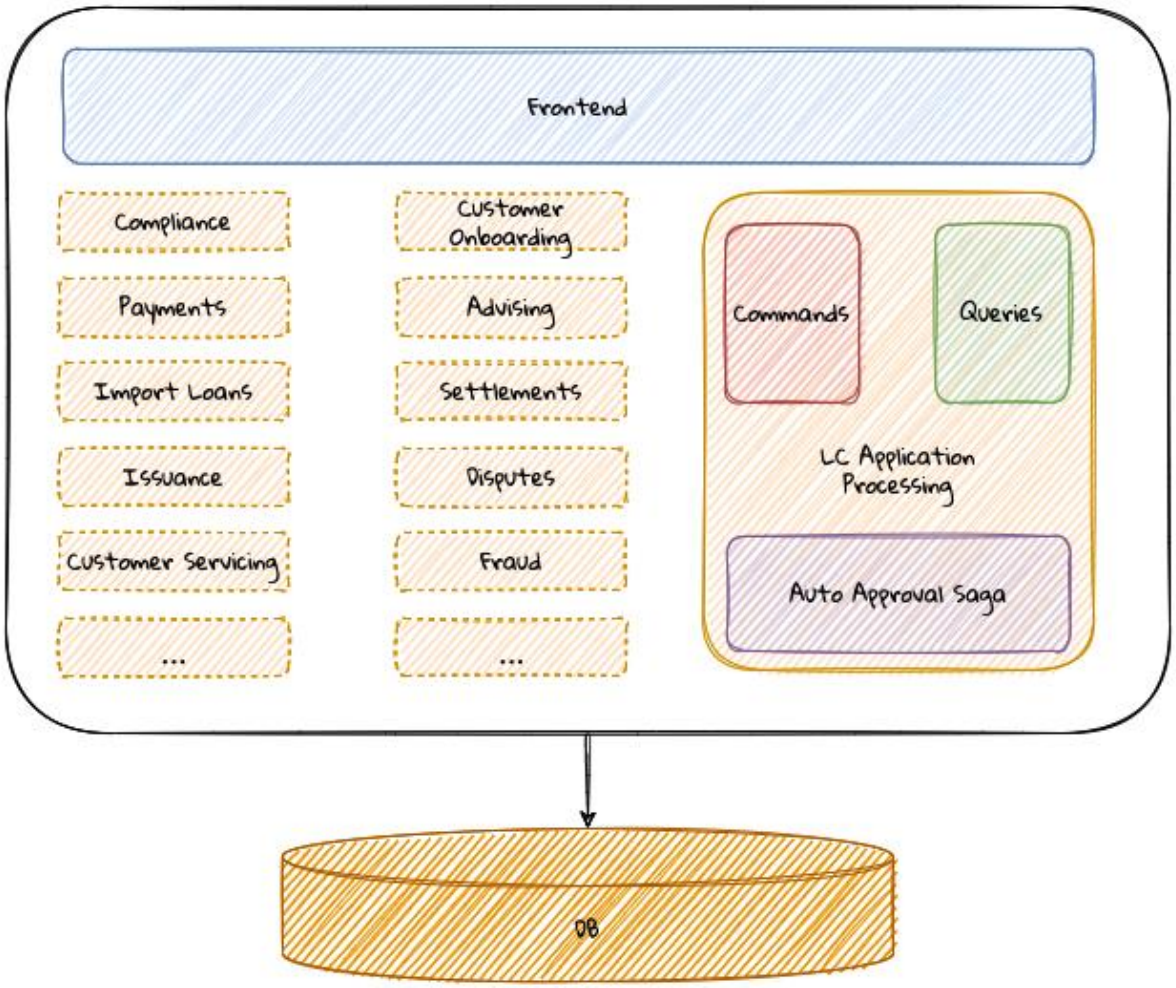


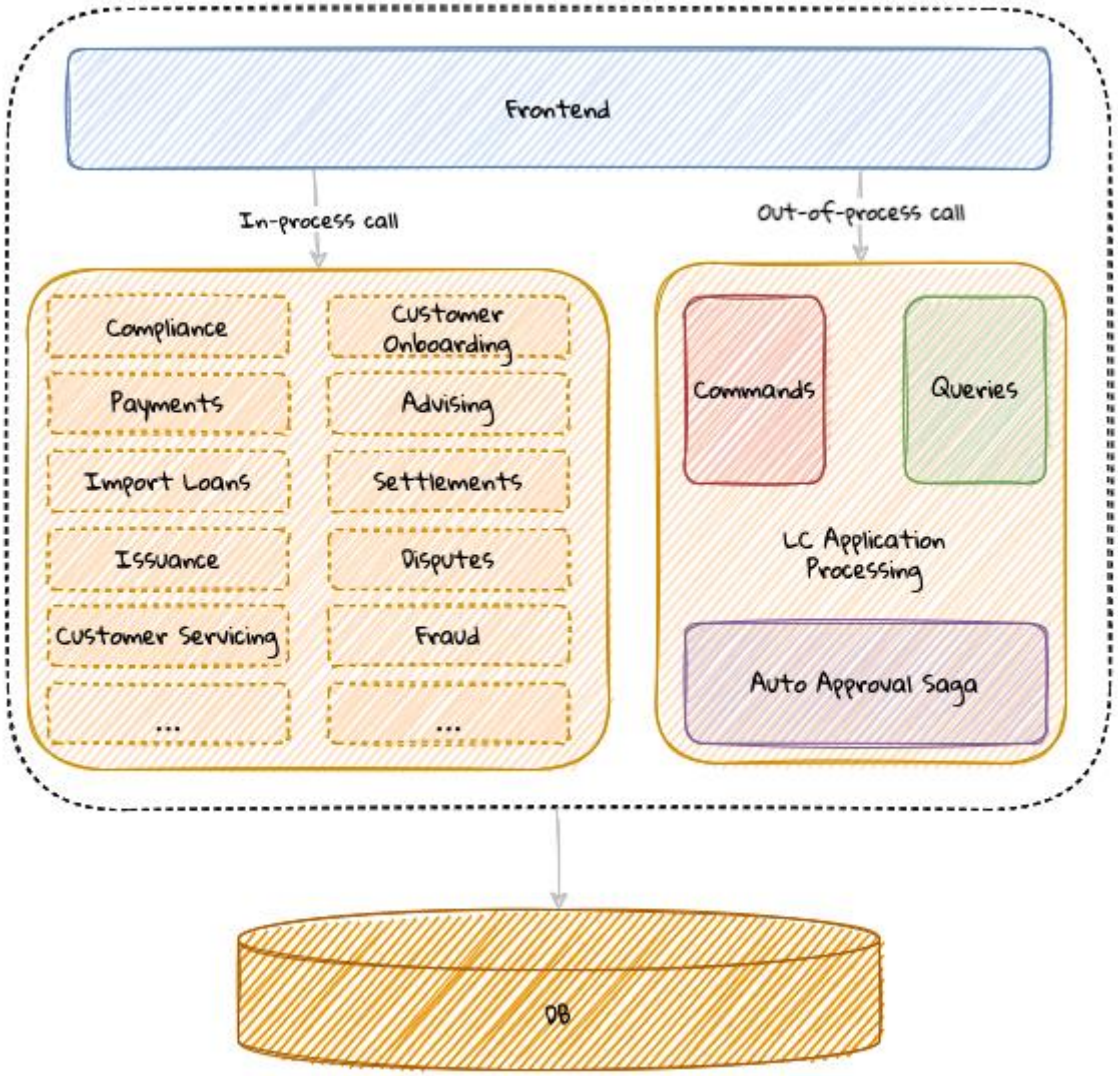


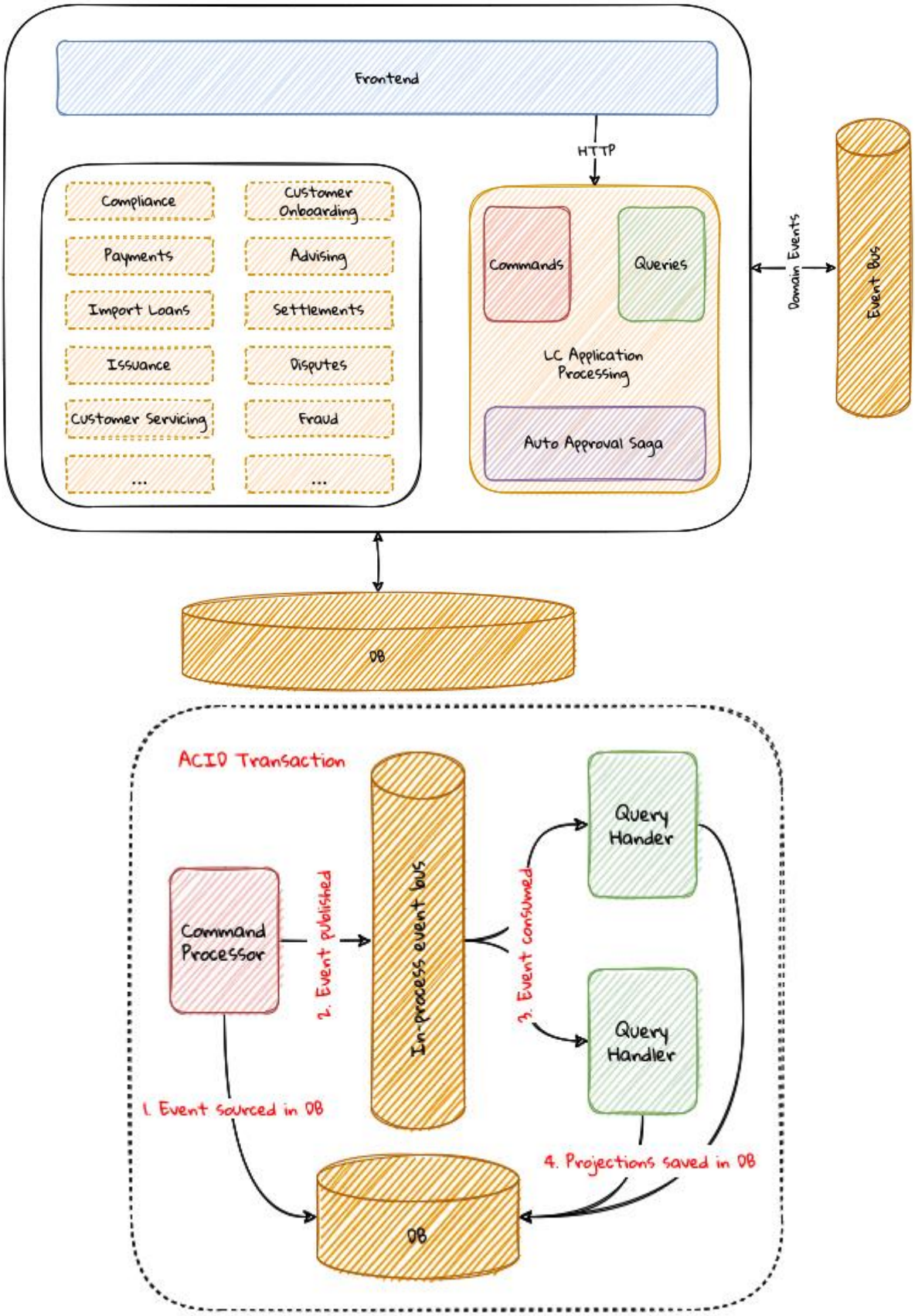
One-to-One

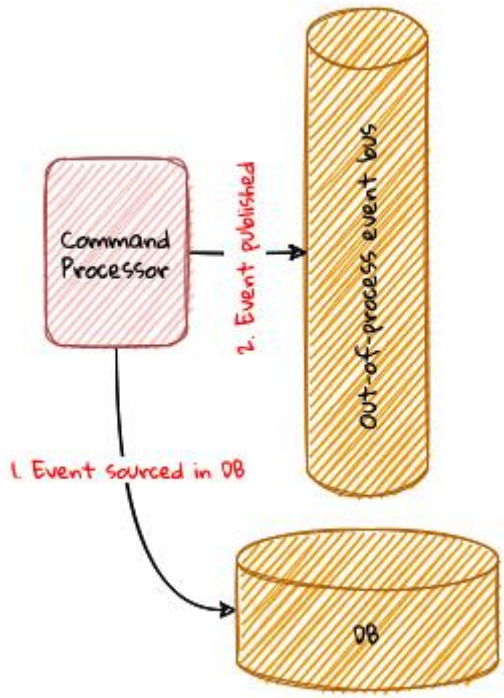
One-to-many

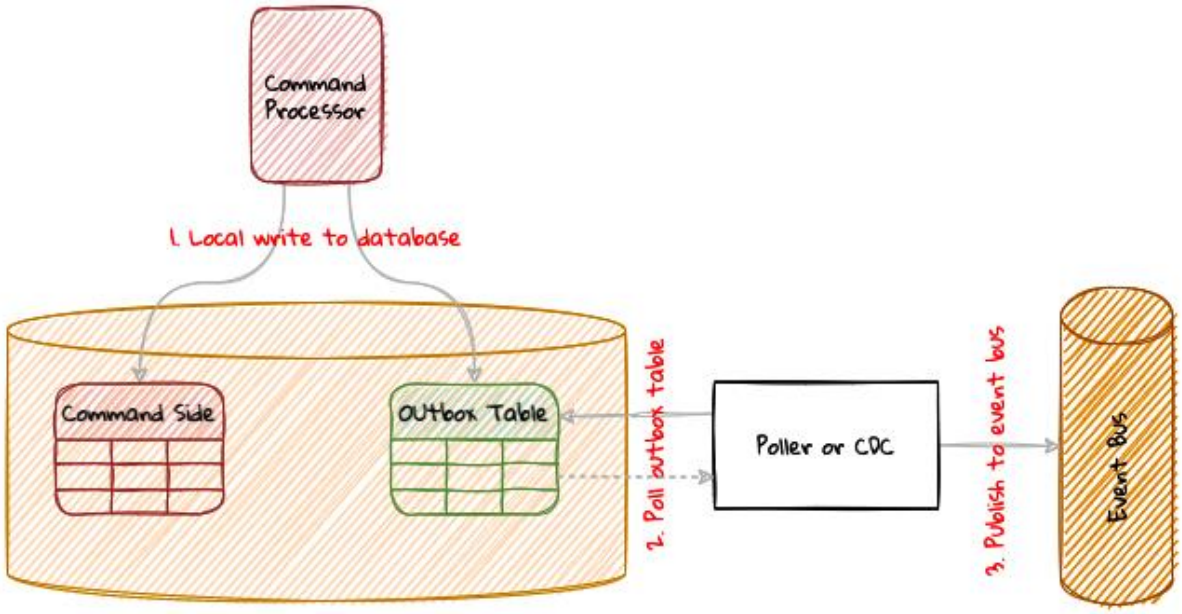
Chapter 10: Beginning the decomposition journey

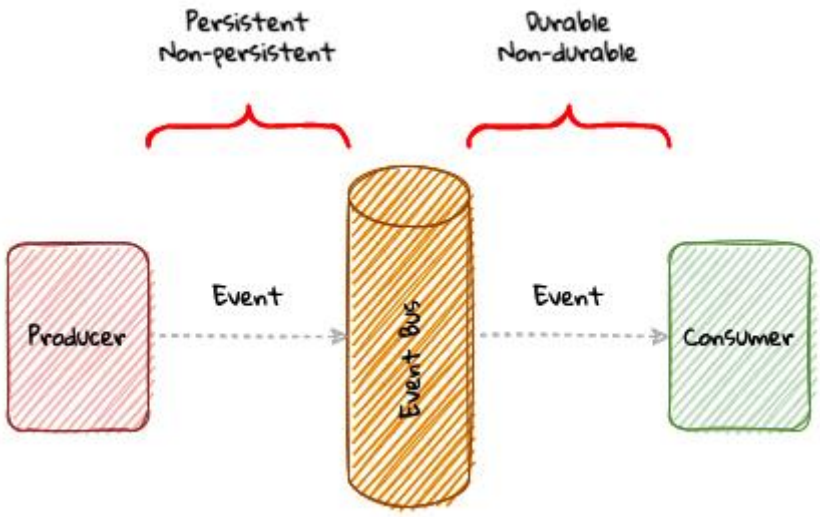


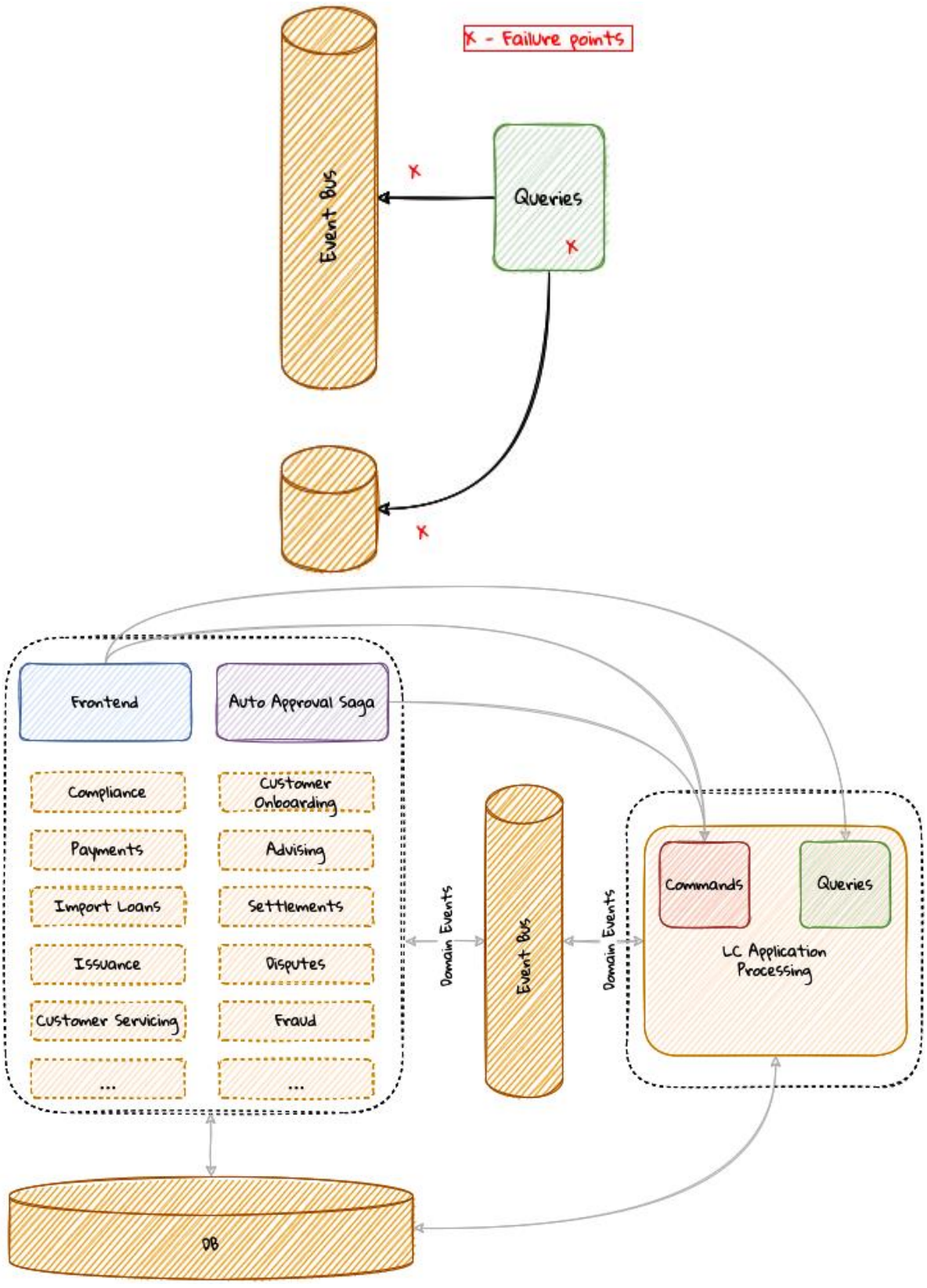

Frontend
In-process call
Out-of-process call
Compliance
Customer Onboarding
Payments
Advising
Import Loans
Settlements
Issuance
Disputes
Customer Servicing
Fraud
...
...
Commands
Queries
LC Application Processing
Auto Approval Saga
DB

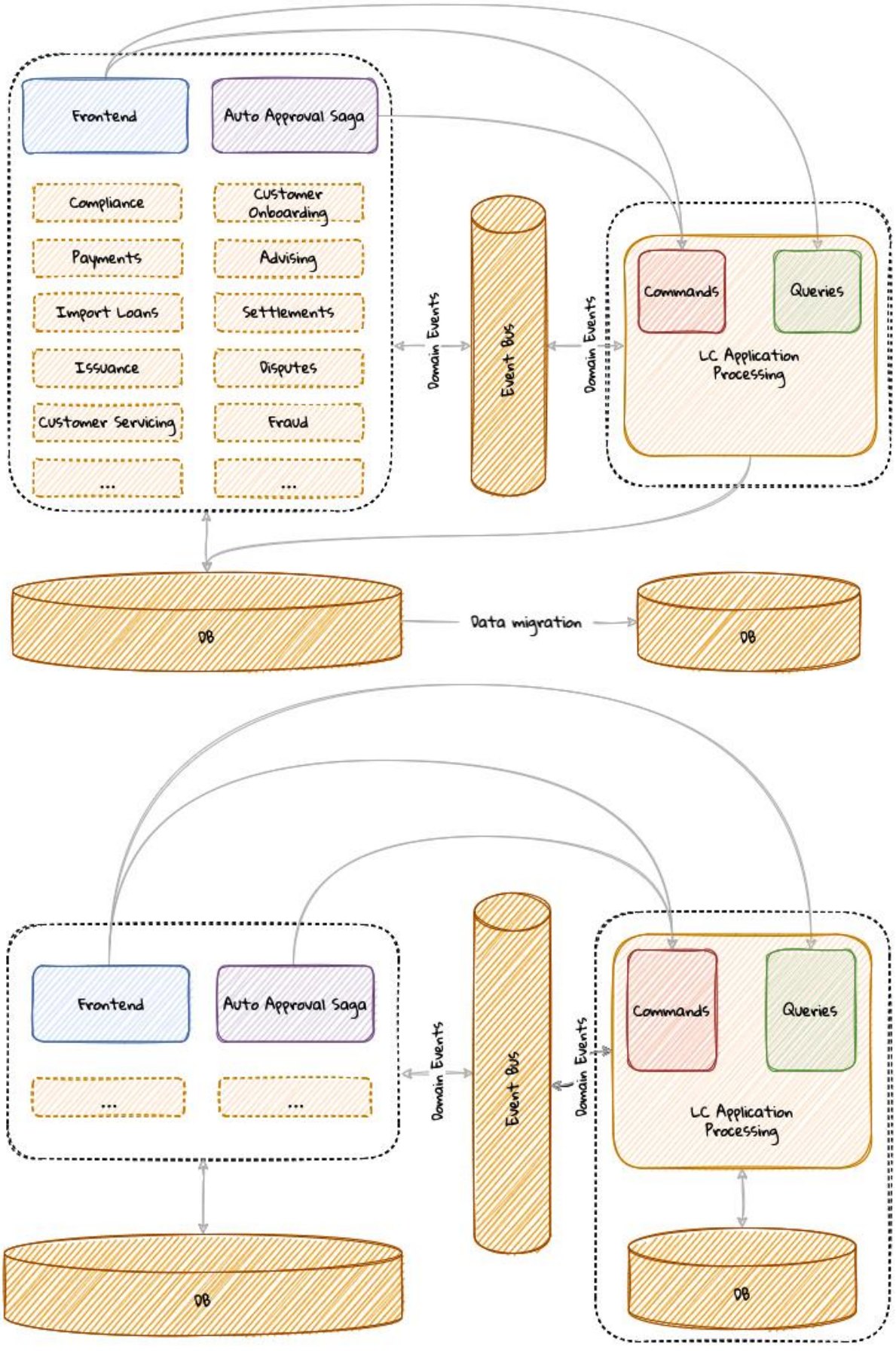

Frontend
HTTP
Compliance
Customer Onboarding
Payments
Advising
Import Loans
Settlements
Issuance
Disputes
Customer Servicing
Fraud
...
...
Commands
Queries
LC Application Processing
Auto Approval Saga
Domain Events
Event Bus
DB
ACID Transaction
Command Processor
2. Event published
In-process event bus
3. Event consumed
Query Hander
Query Handler
1. Event sourced in DB
4. Projections saved in DB
DB


Command Processor
2. Event published
Out-of-process event bus
1. Event sourced in DB
DB

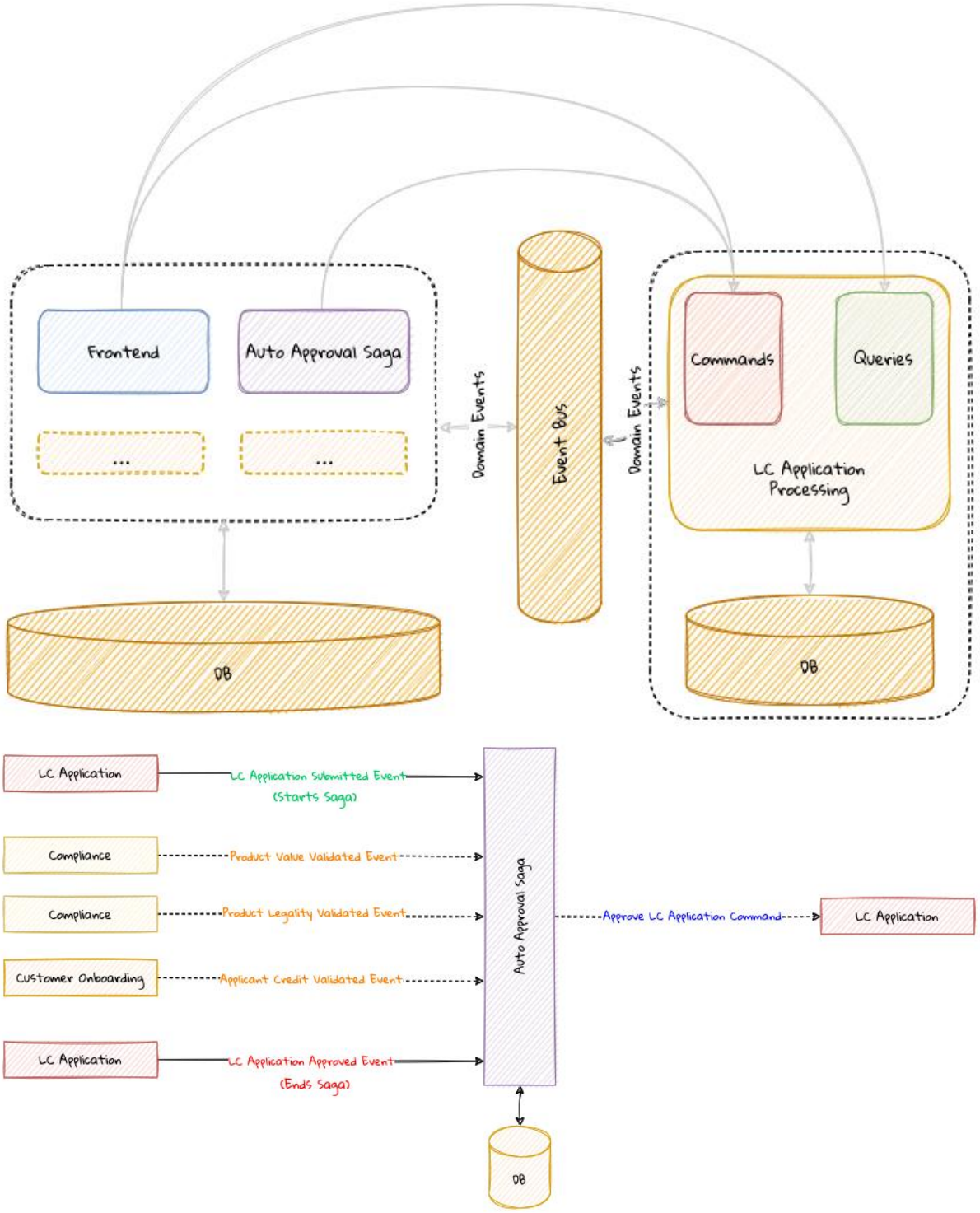

Command Processor
1. Local write to database
Command Side
Outbox Table
2. Poll outbox table
Poller or CDC
3. Publish to event bus
Event Bus

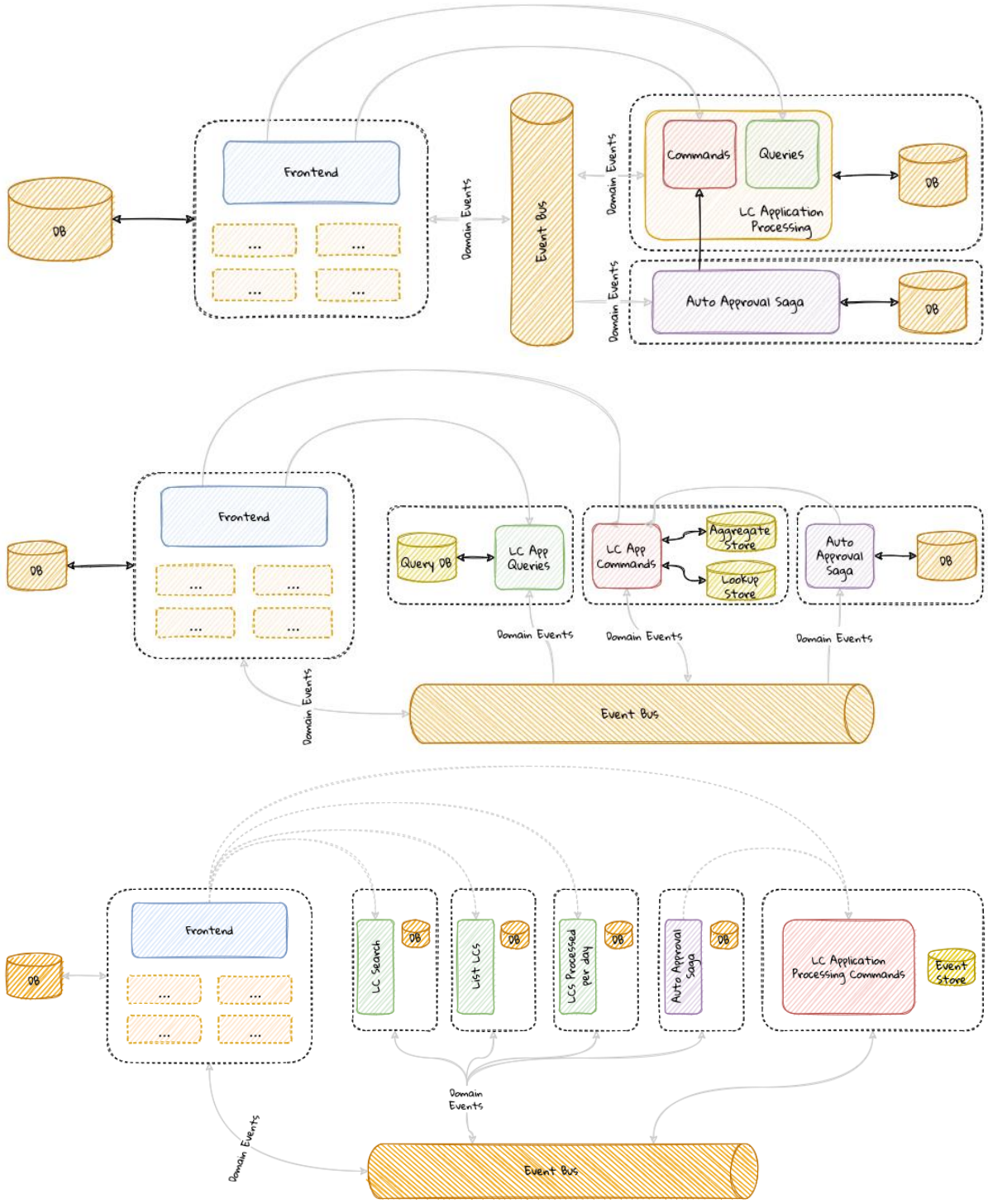

Persistent
Non-persistent
Durable
Non-durable
Producer
Event
Event Bus
Event
Consumer

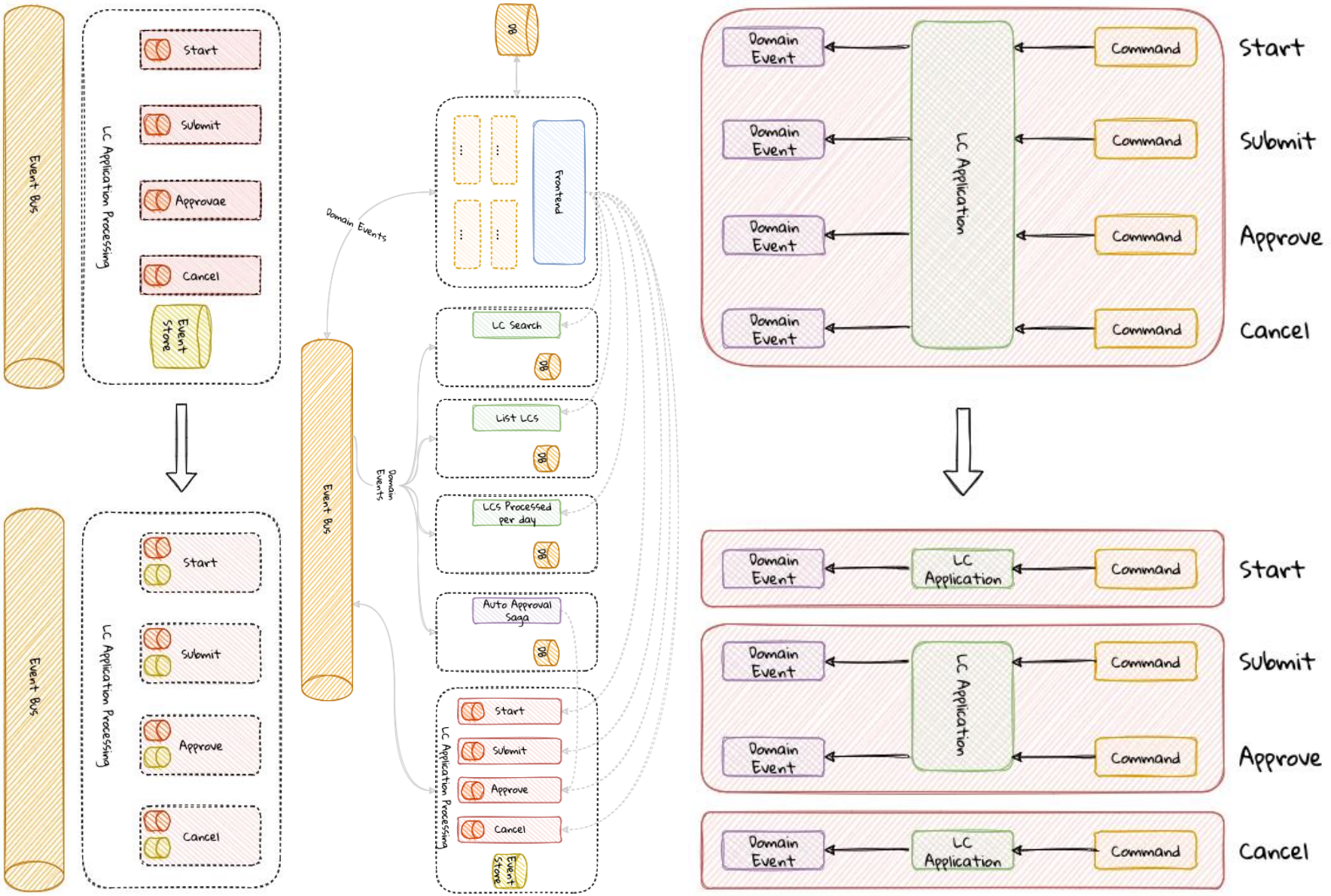

X - Failure points
Event Bus
Queries
Frontend
Auto Approval Saga
Compliance
Customer Onboarding
Payments
Advising
Import Loans
Settlements
Issuance
Disputes
Customer Servicing
Fraud
...
...
Domain Events
Event Bus
Domain Events
Commands
Queries
LC Application Processing
DB

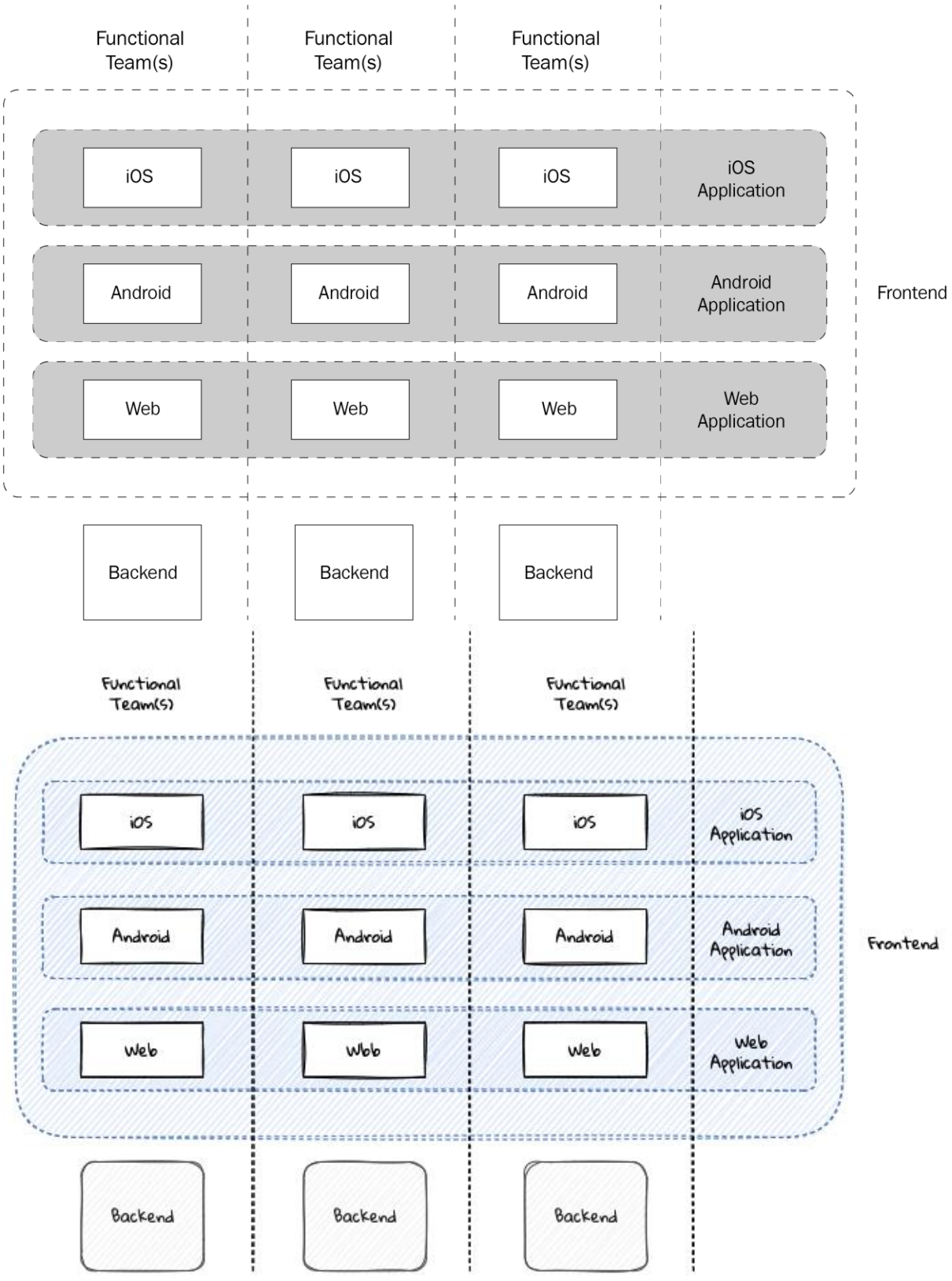

Frontend
Auto Approval Saga
Compliance
Customer Onboarding
Payments
Advising
Import Loans
Settlements
Issuance
Disputes
Customer Servicing
Fraud
...
...
Domain Events
Event Bus
Domain Events
Commands
Queries
LC Application Processing
DB
Data migration
DB
Frontend
Auto Approval Saga
...
...
Domain Events
Event Bus
Domain Events
Commands
Queries
LC Application Processing
DB
DB

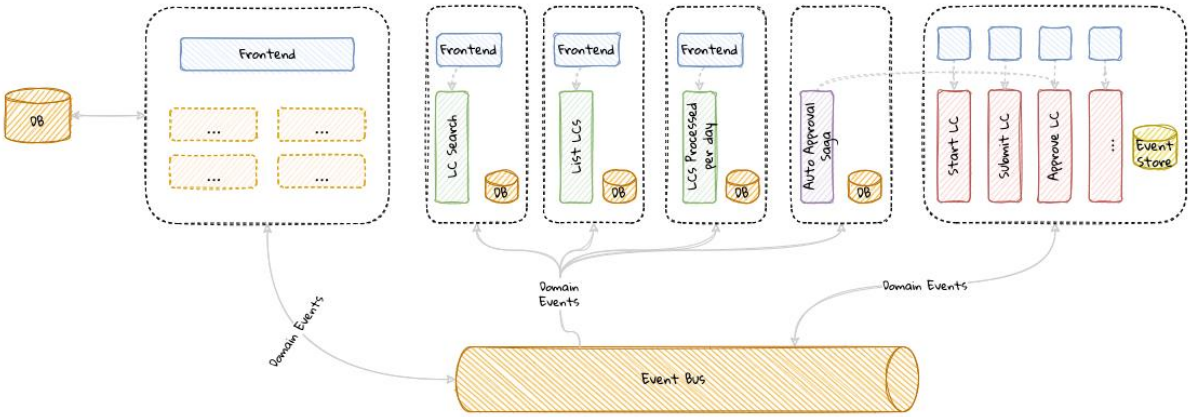
Chapter 11: Decomposing into finer-grained components

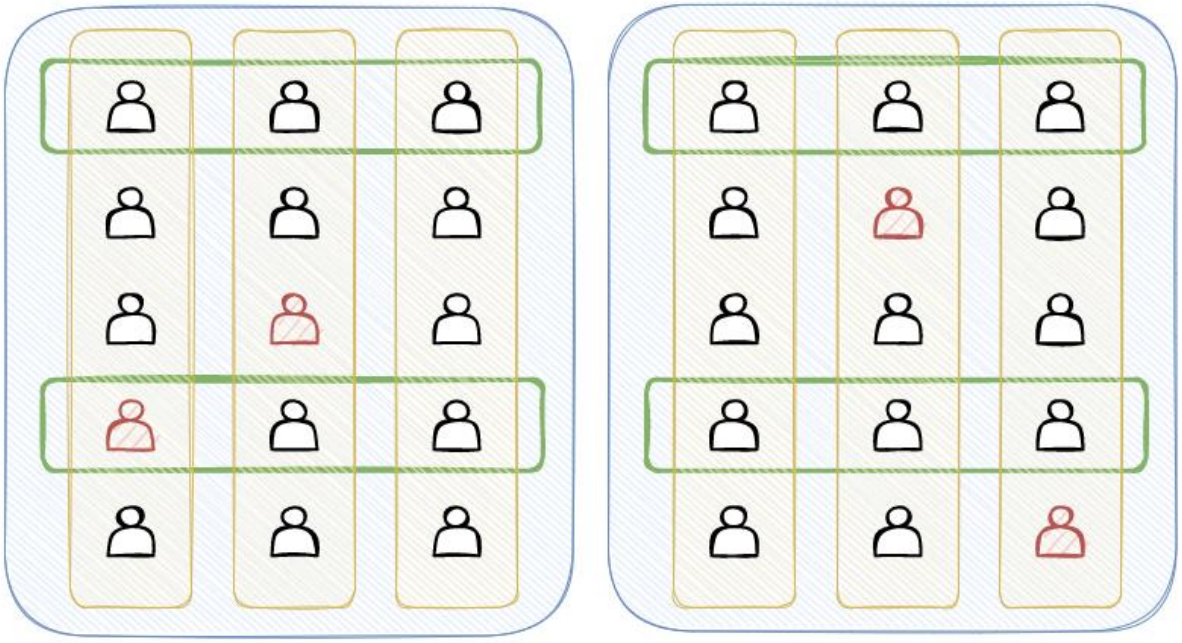



DB
Frontend
...
...
...
...
Domain Events
Event Bus
Domain Events
Commands
Queries
DB
LC Application Processing
Domain Events
Auto Approval Saga
DB
DB
Frontend
...
...
...
...
Query DB
LC App Queries
LC App Commands
Aggregate Store
Lookup Store
Auto Approval Saga
DB
Domain Events
Domain Events
Domain Events
Domain Events
Event Bus
DB
Frontend
...
...
...
...
LC Search
DB
List LCs
DB
LCs Processed per day
DB
Auto Approval Saga
DB
LC Application Processing Commands
Event Store
Domain Events
Domain Events
Event Bus

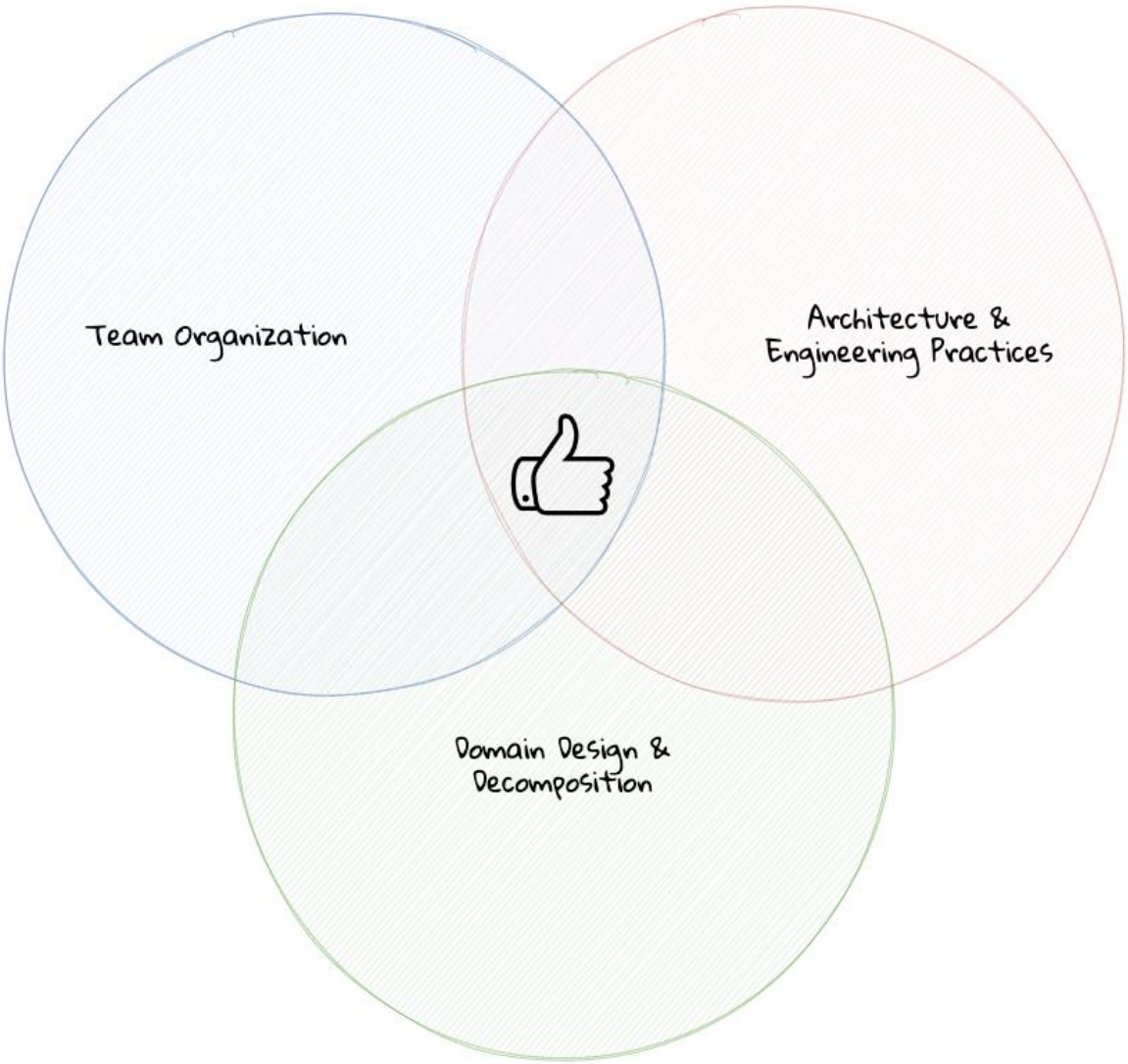

Event Bus
LC Application Processing
Start
Submit
Approvae
Cancel
Event Store
Domain Events
DB
Frontend
LC Search
List LCS
LCS Processed per day
Auto Approval Saga
Approve
Domain Event
Command
LC Application
Start
Submit
Approve
Cancel


Functional Team(s)
Functional Team(s)
Functional Team(s)
iOS
iOS
iOS
iOS Application
Android
Android
Android
Android Application
Web
Web
Web
Web Application
Frontend
Backend
Backend
Backend
Functional Team(s)
Functional Team(s)
Functional Team(s)
iOS
iOS
iOS
iOS Application
Android
Android
Android
Android Application
Web
Web
Web
Web Application
Frontend
Backend
Backend
Backend

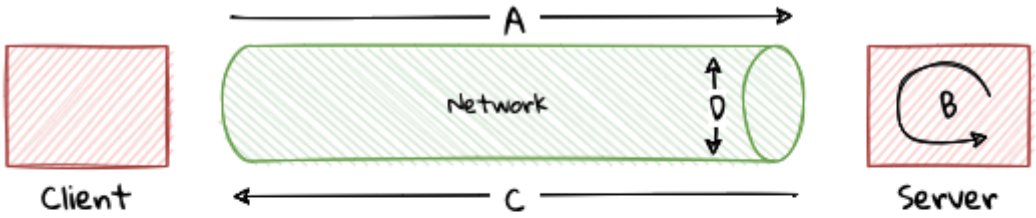




Squad Chapter Tribe Guild


Team Organization
Architecture &
Engineering Practices
Domain Design &
Decomposition

Chapter 12: Beyond Functional Requirements

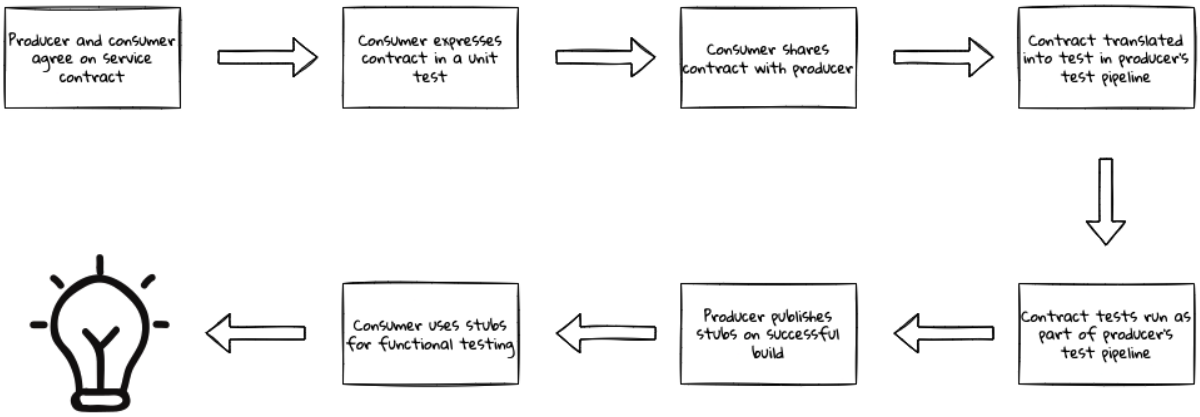


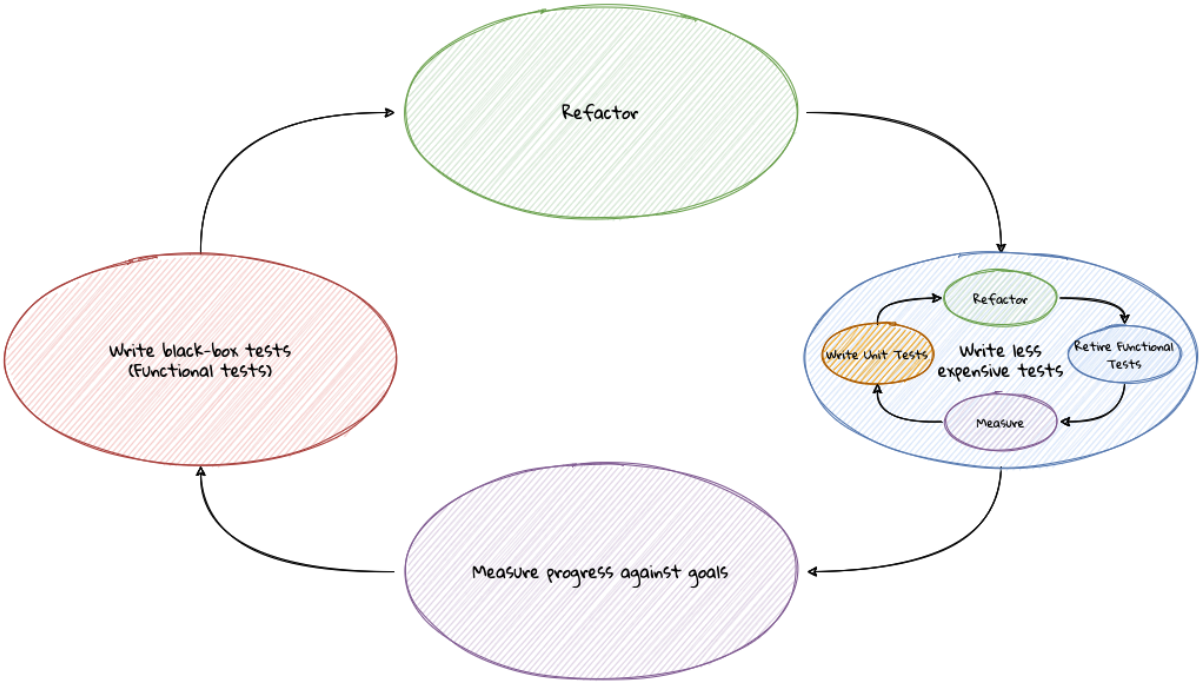
A - Amount of time taken for a request to reach the server

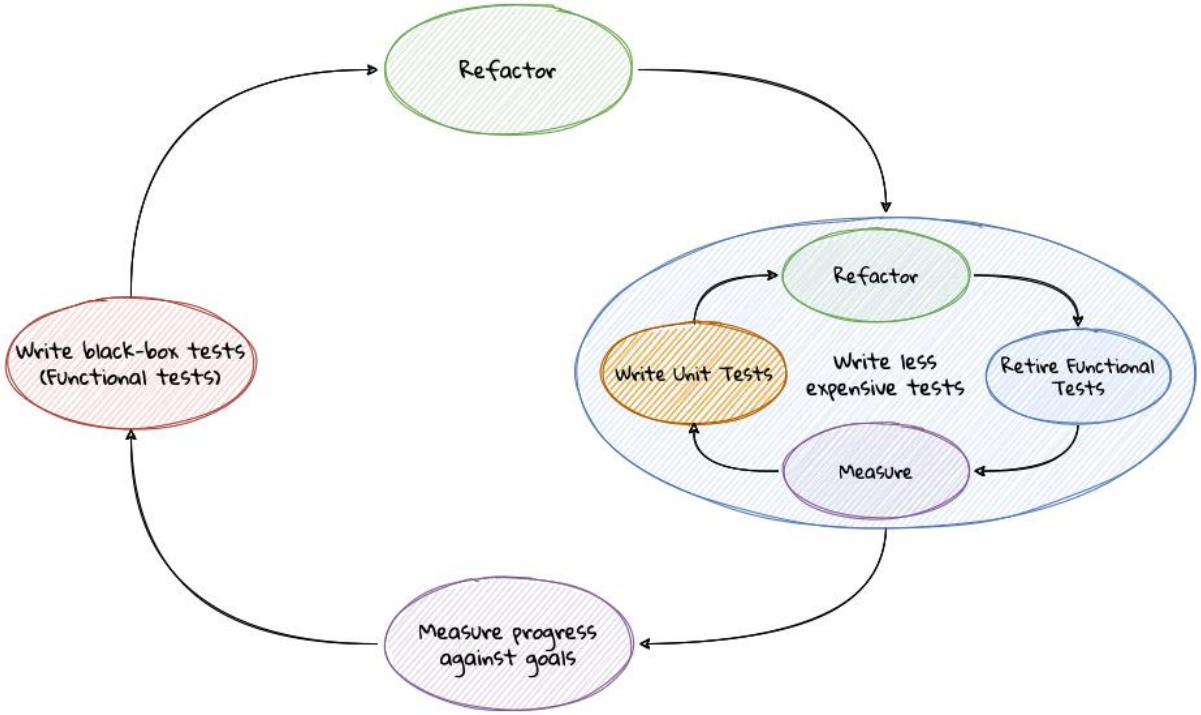
B - Processing time on the server

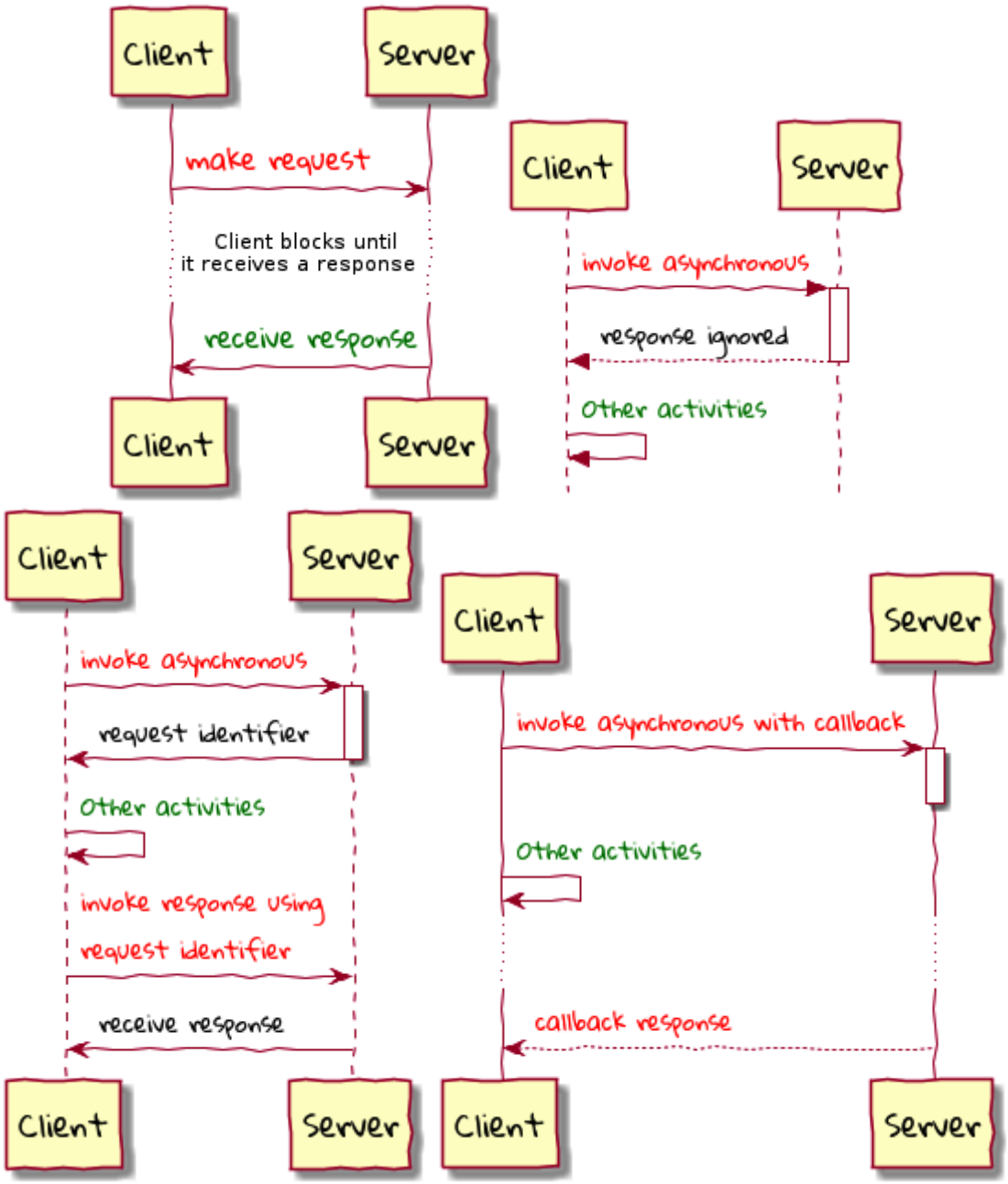
C - Amount of time taken for the response to reach the consumer

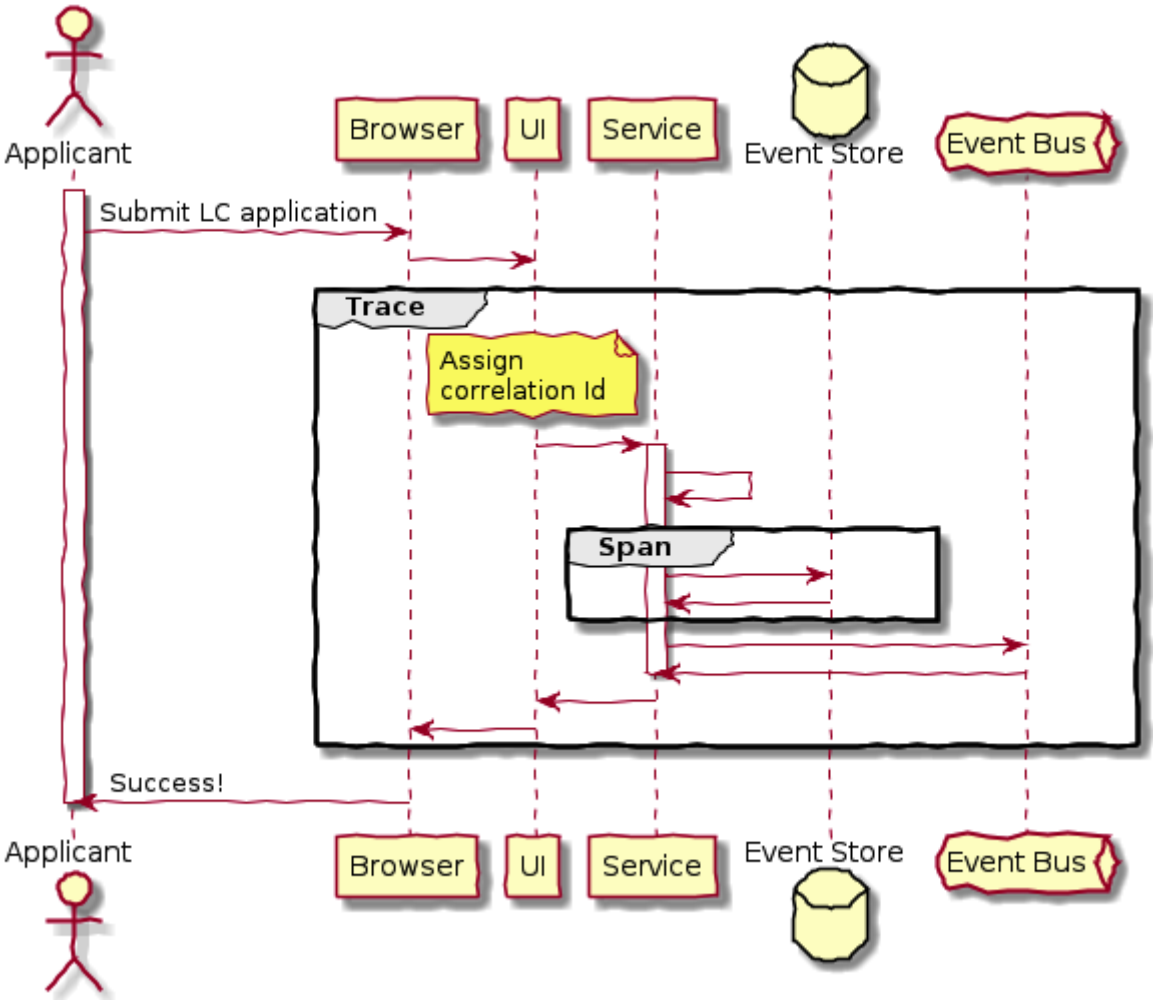
D - Maximum advertised capacity of the network




Refactor
Write black-box tests
(Functional tests)
Refactor
Write Unit Tests
Write less
expensive tests
Retire Functional
Tests
Measure
Measure progress against goals


Refactor
Write black-box tests
(Functional tests)
Refactor
Write Unit Tests
Write less
expensive tests
Retire Functional
Tests
Measure
Measure progress
against goals


Client
Server
make request
Client blocks until
it receives a response
receive response
Client
Server
Client
Server
invoke asynchronous
response ignored
Other activities
Client
Server
Client
Server
invoke asynchronous
request identifier
Other activities
invoke response using
request identifier
receive response
Client
Server
Client
Server
invoke asynchronous with callback
Other activities
callback response
Client
Server


Applicant
Browser
UI
Service
Event Store
Event Bus
Submit LC application
Trace
Assign
correlation Id
Span
Success!
Applicant
Browser
UI
Service
Event Store
Event Bus