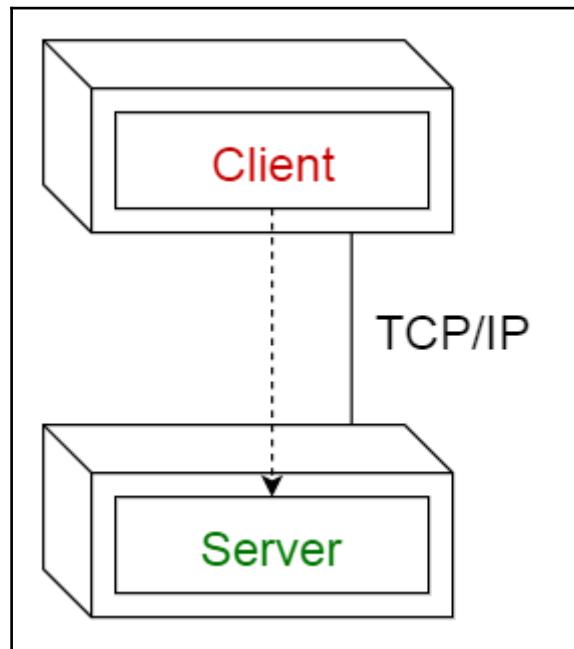
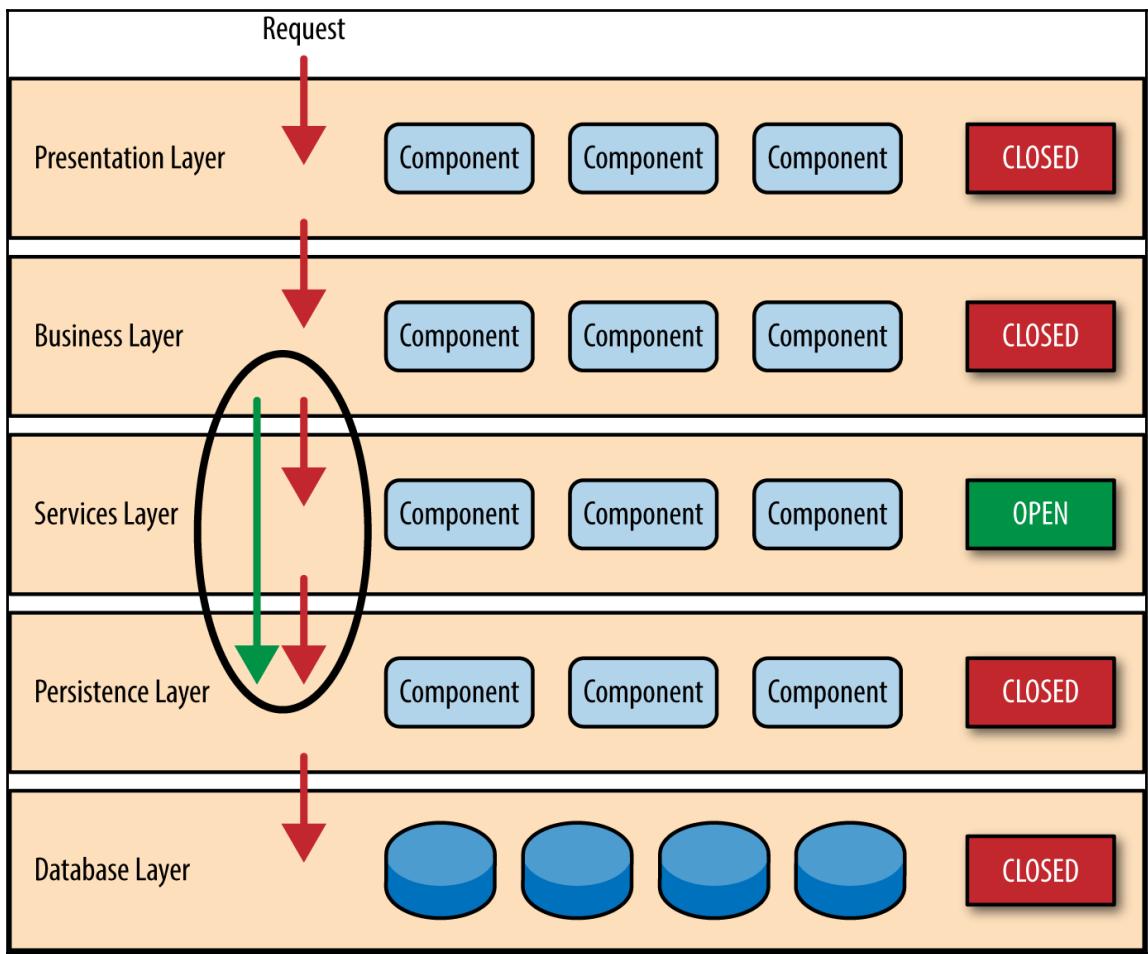
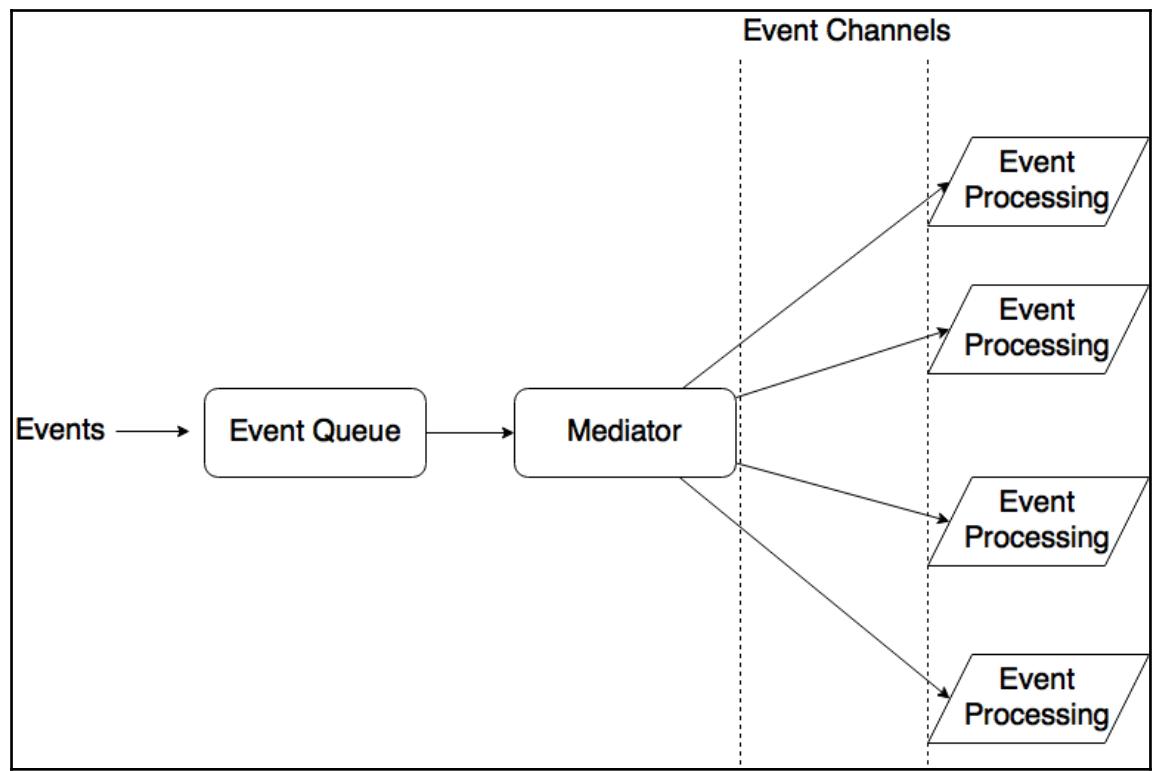
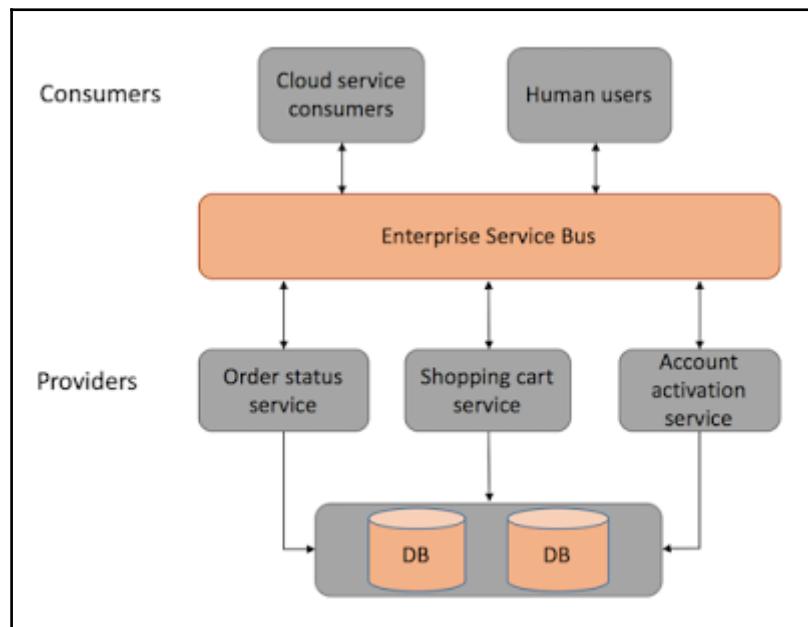
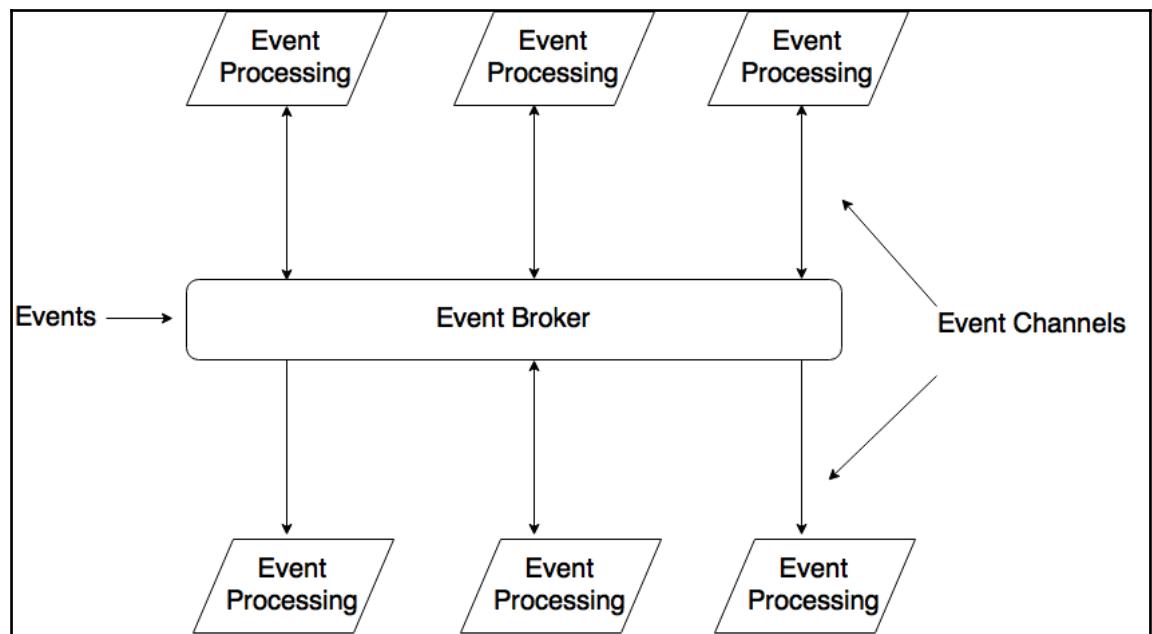


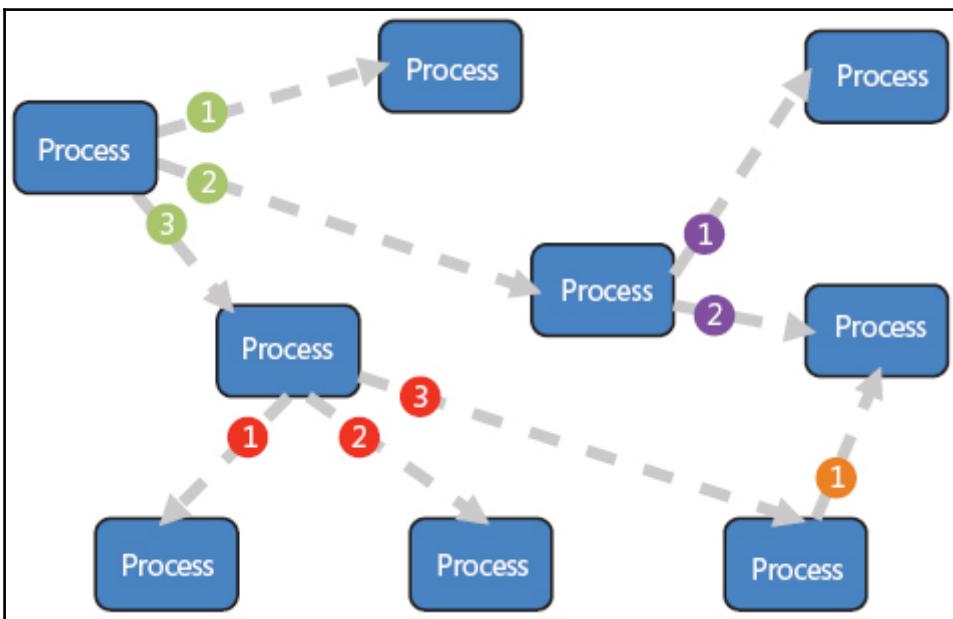
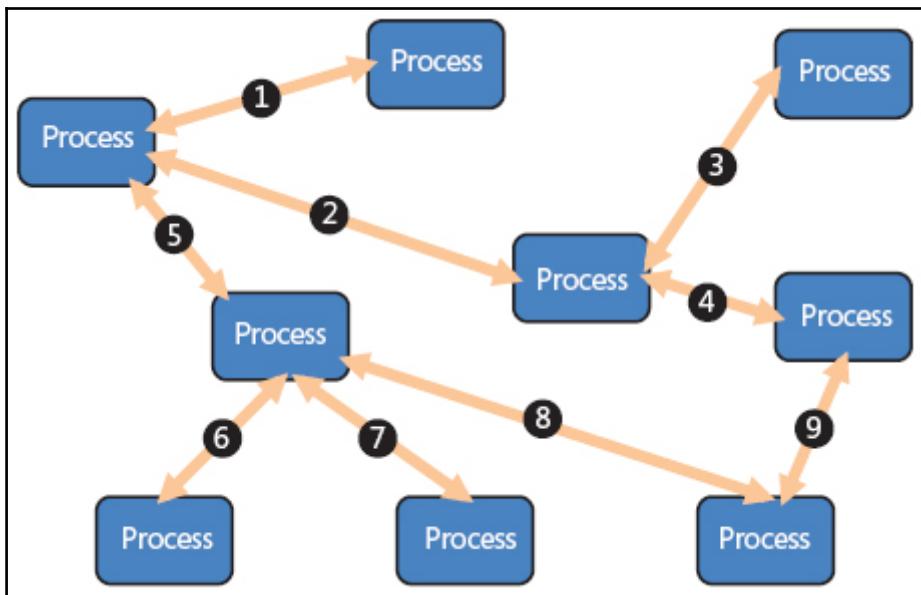
Chapter 1 - Demystifying Software Architecture Patterns

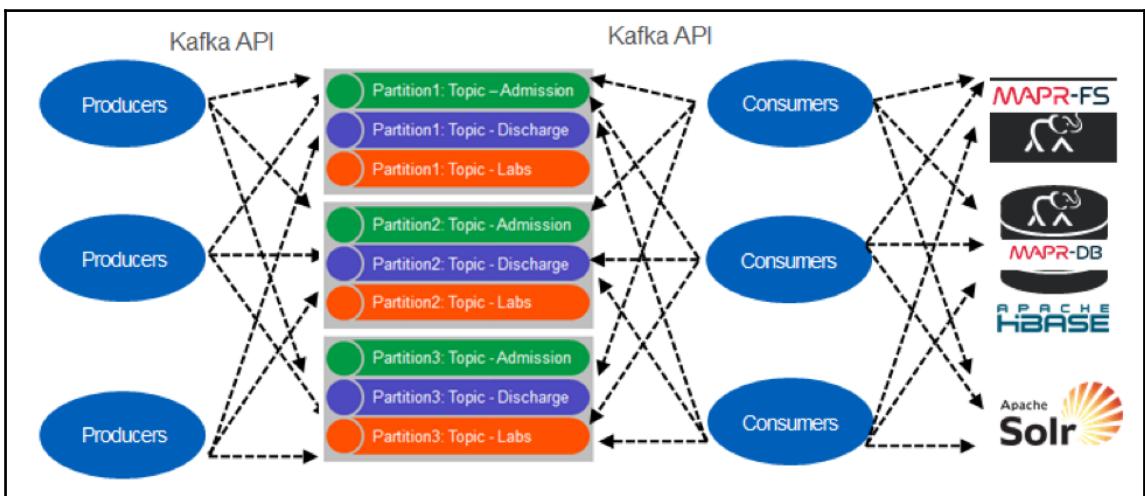
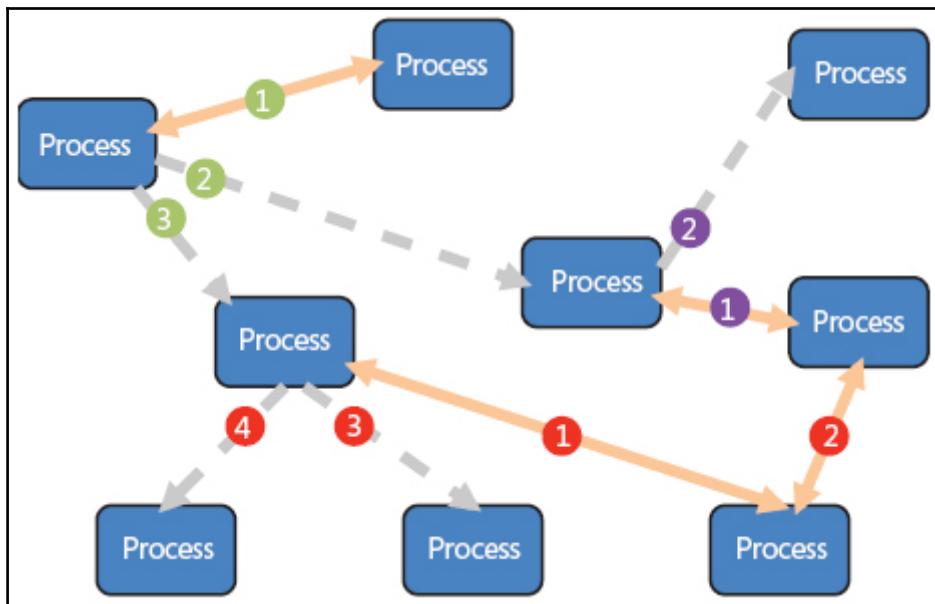


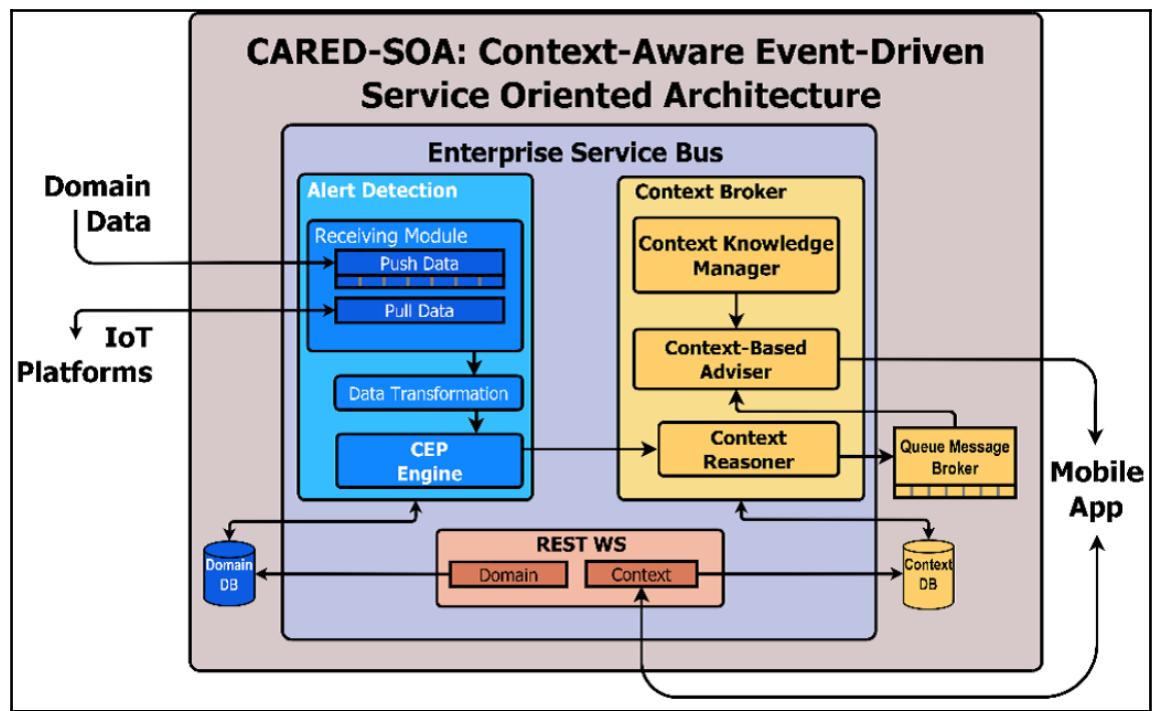




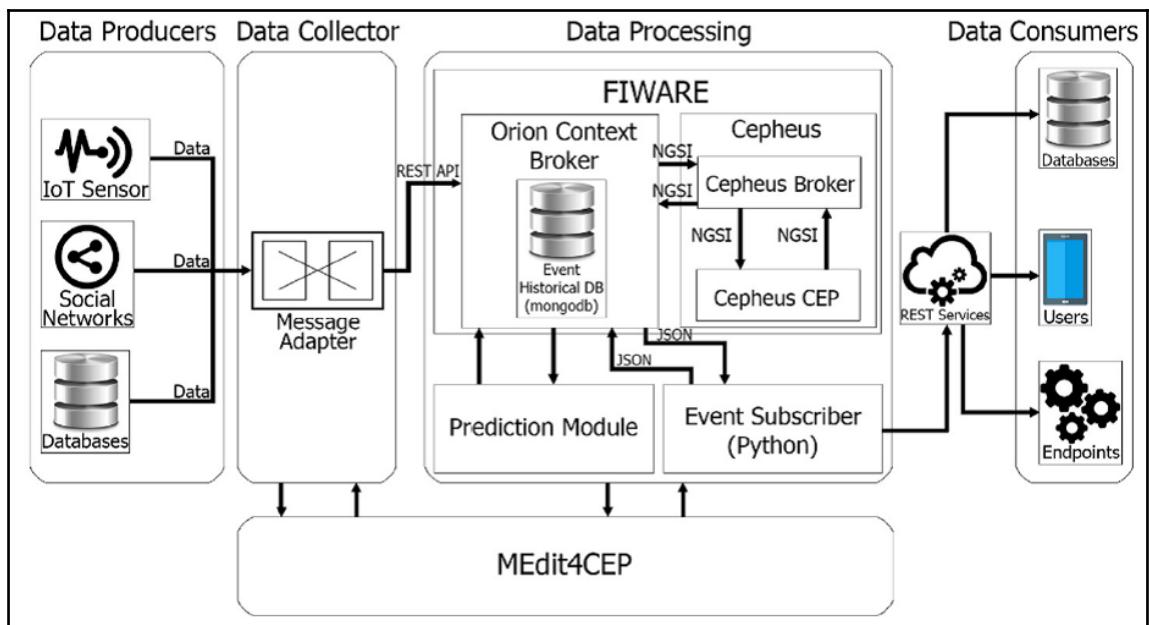




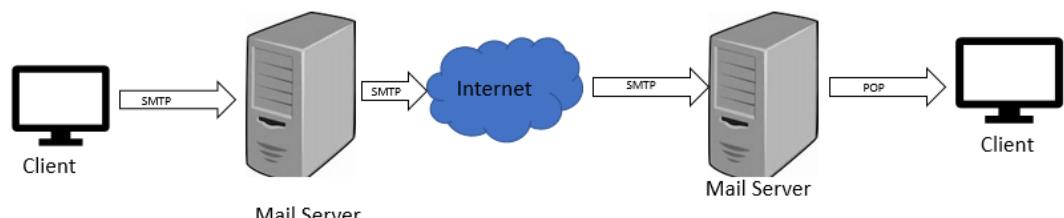
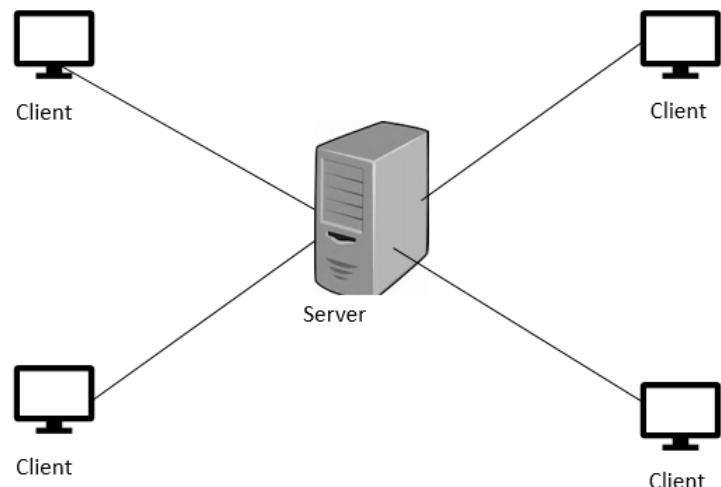


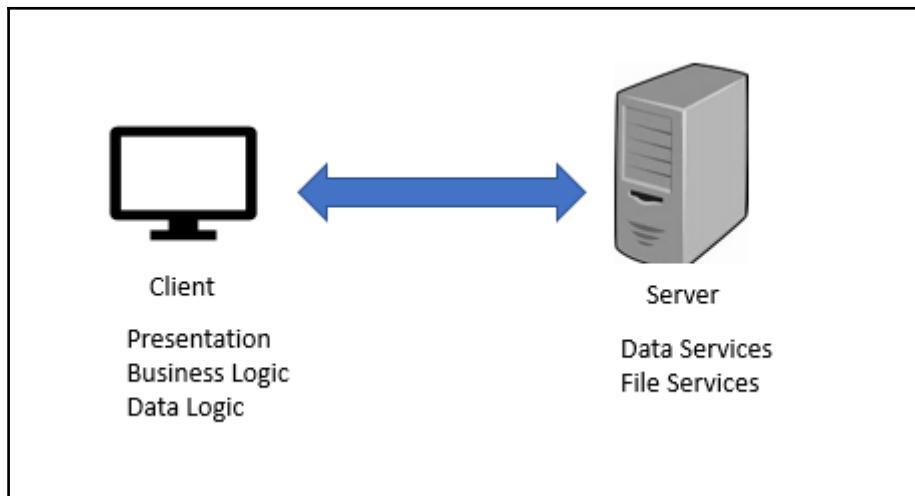
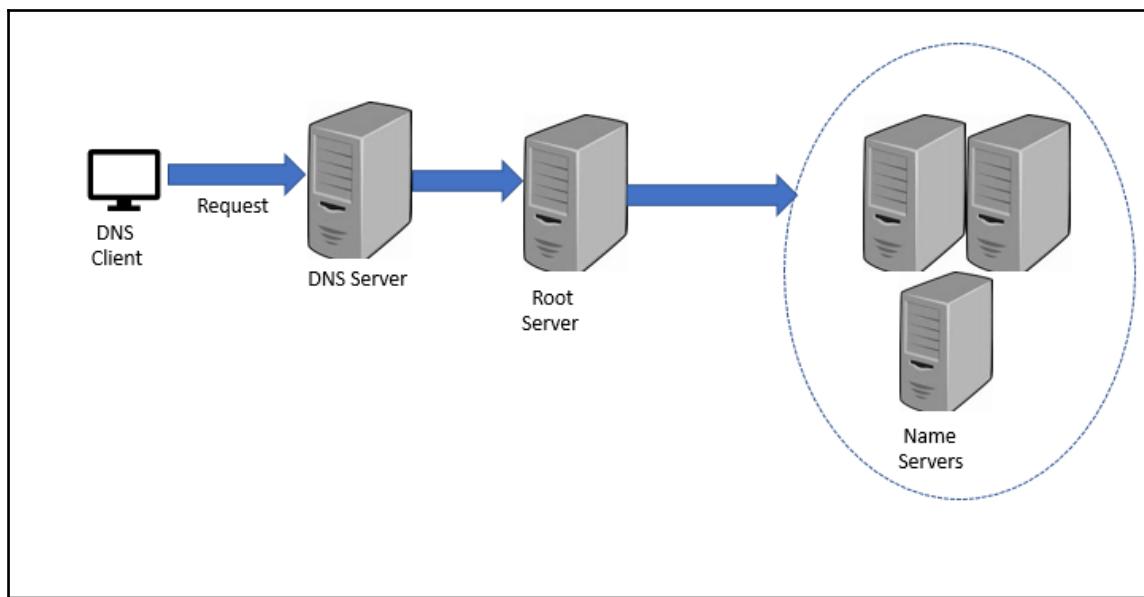


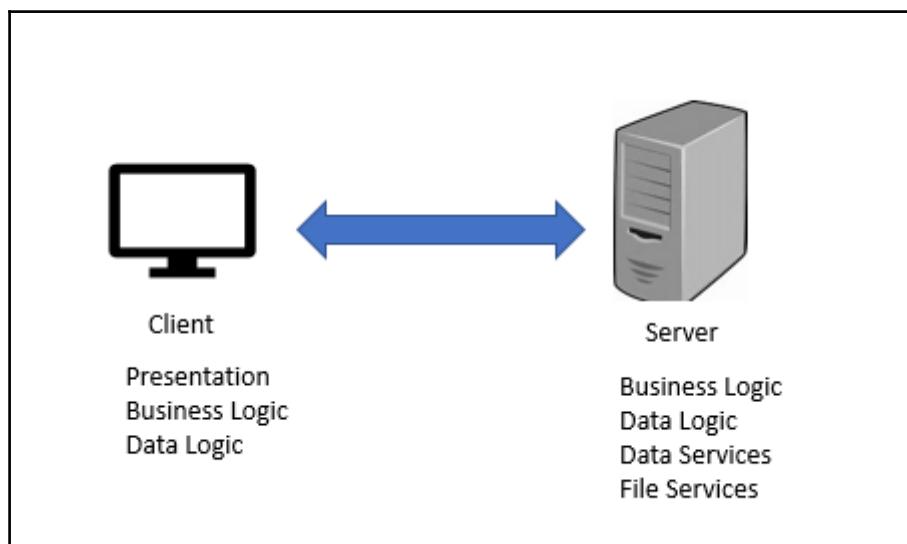
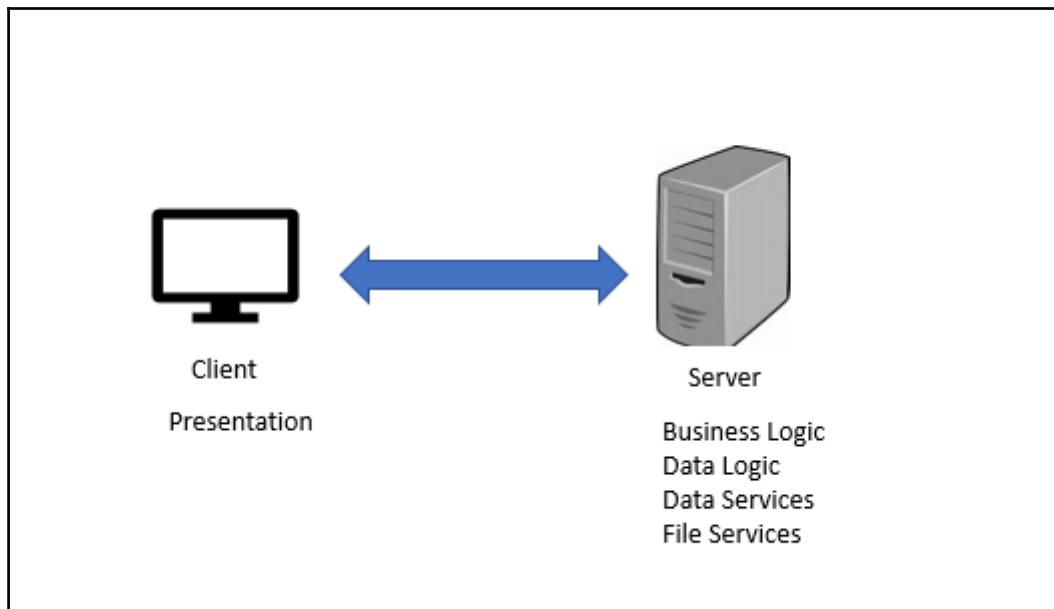
>

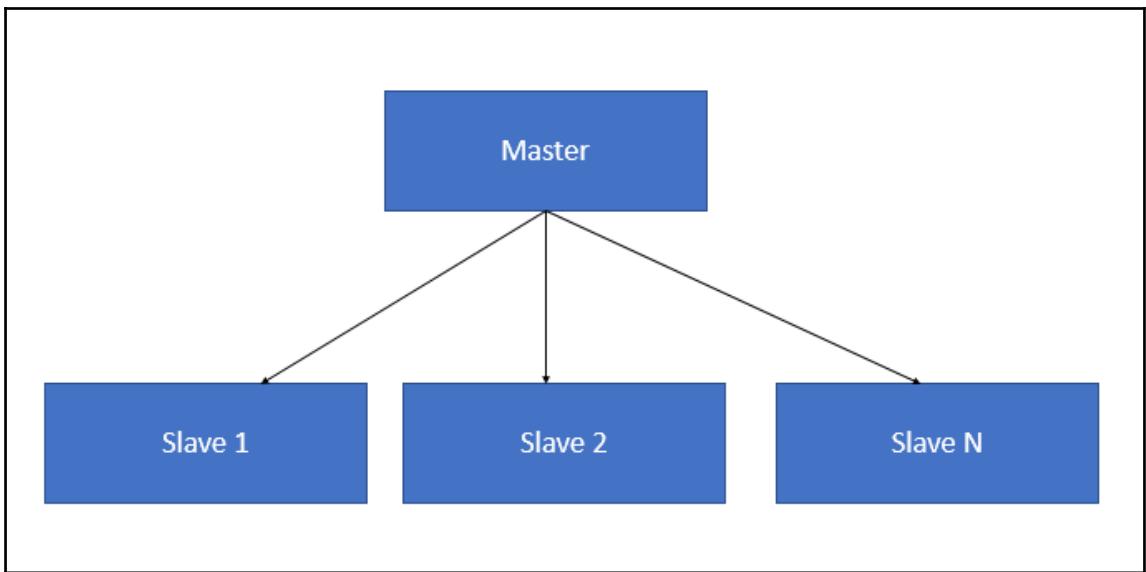


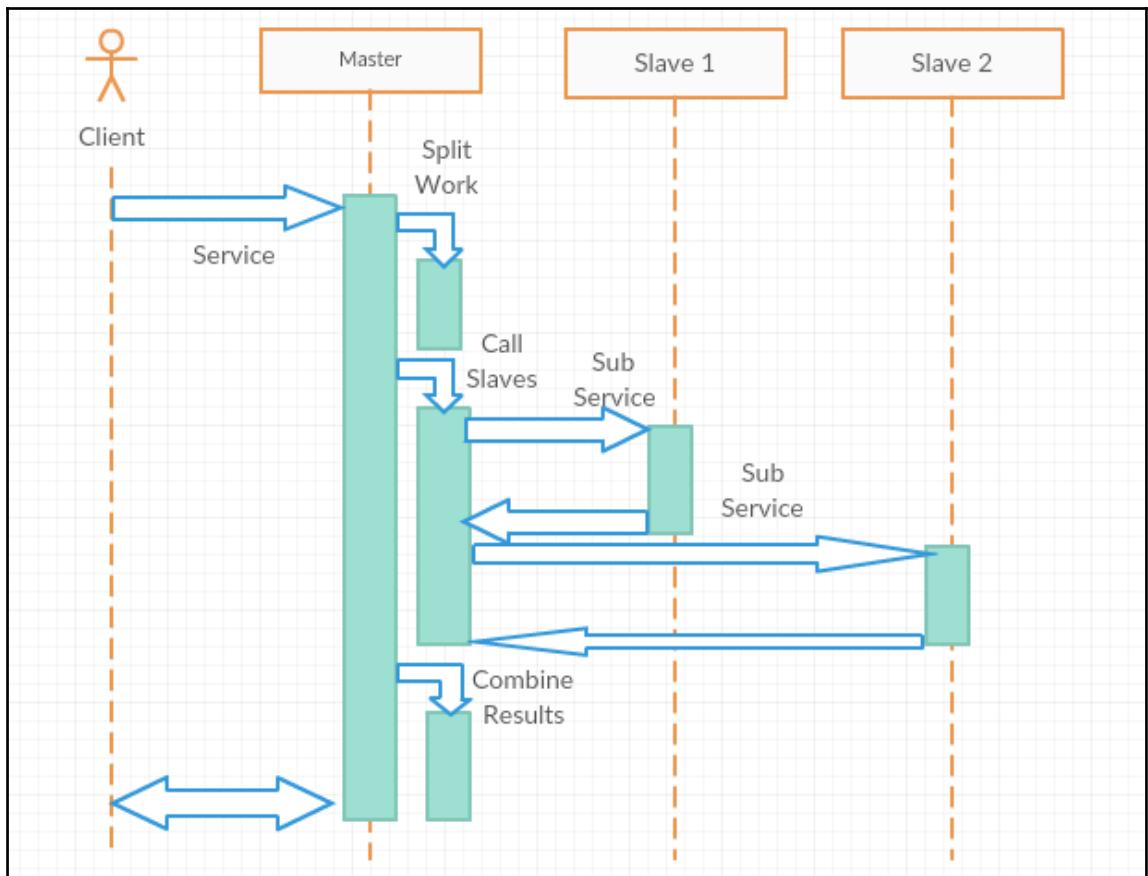
Chapter 2 - Client/Server Multi-Tier Architectural Patterns

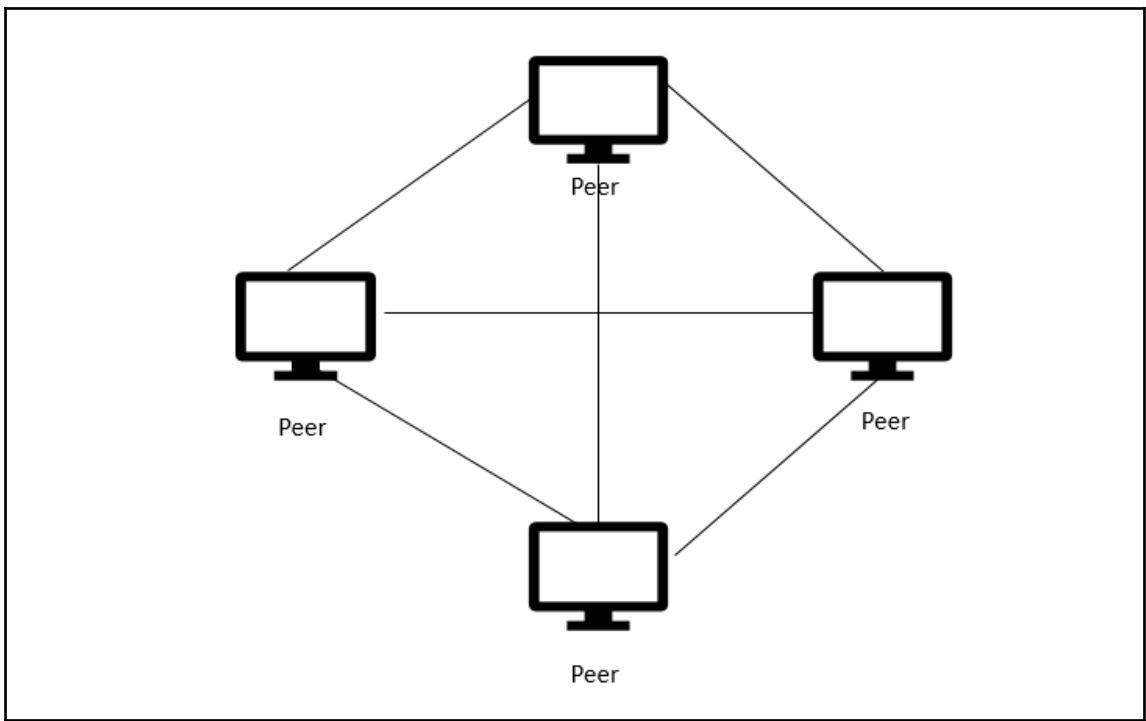


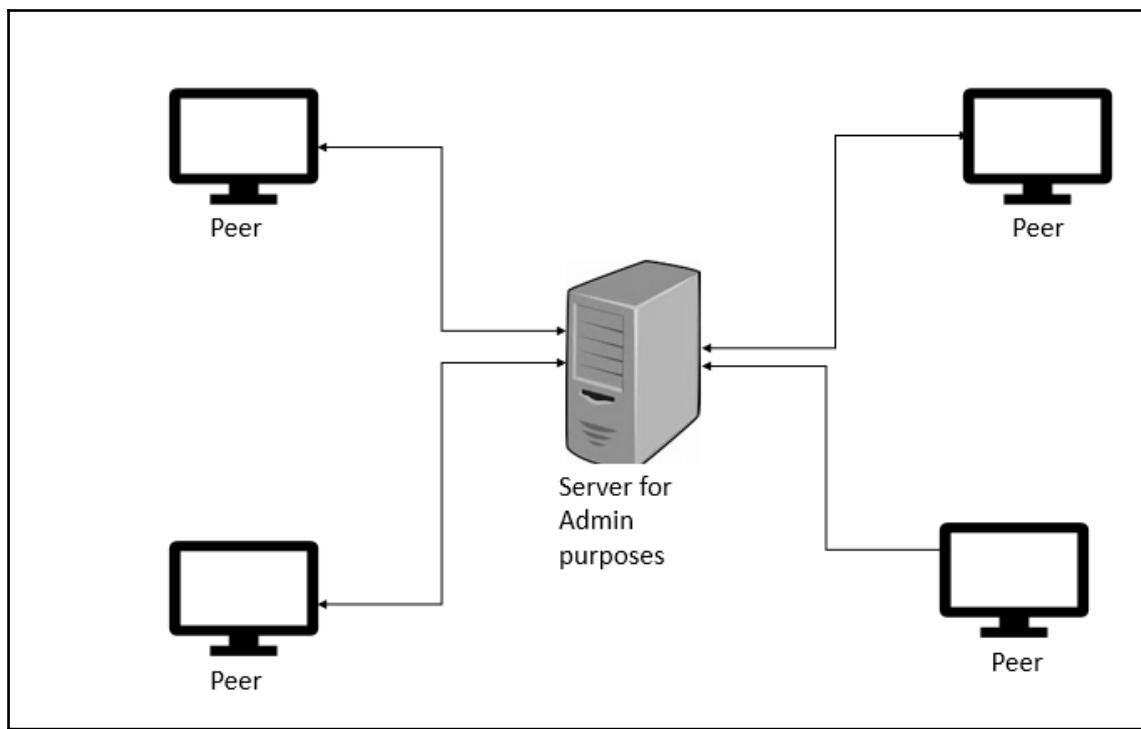


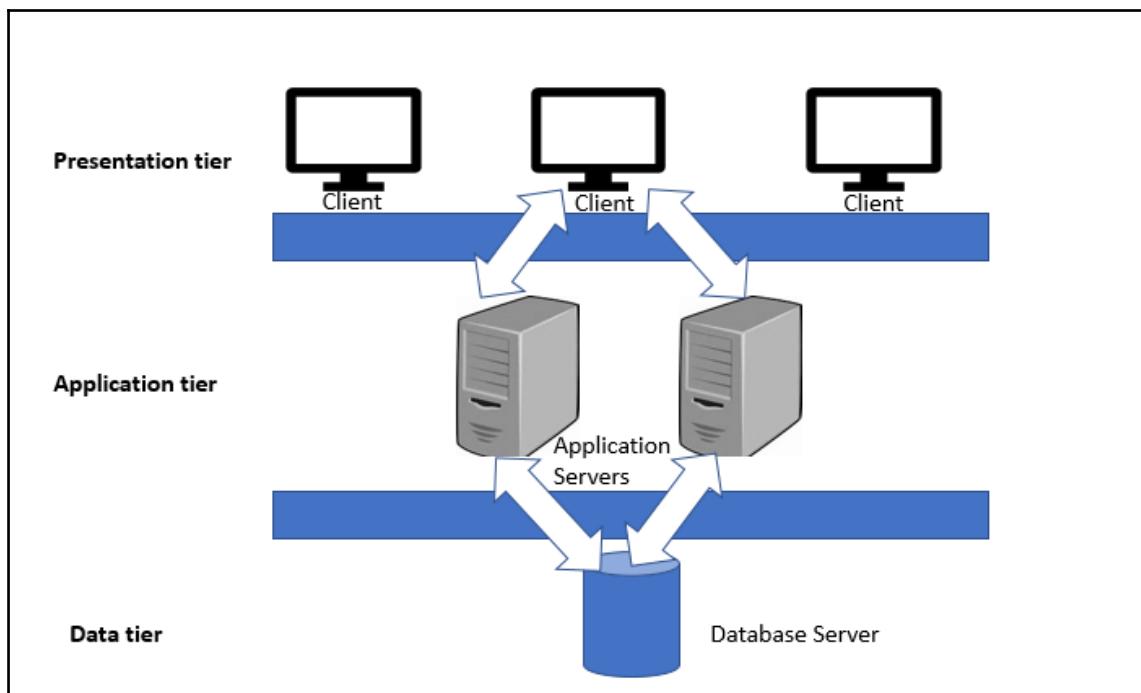


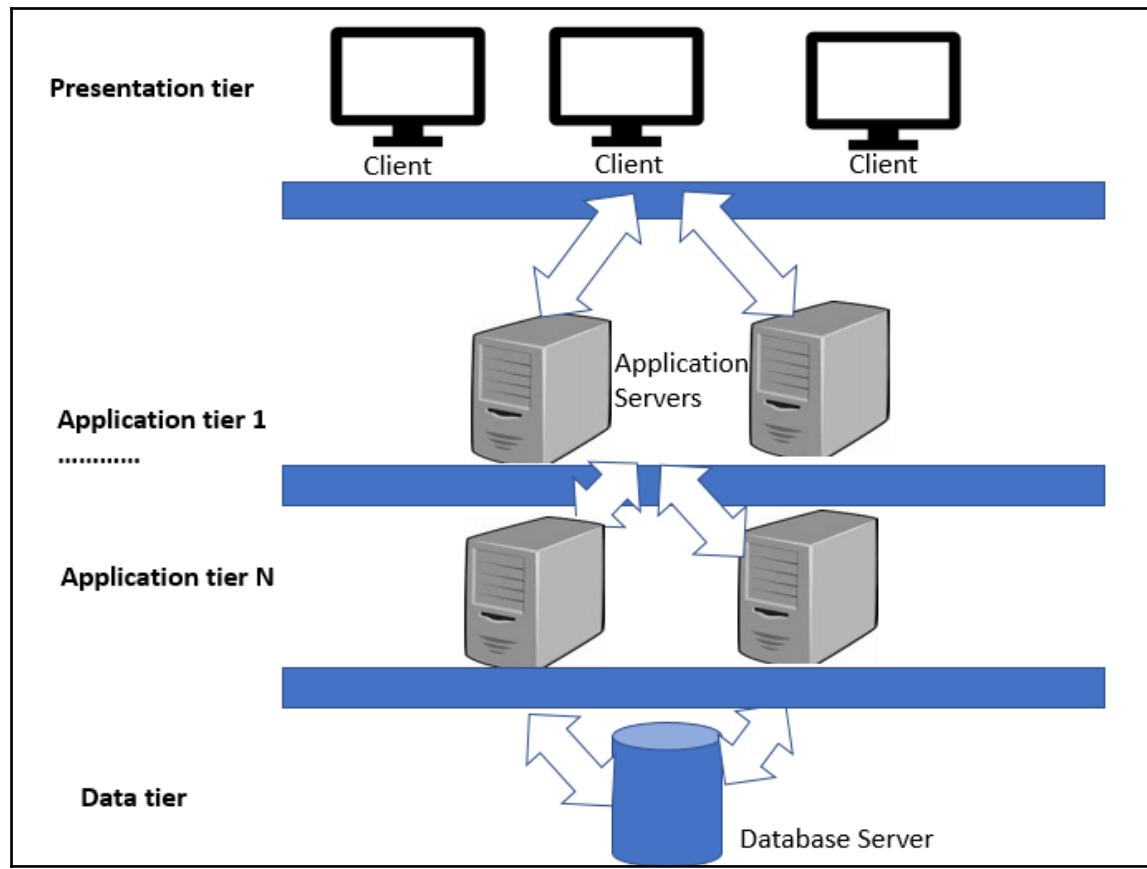


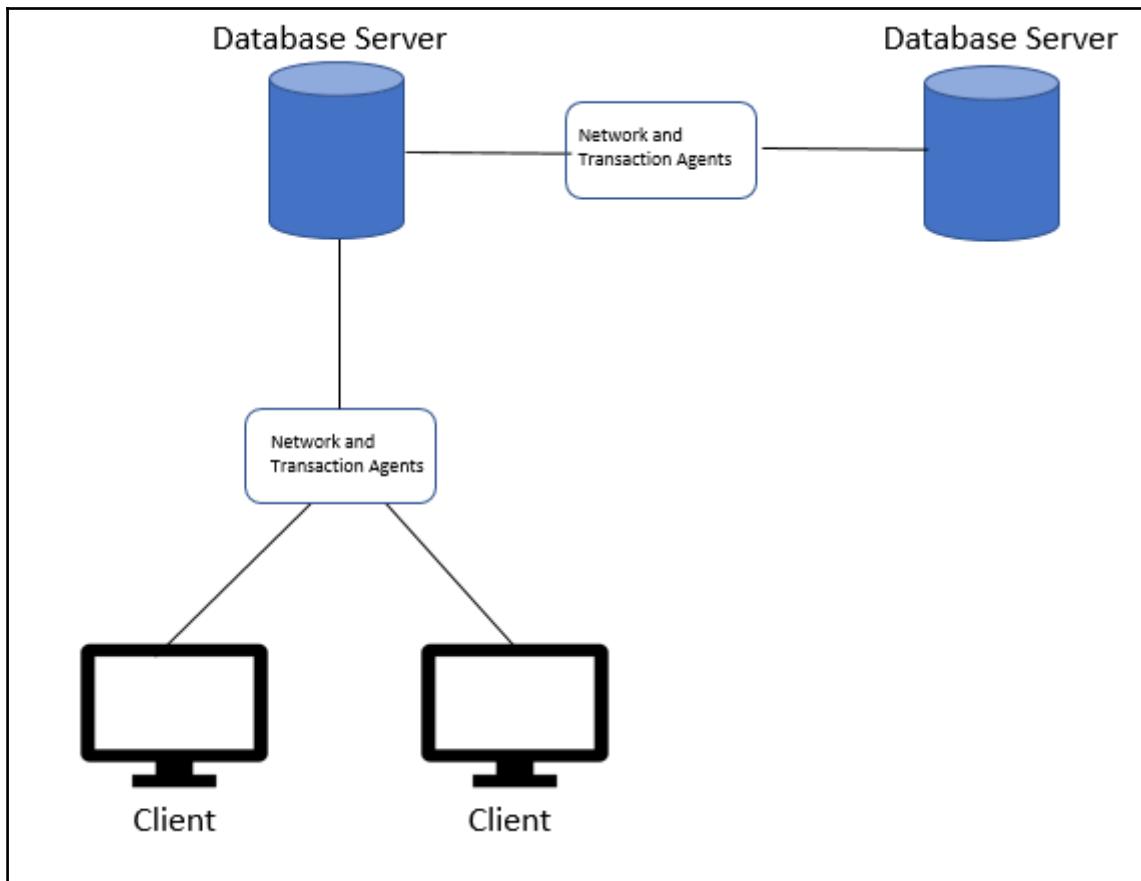


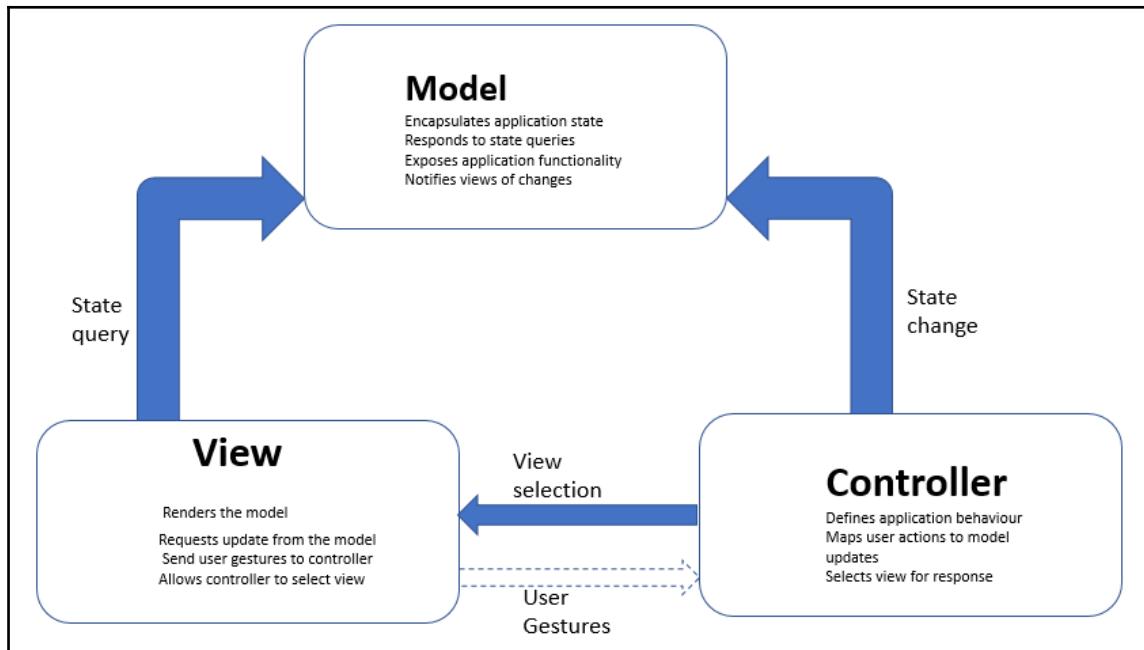
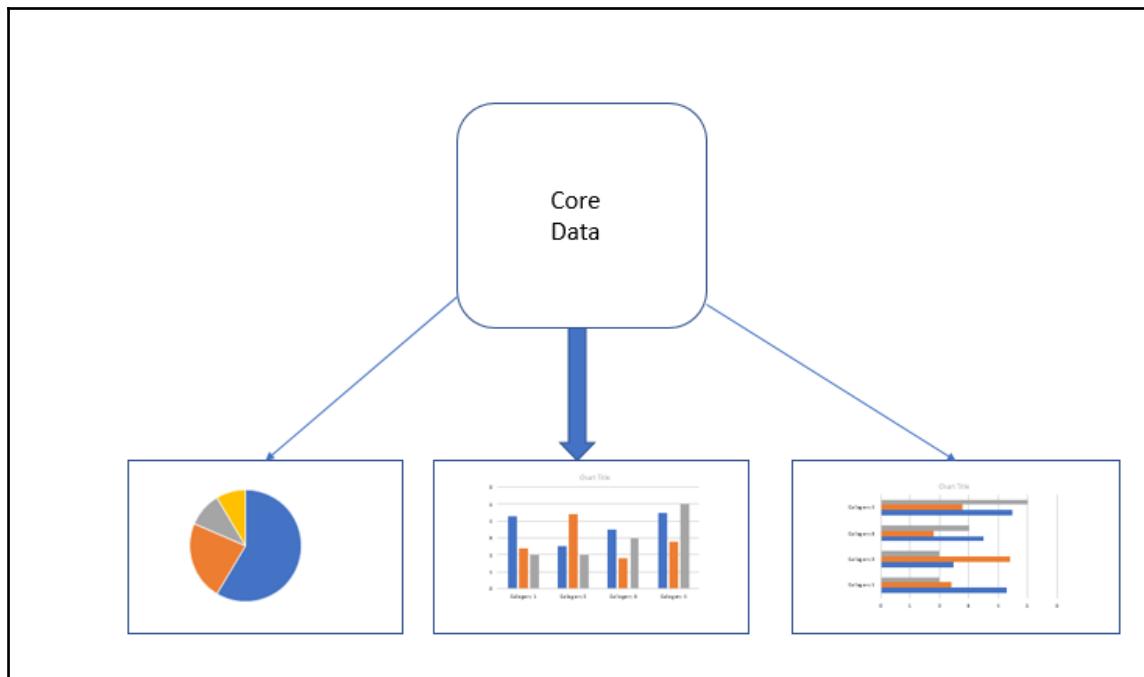


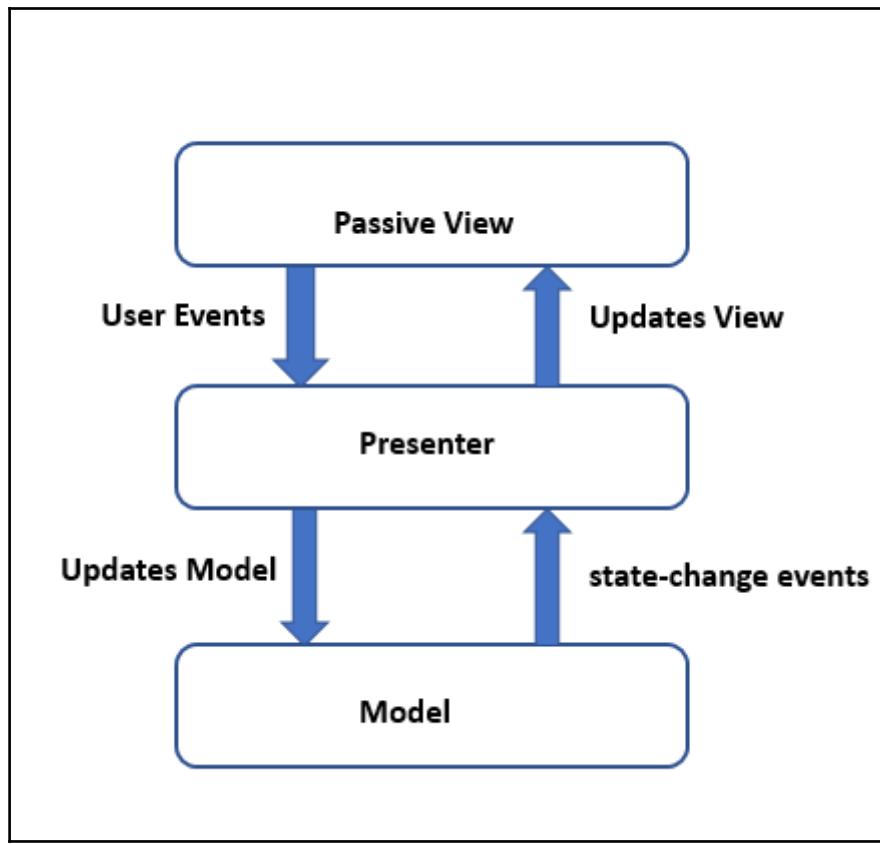


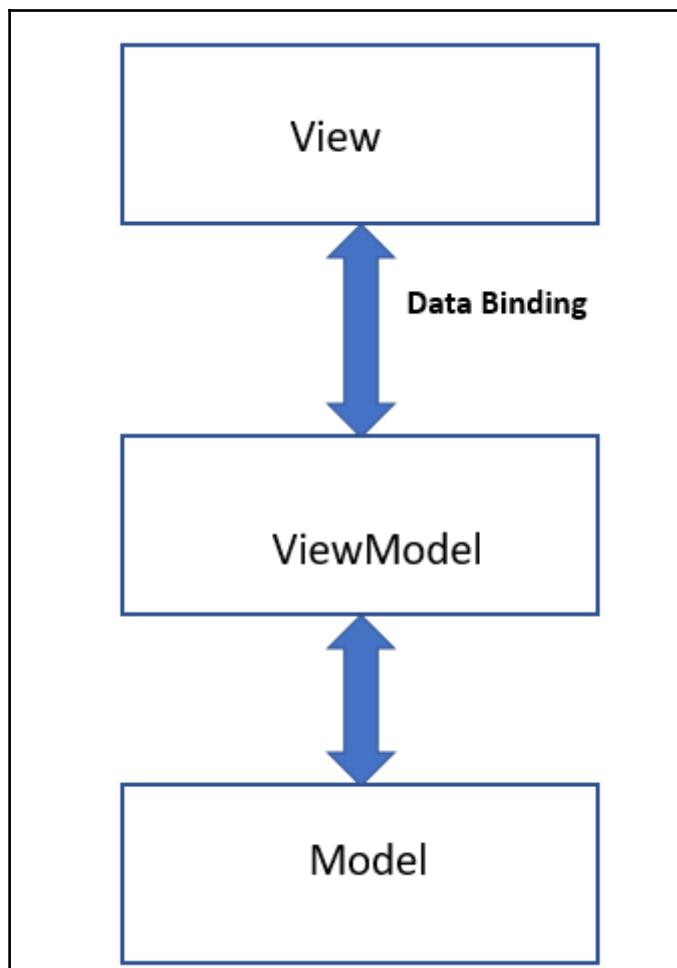


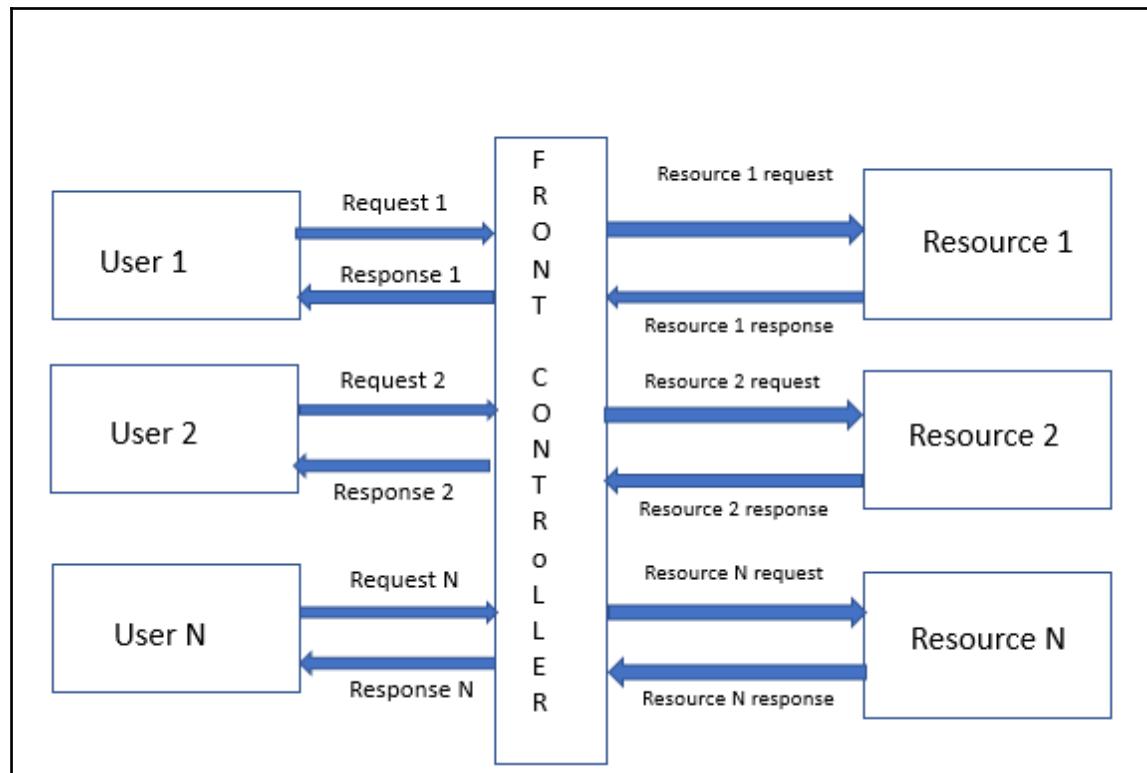


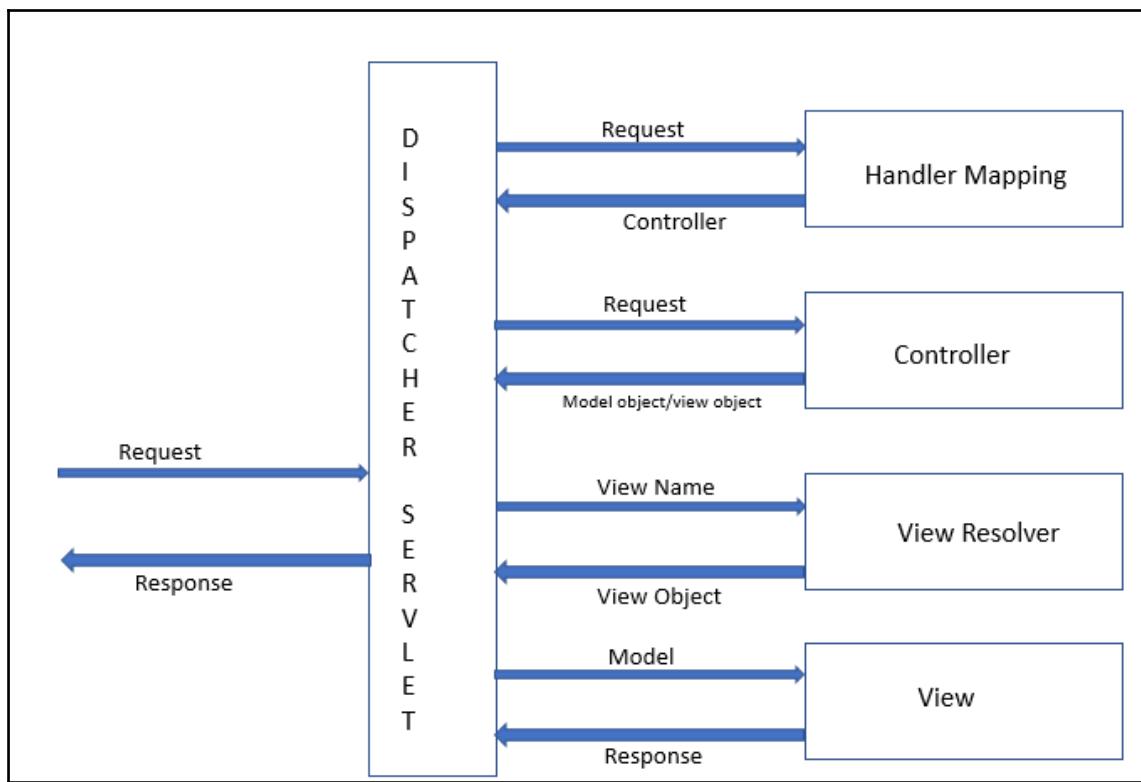




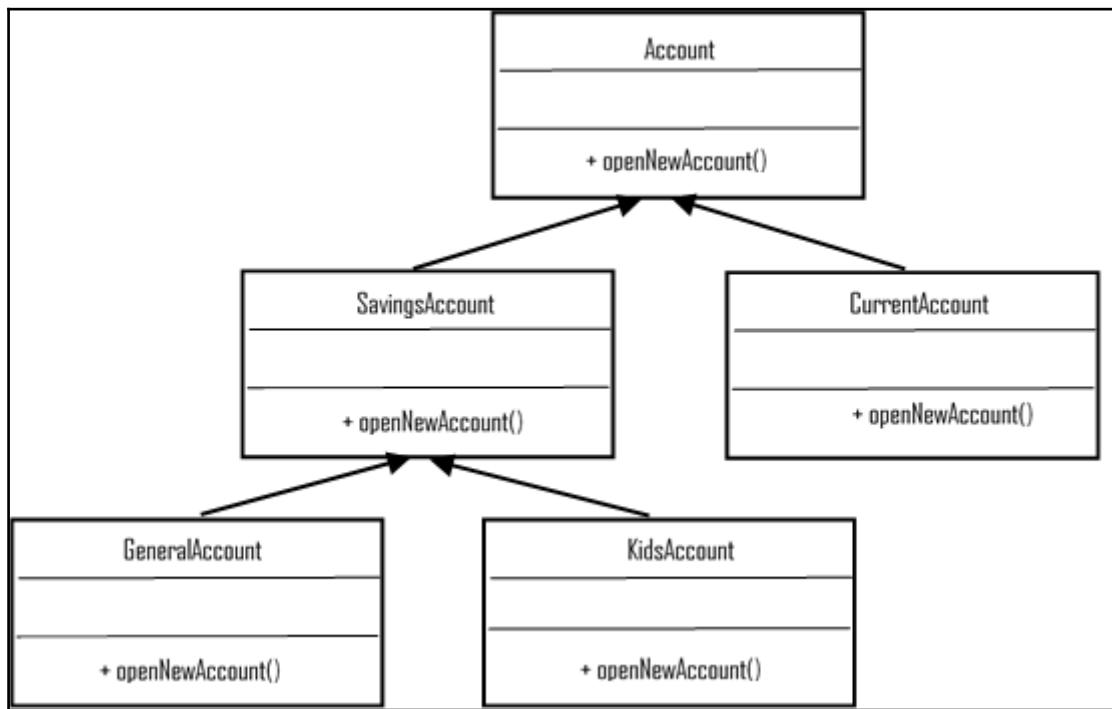
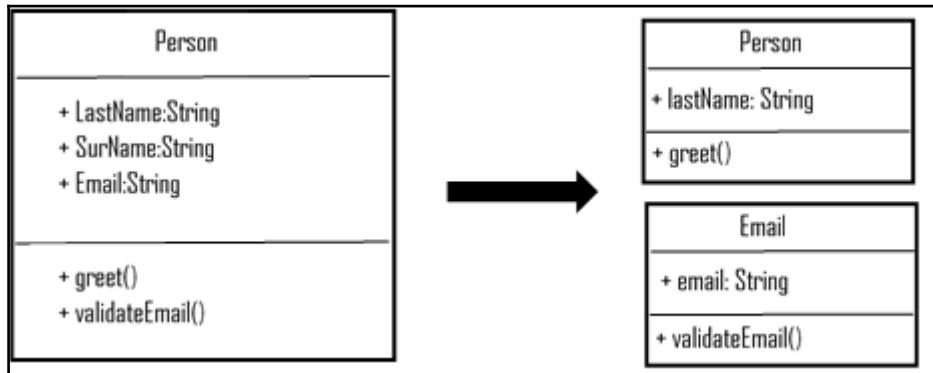


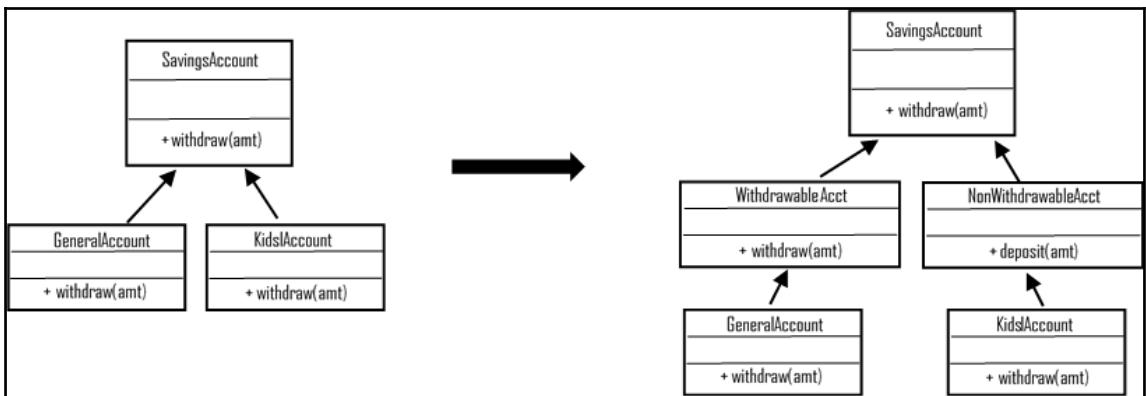






Chapter 3 - Object-Oriented Software Engineering Patterns

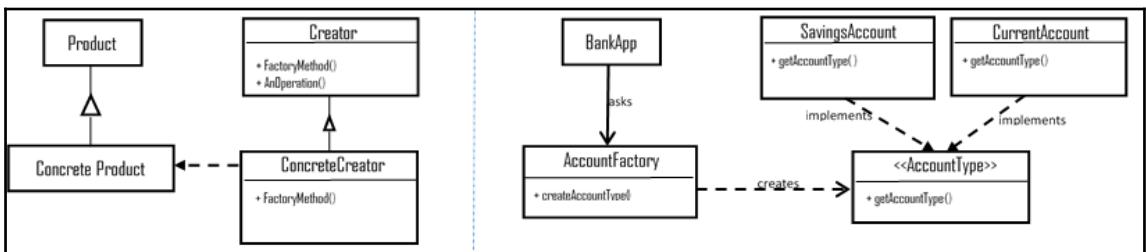
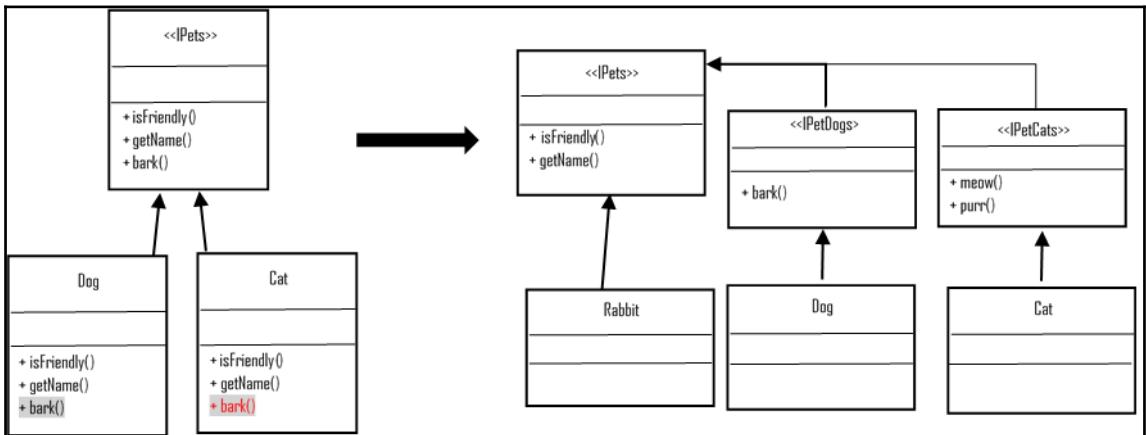


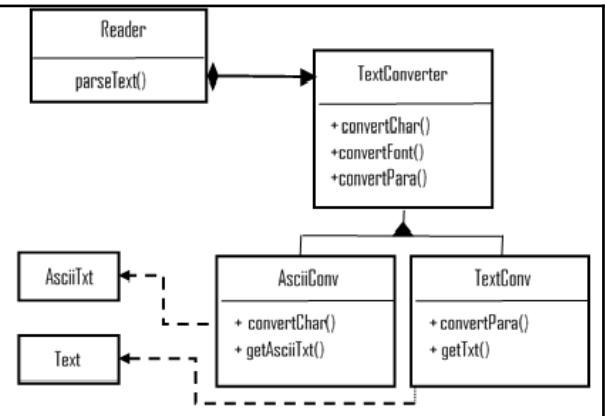
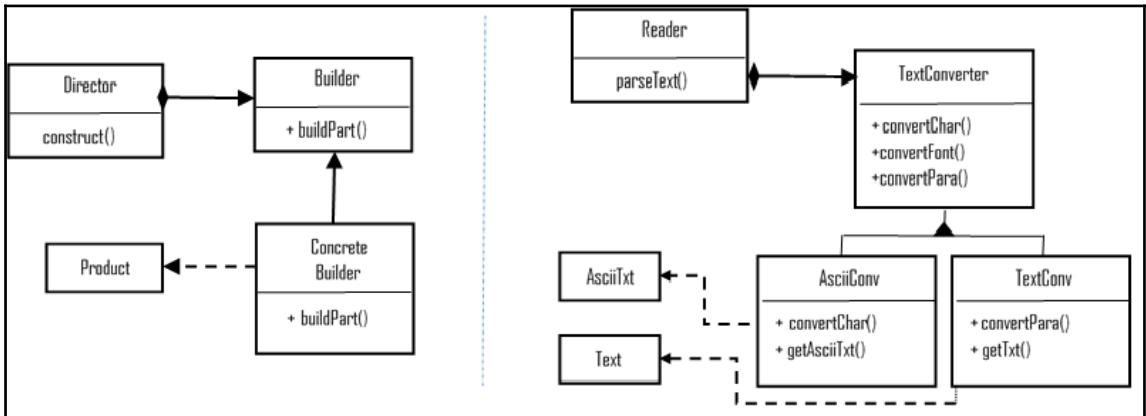
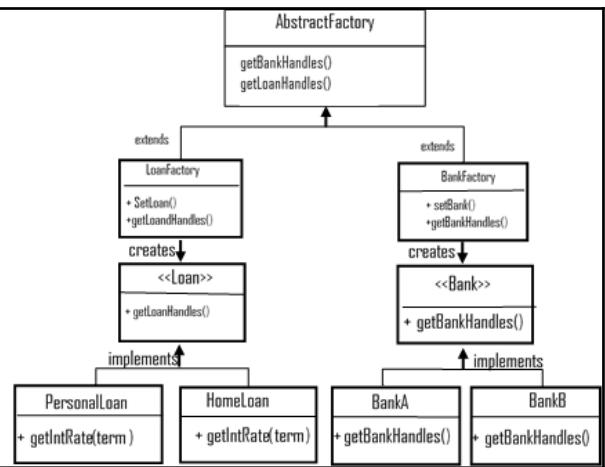
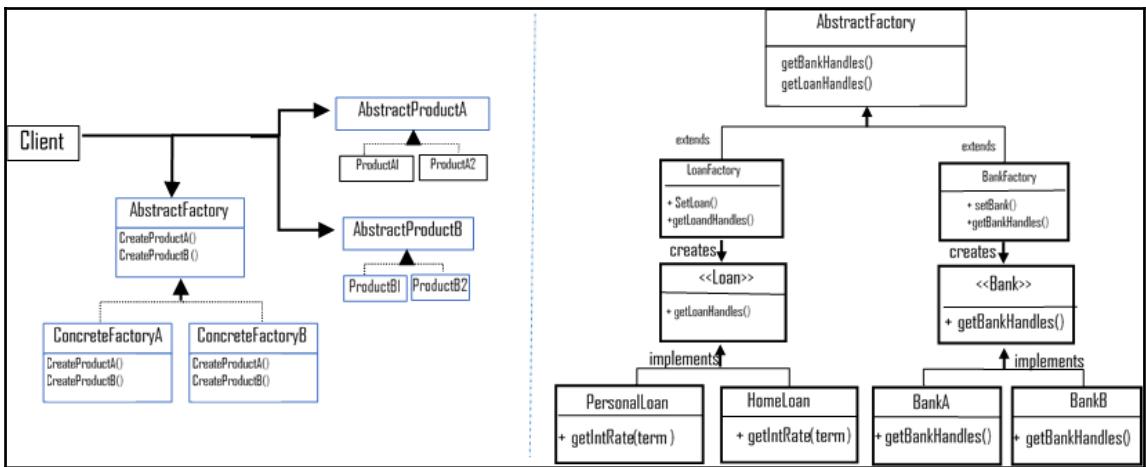


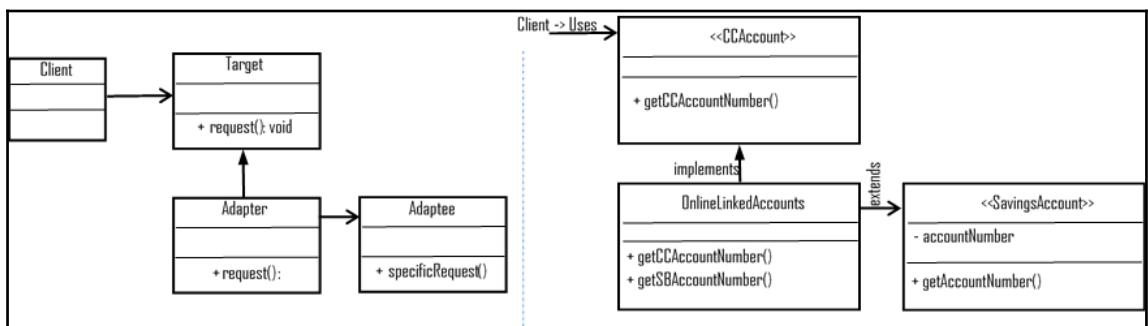
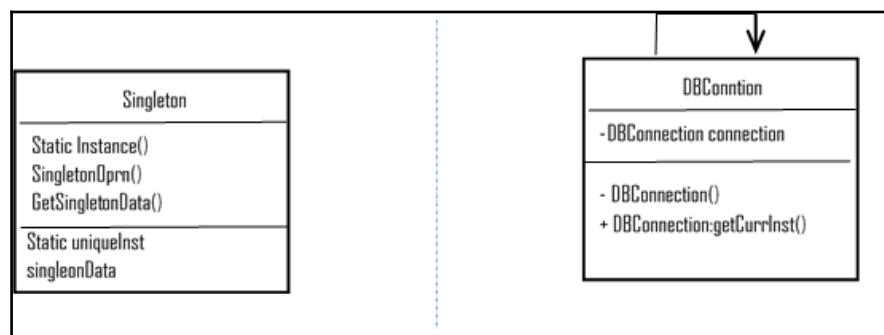
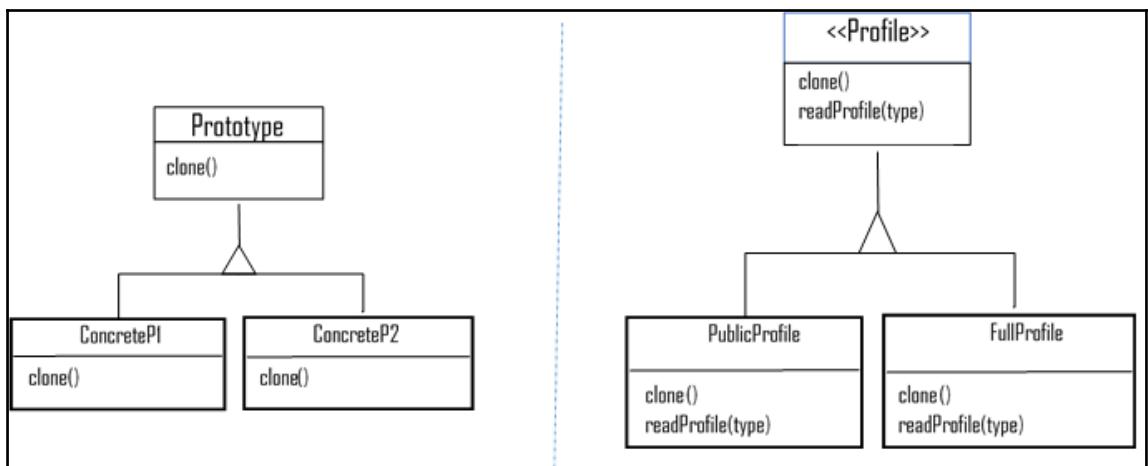
```
//works ok
Account.withdraws(new GeneralAccount());
// throws runtime exception i.e. changes the base class behavior
Account.withdraw(new KidsAccount());
```

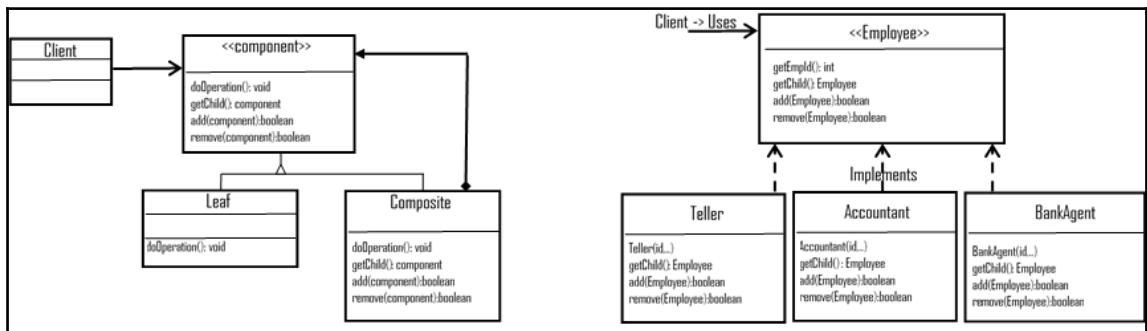
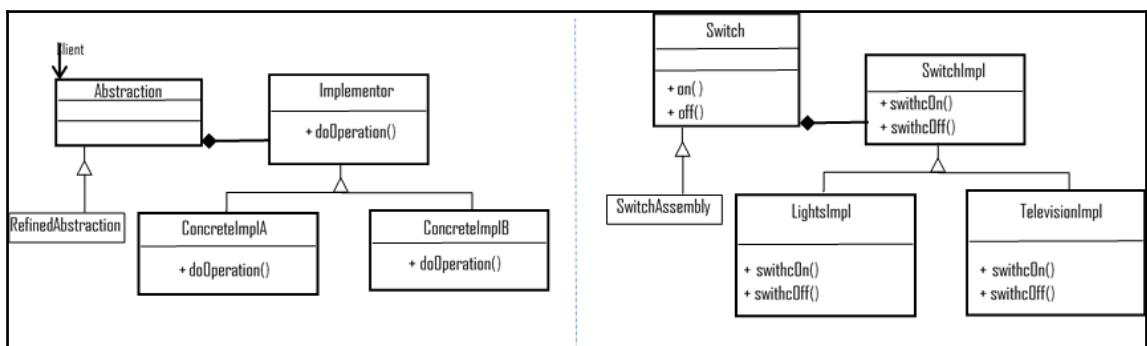
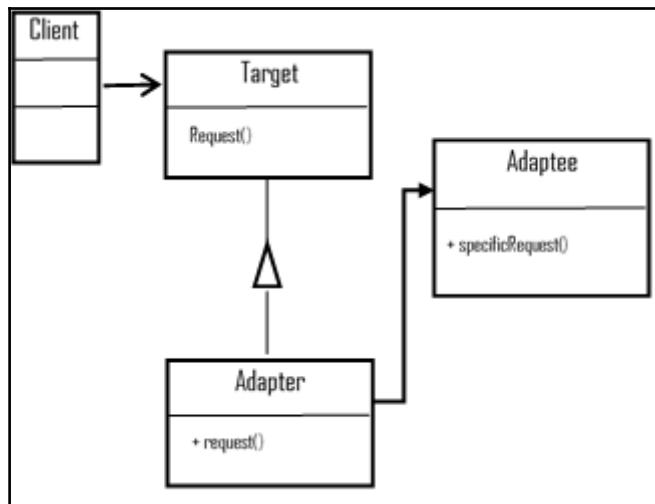
→

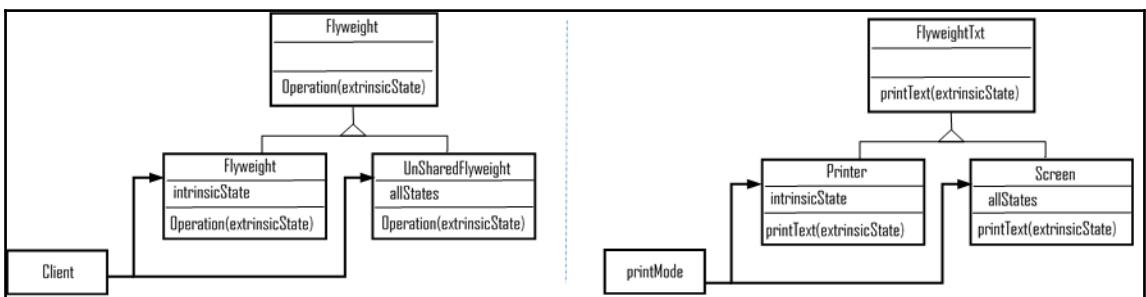
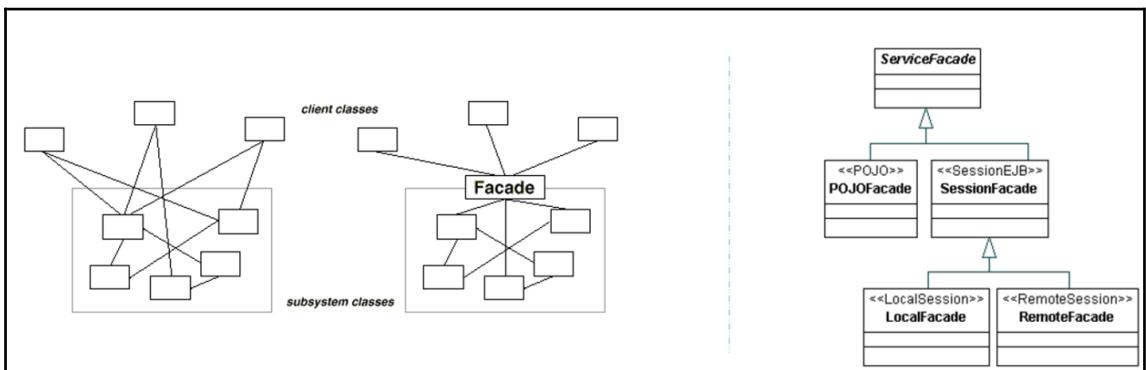
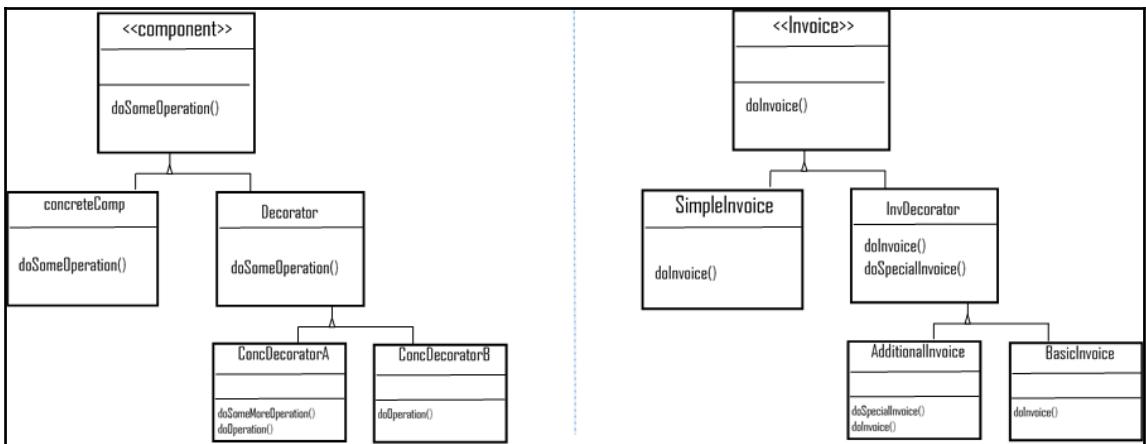
```
//works ok
Account.withdraw(new GeneralAccount());
// gives compilation error as it should
Account.withdraw(new KidsAccount());
```

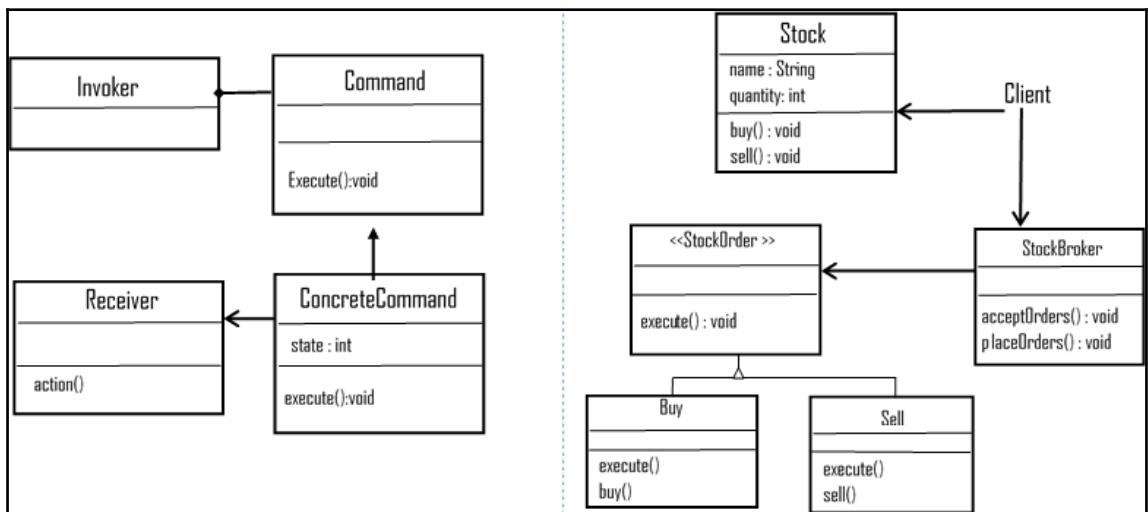
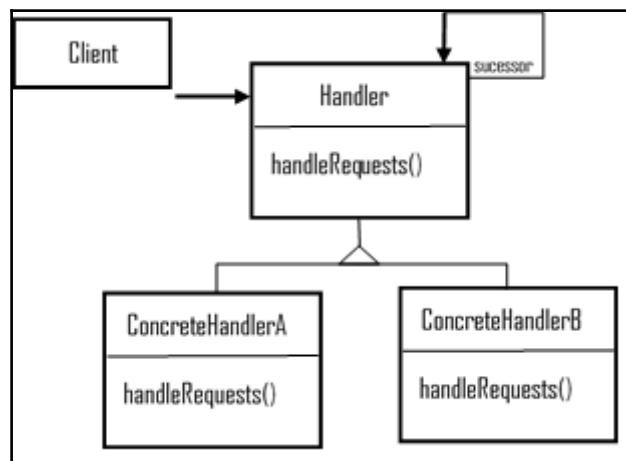
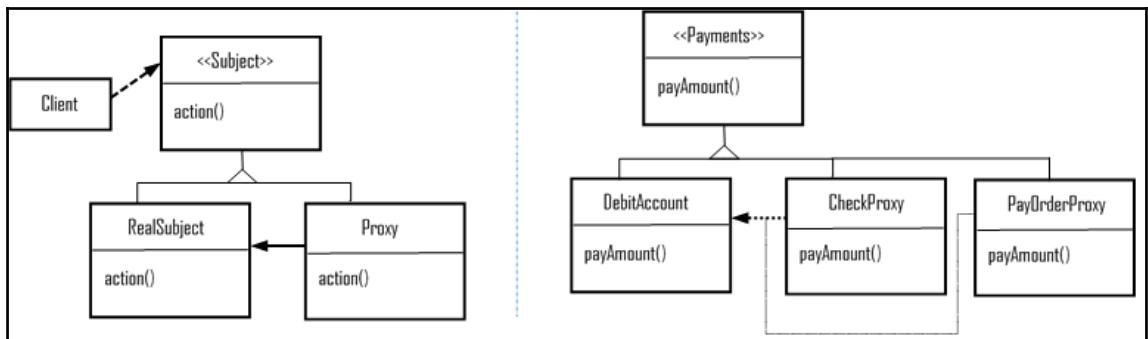


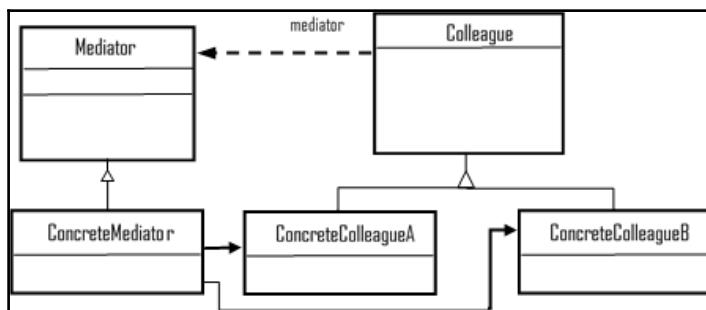
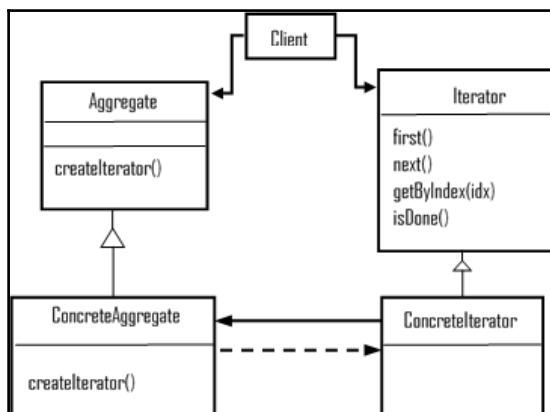
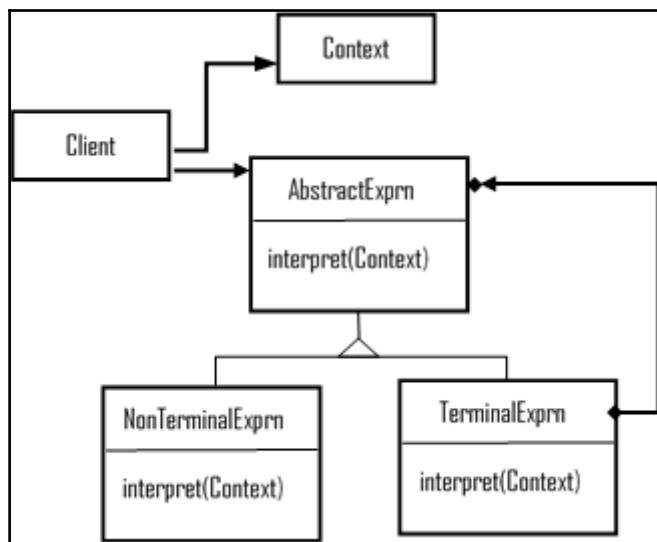


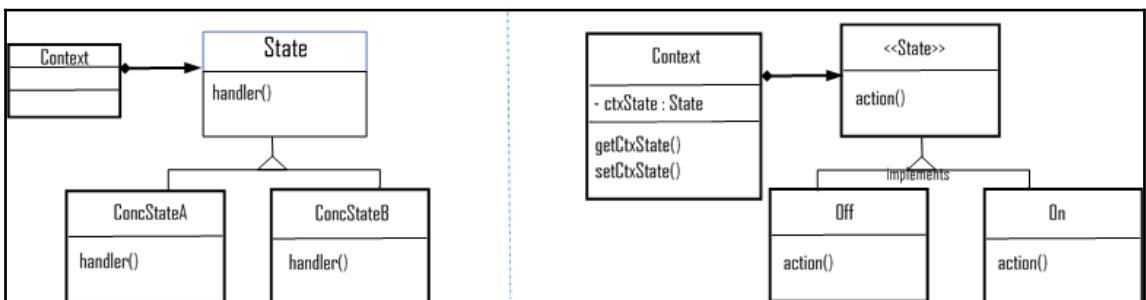
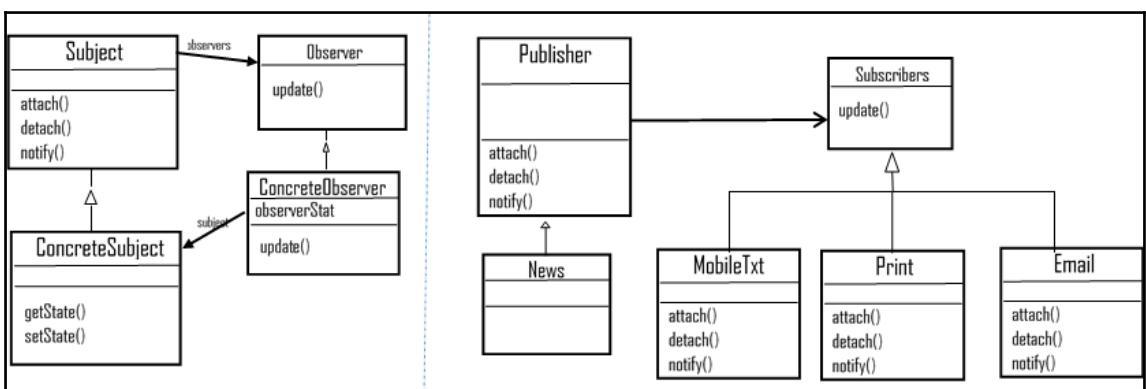
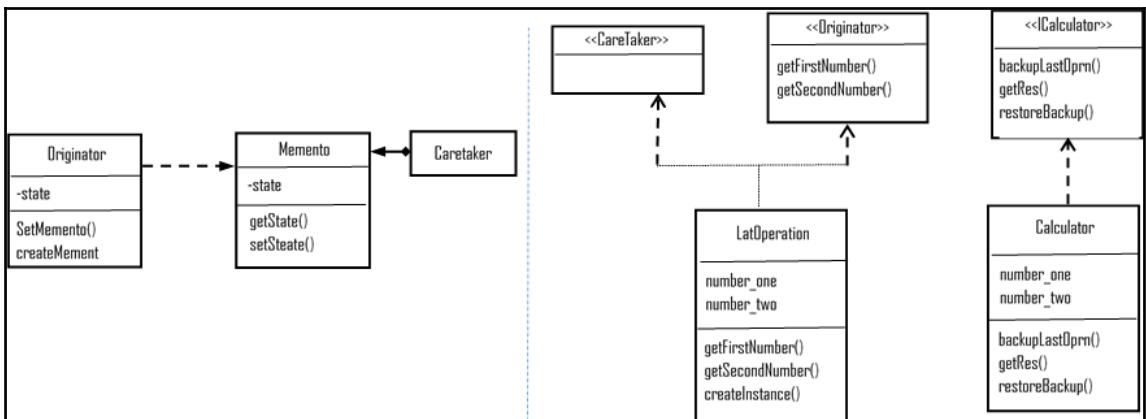


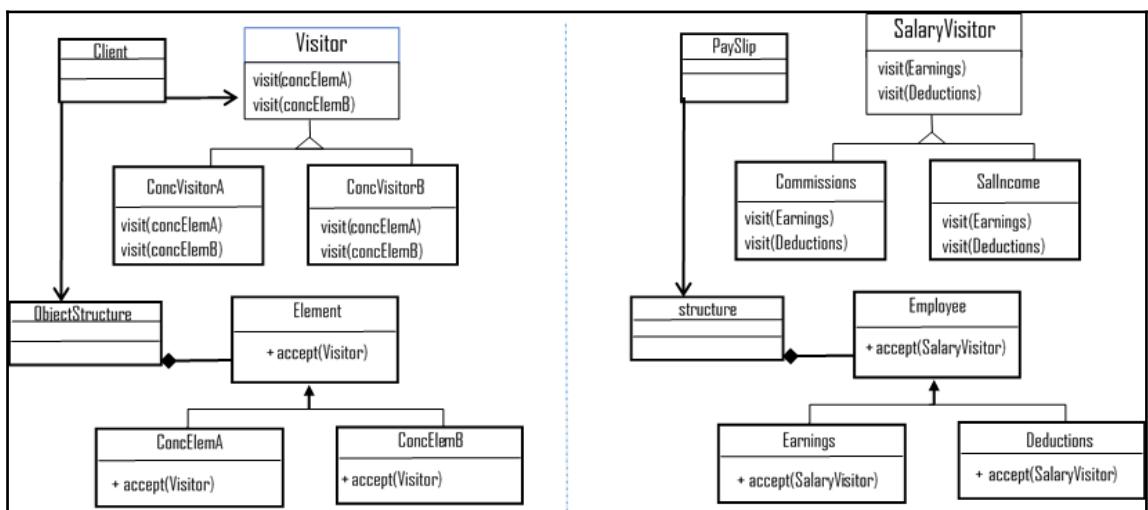
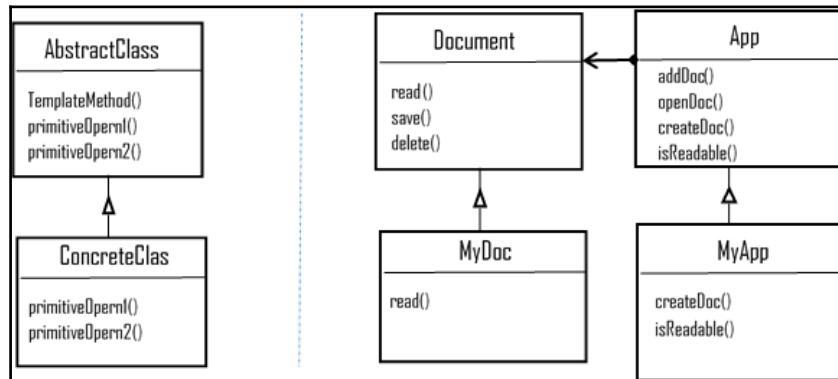
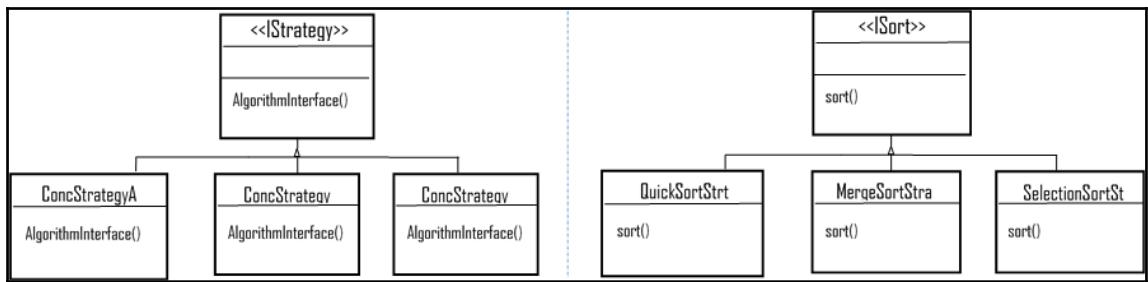






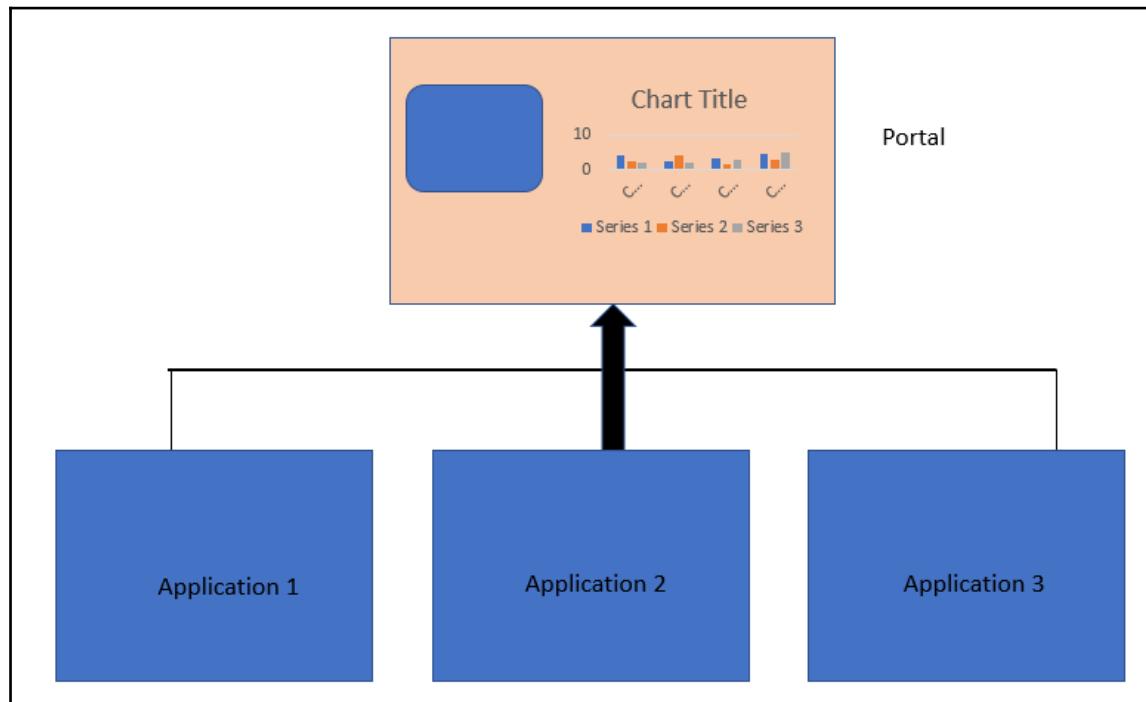


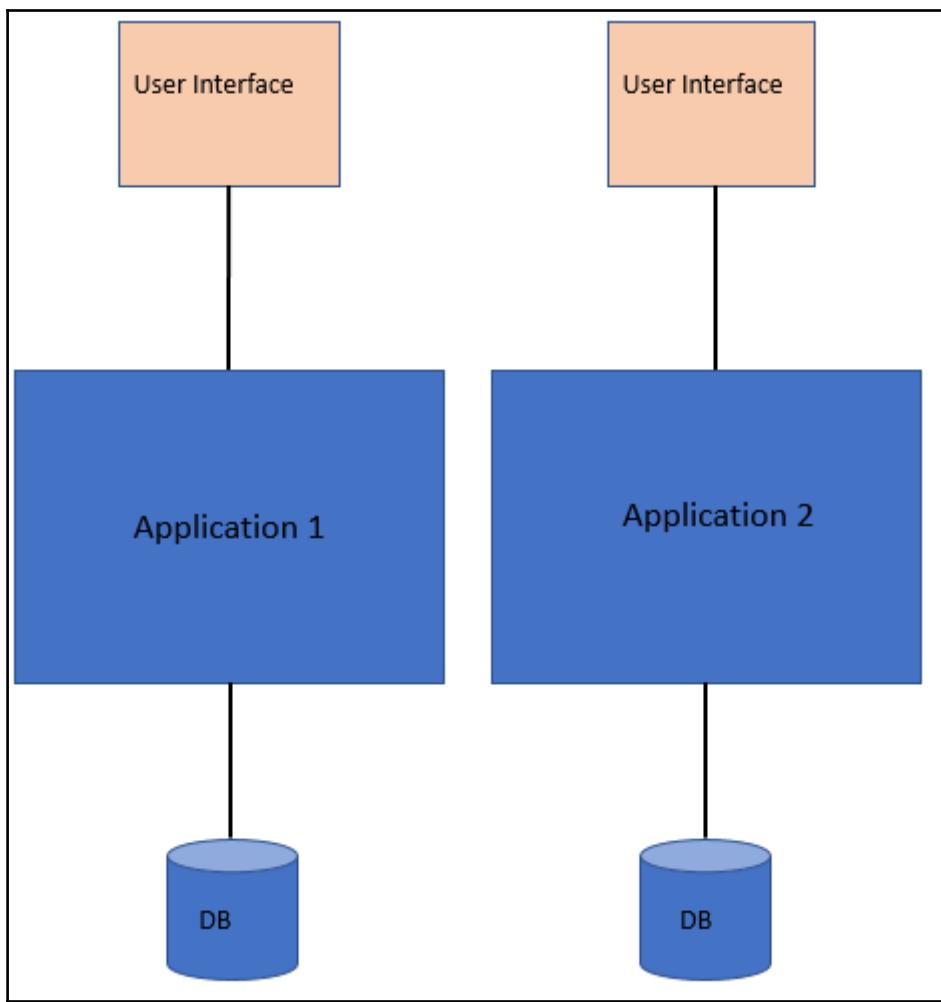


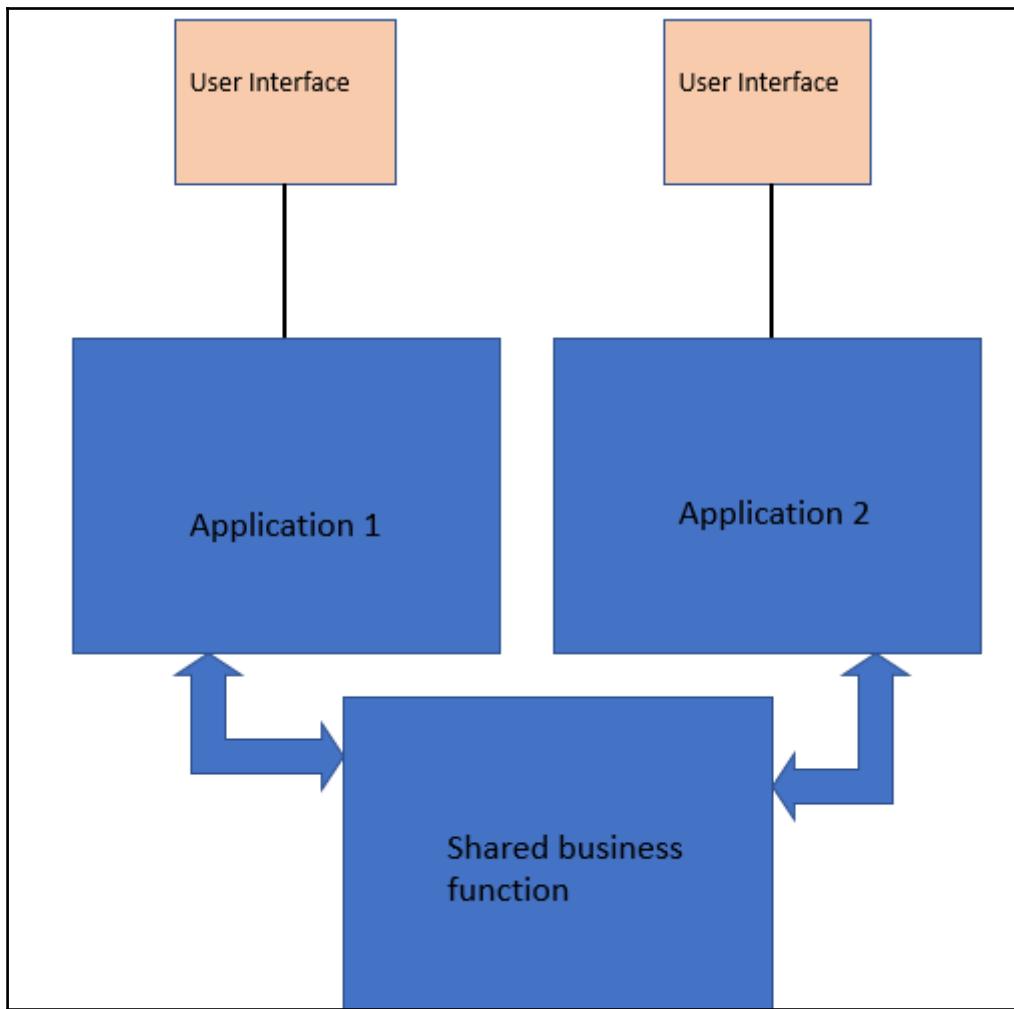


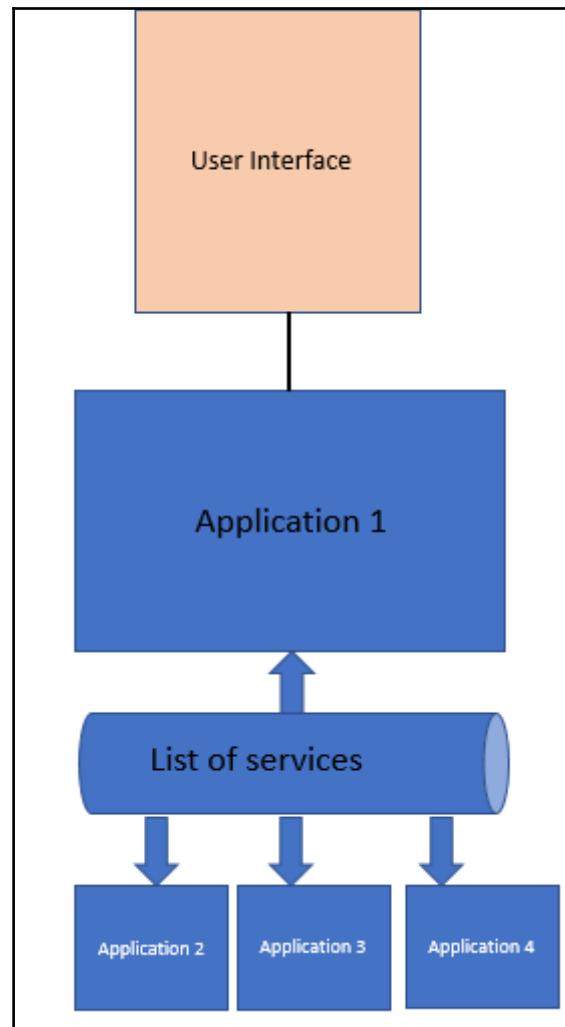
		Scope	
		Class	Object
Purpose / Type	Creational	(1) Factory Method	1. Abstract Factory 2. Builder 3. Prototype 4. Singleton
	Structural	(2) Adapter (class)	5. Adapter (Object) 6. Bridge 7. Composite 8. Decorator 9. Façade 10. Flyweight 11. Proxy
	Behavioral	(3) Interpreter (4) Template Method	12. Chain of Responsibility 13. Command 14. Iterator 15. Mediator 16. Memento 17. Observer 18. State 19. Strategy 20. Visitor

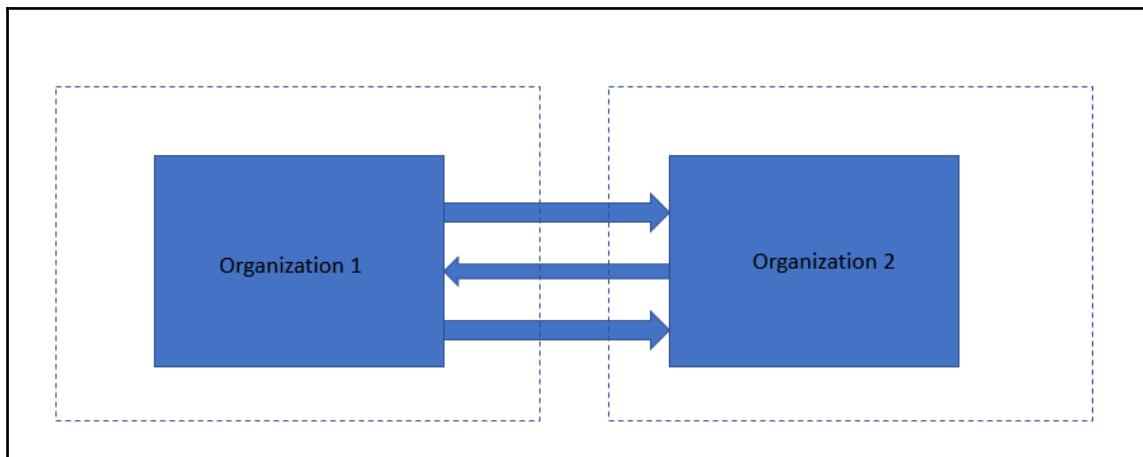
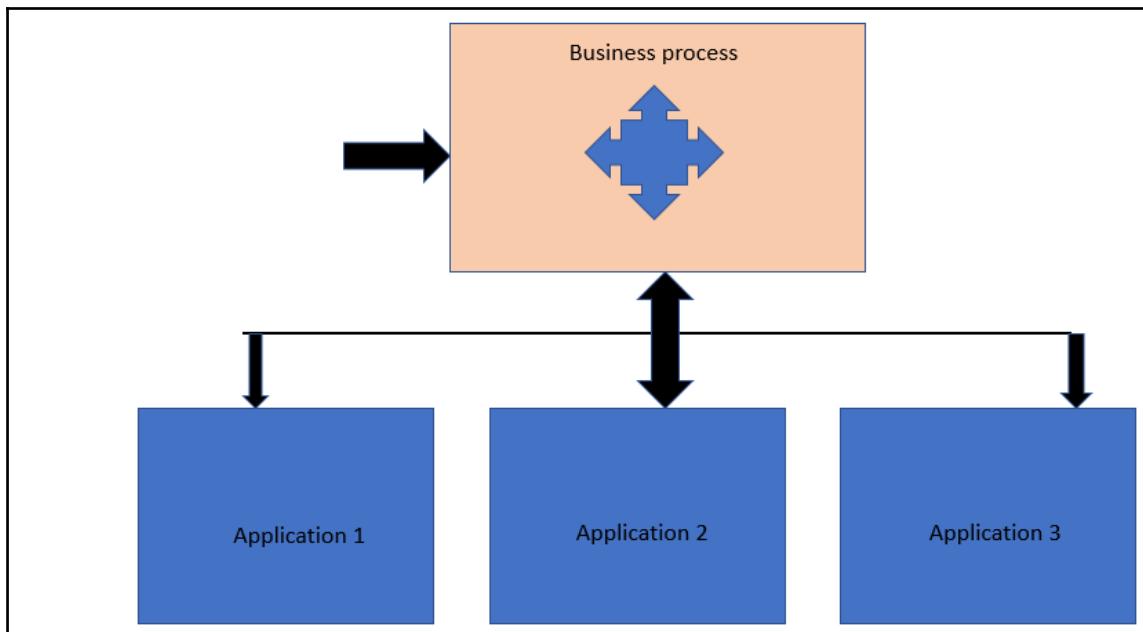
Chapter 4 - Enterprise Integration Patterns











File Transfer

Shared Database

Remote
Procedure
Invocation

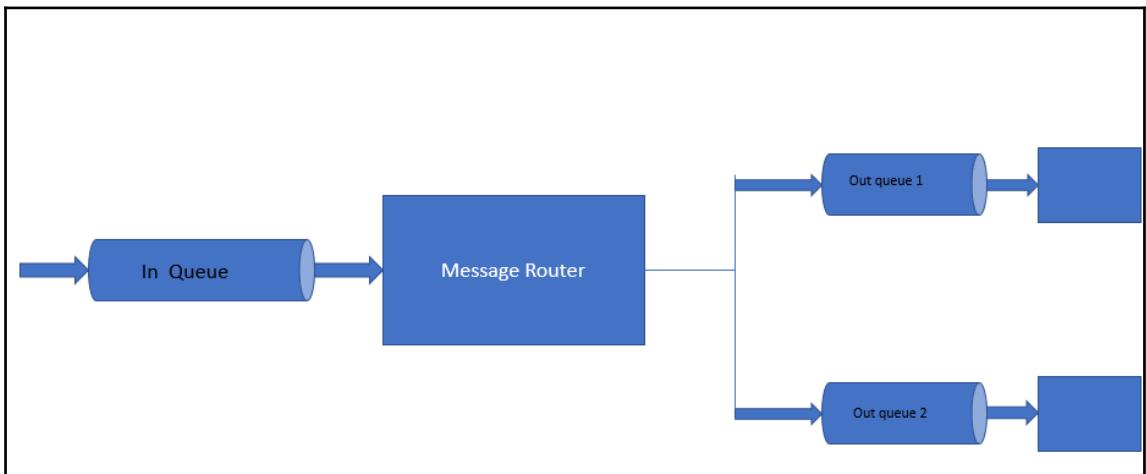
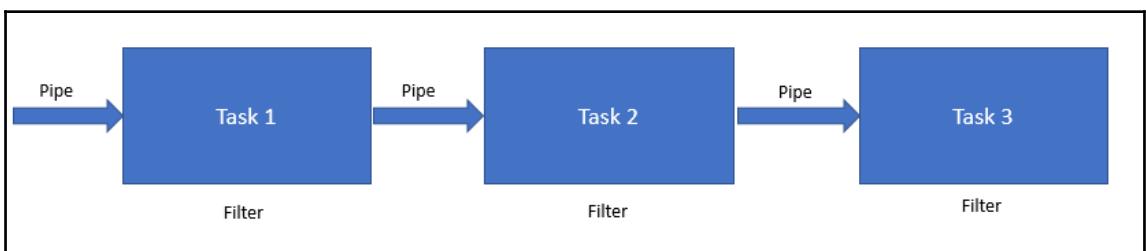
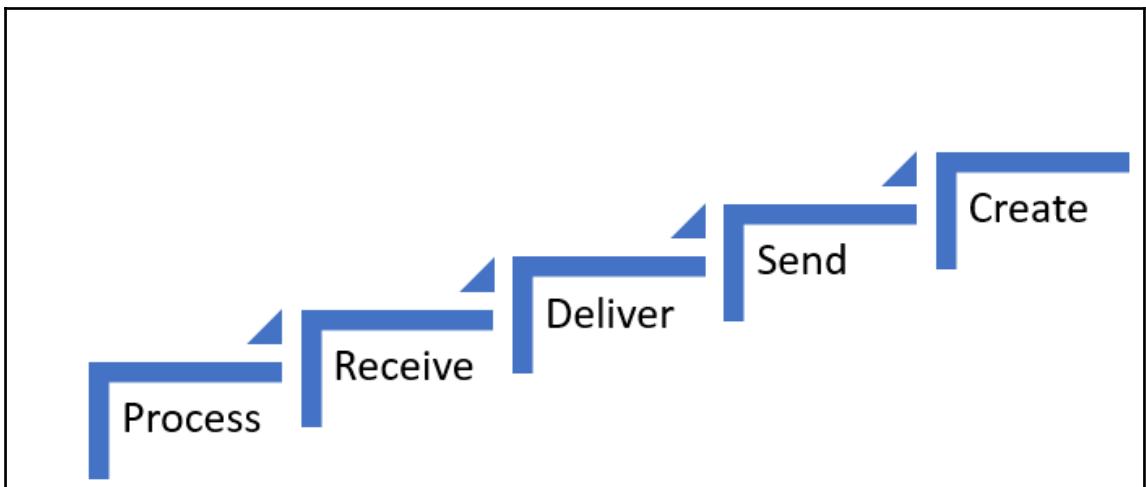
Messaging

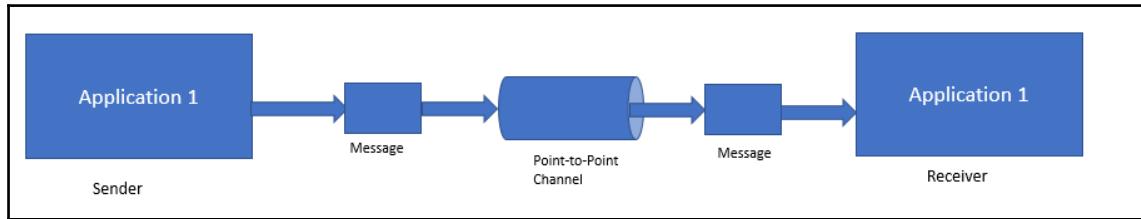
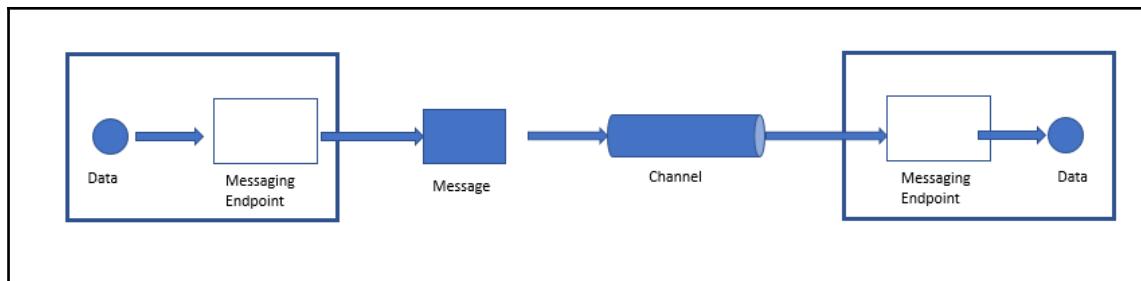
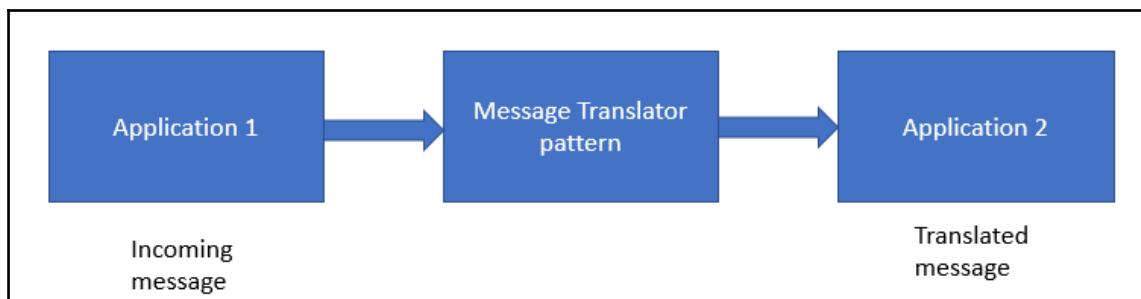
Sender

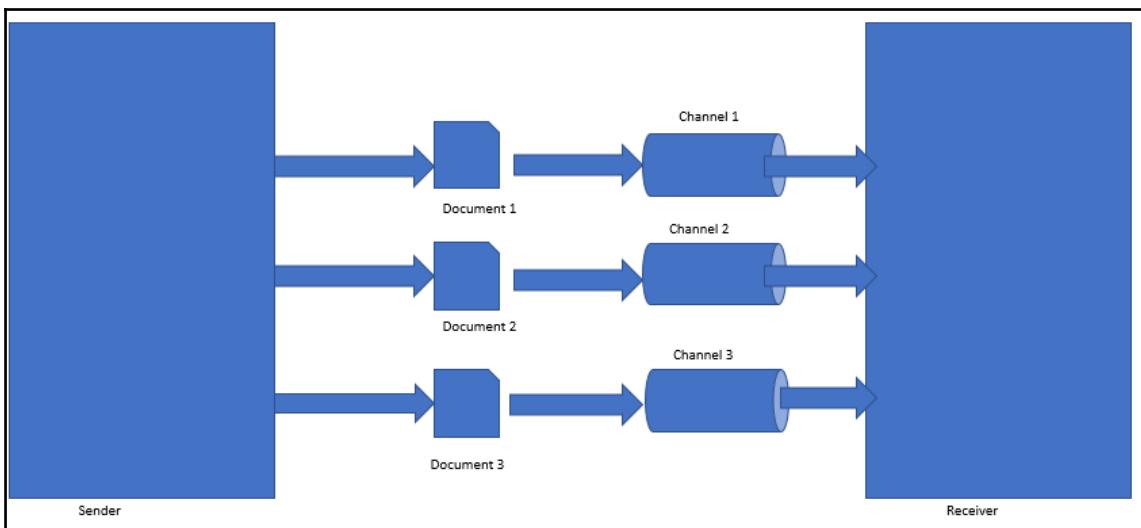
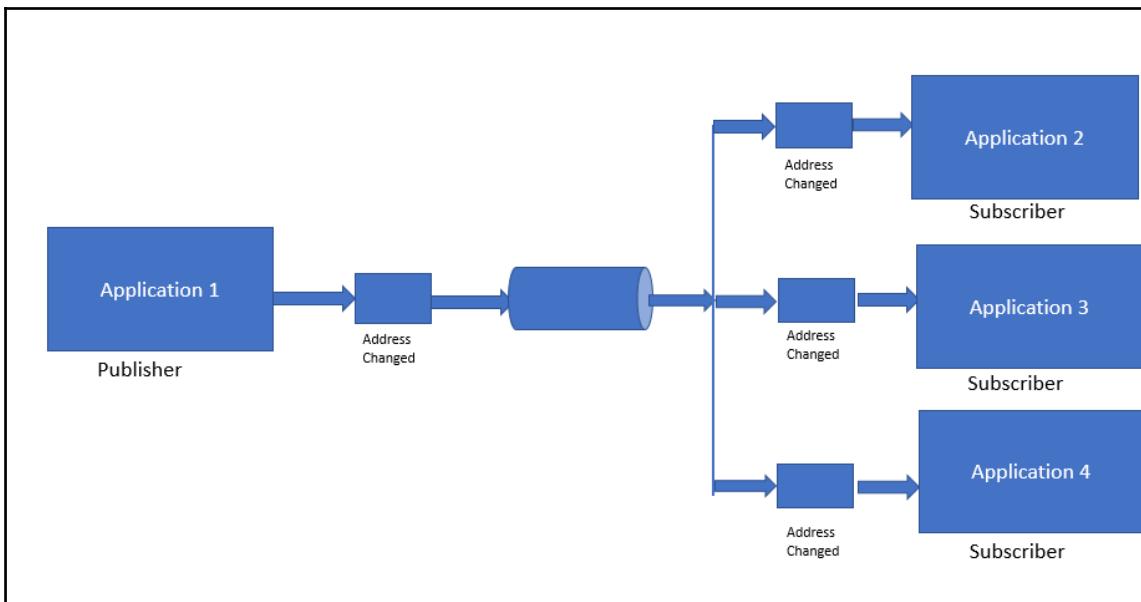
MoM

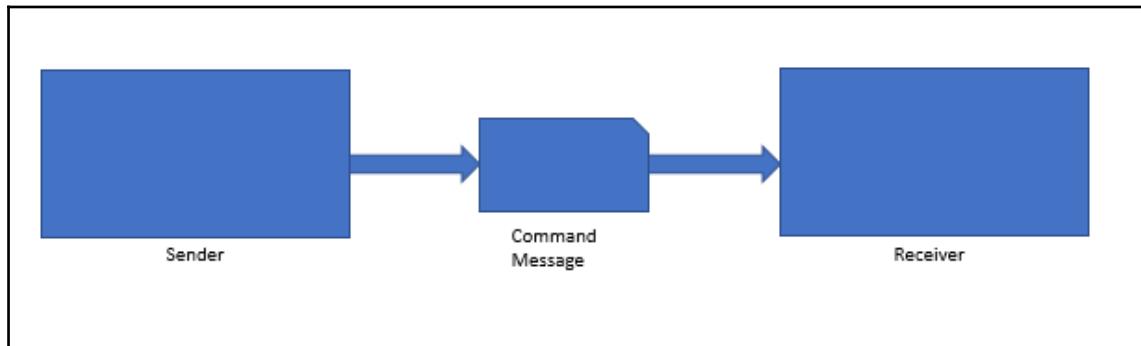
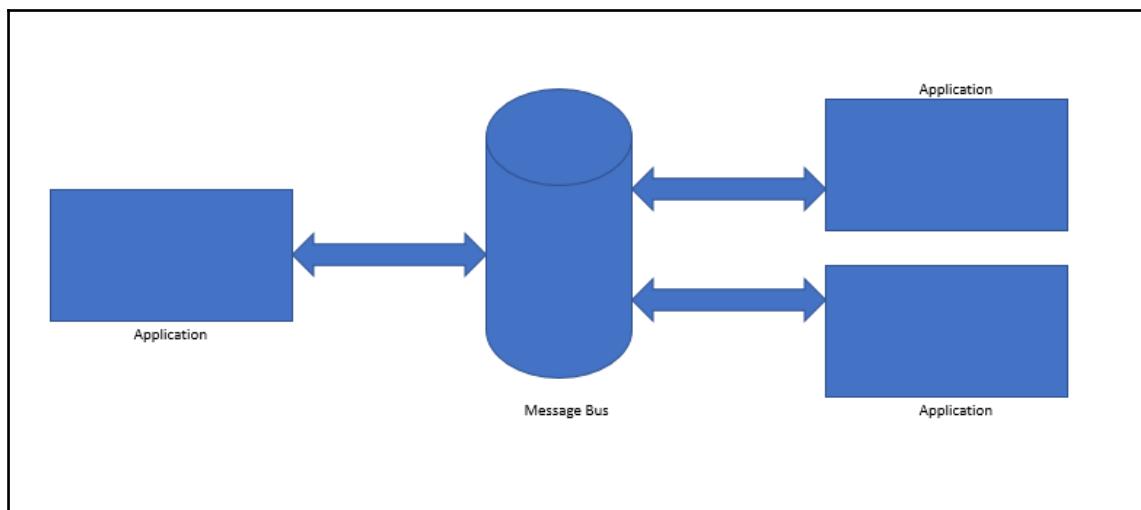
Messaging Channel

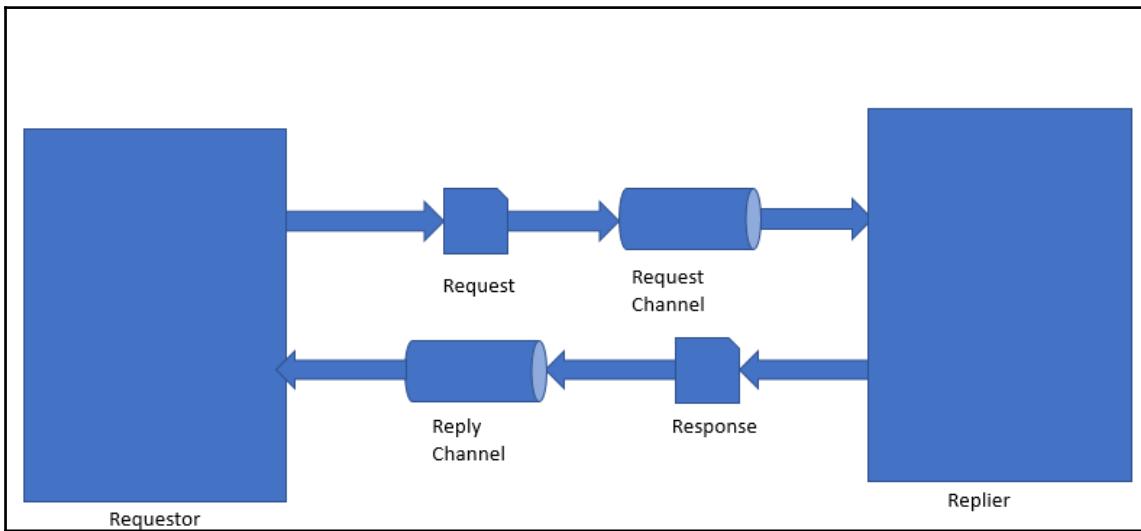
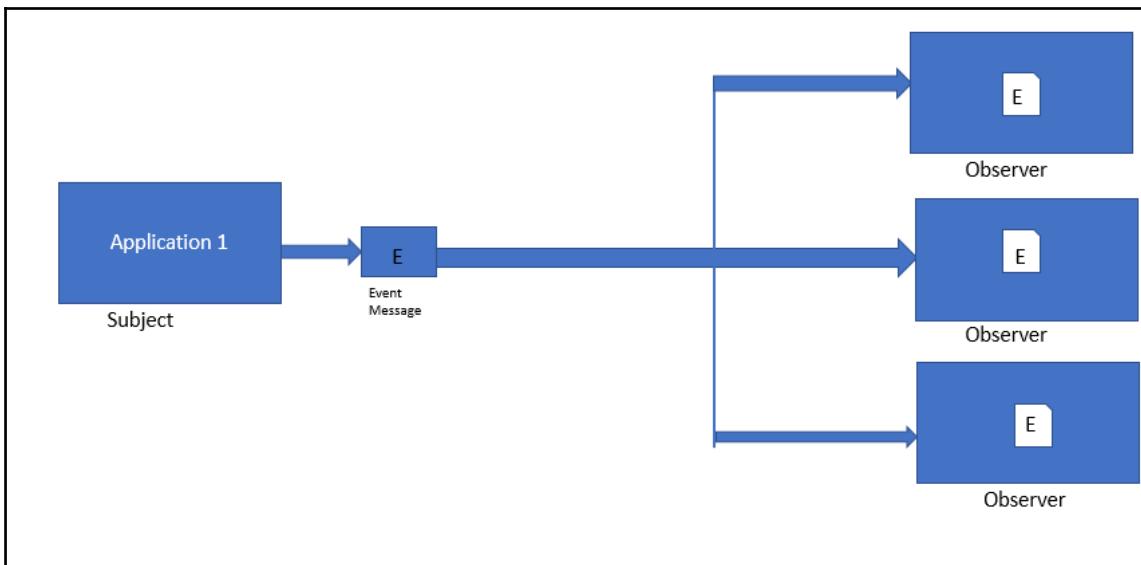
Receiver

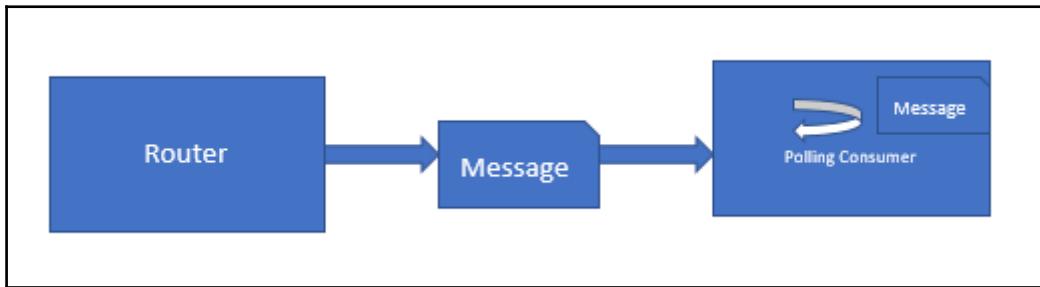
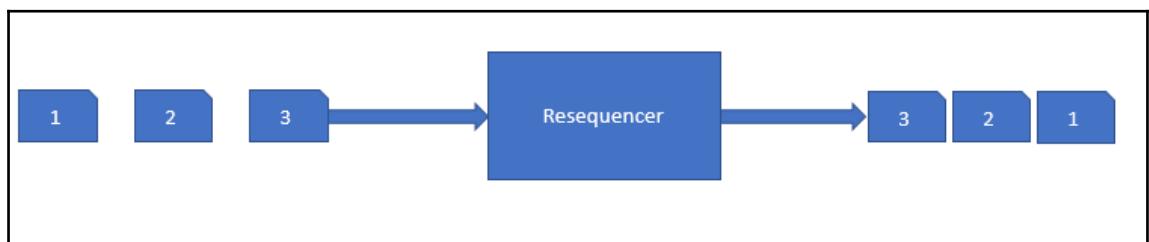
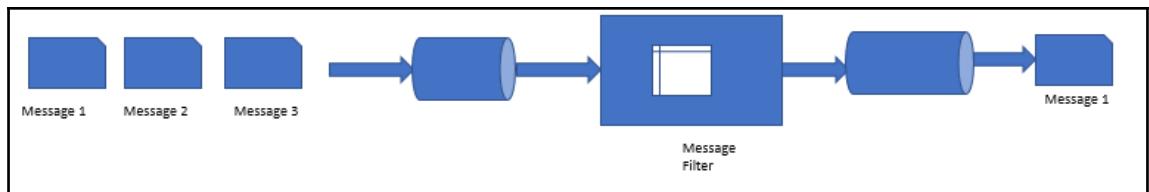
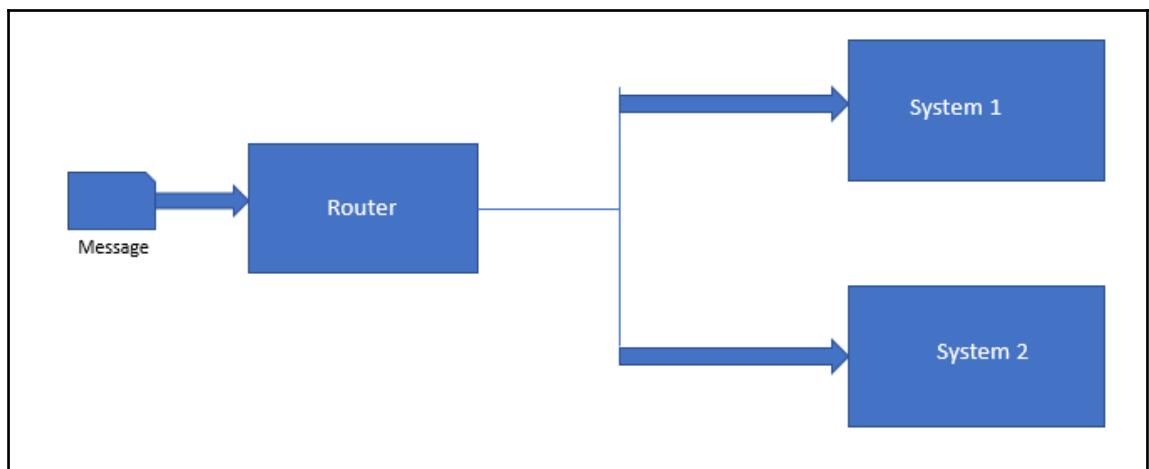


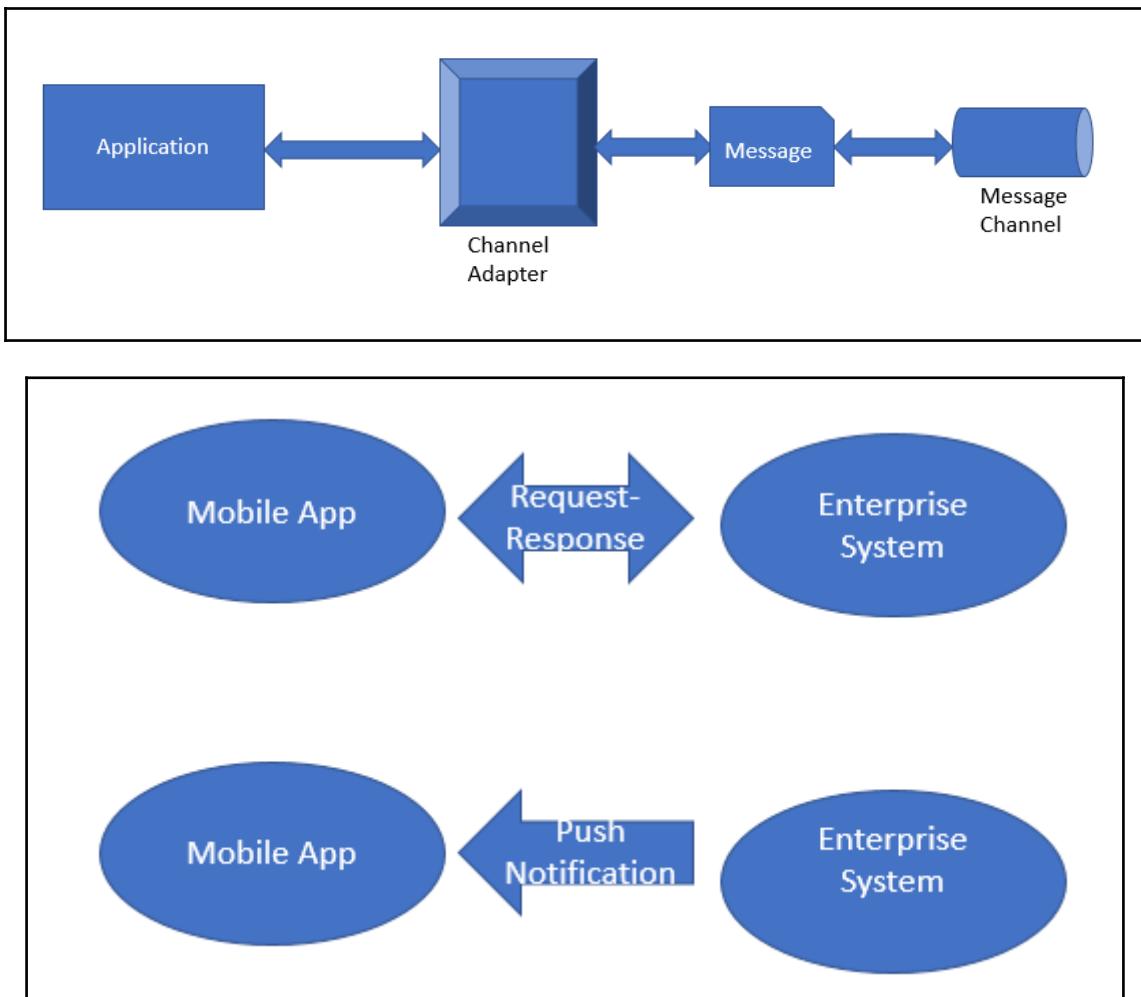


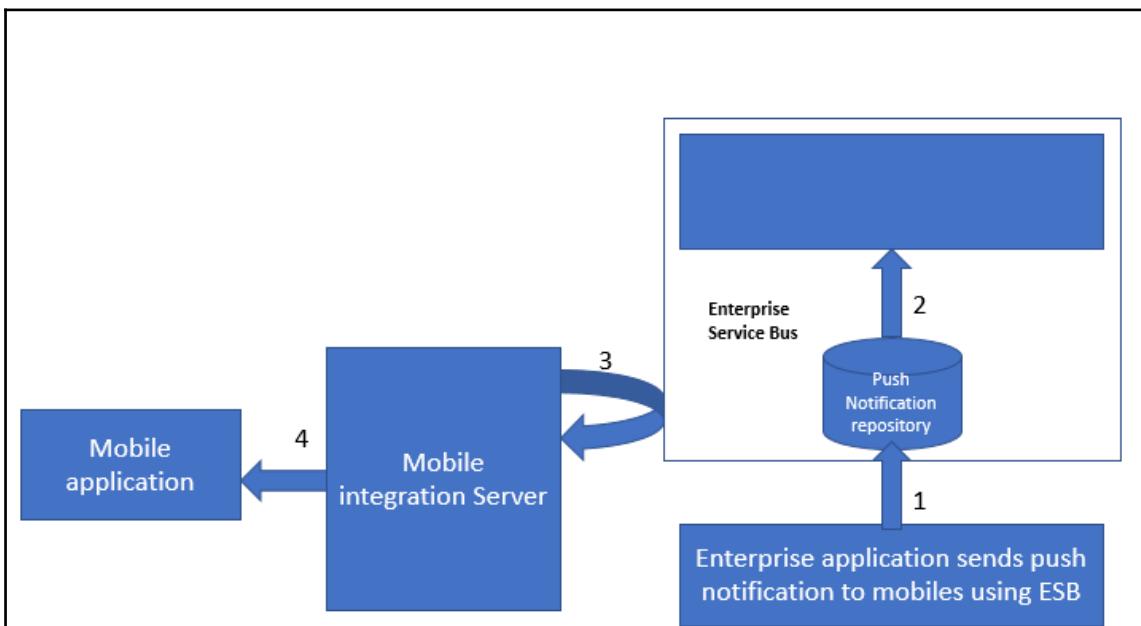
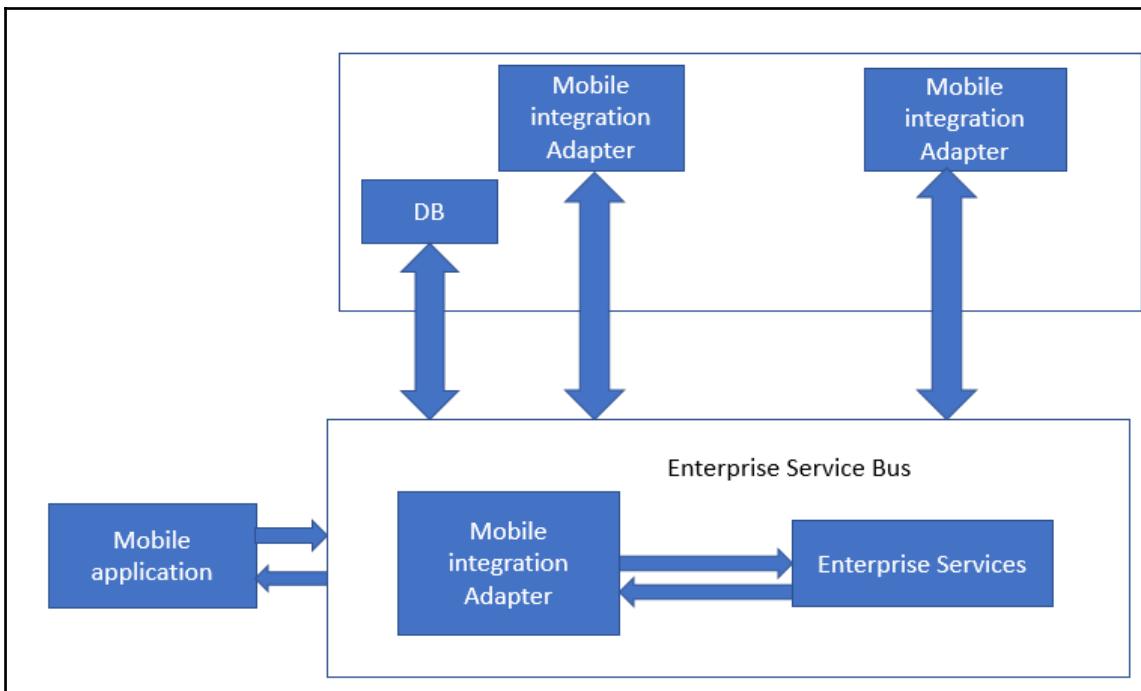




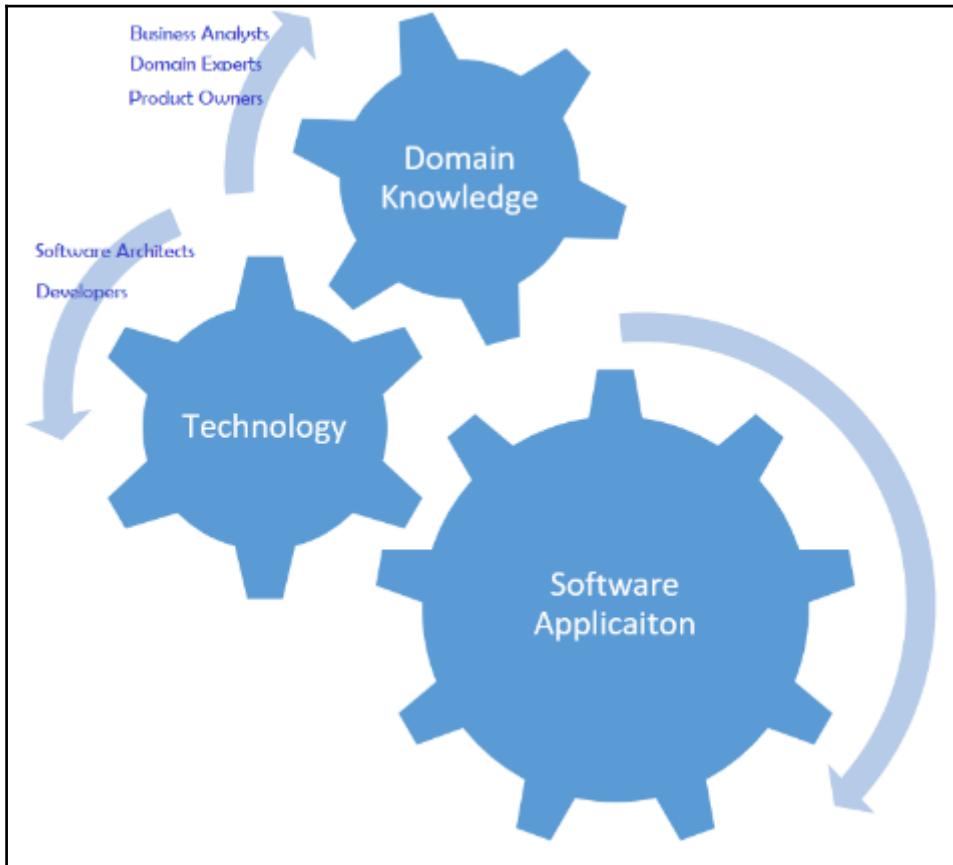


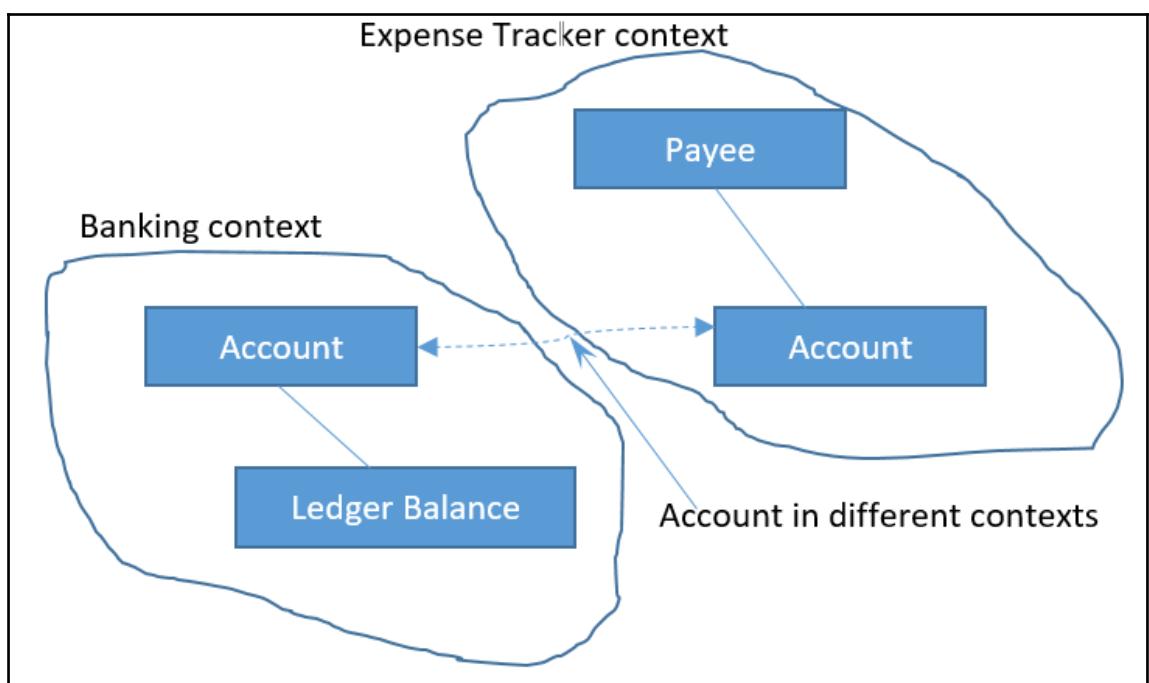


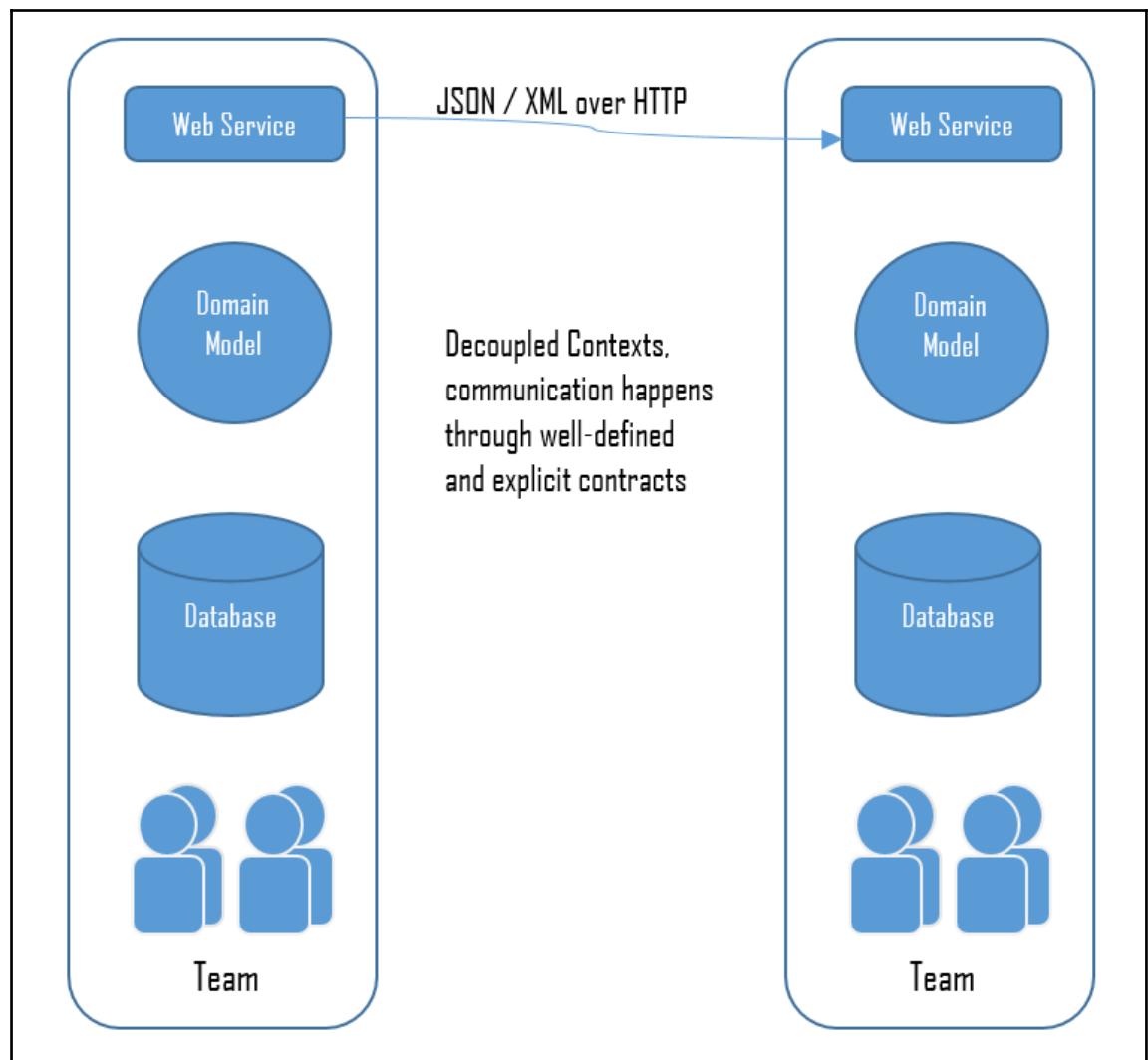


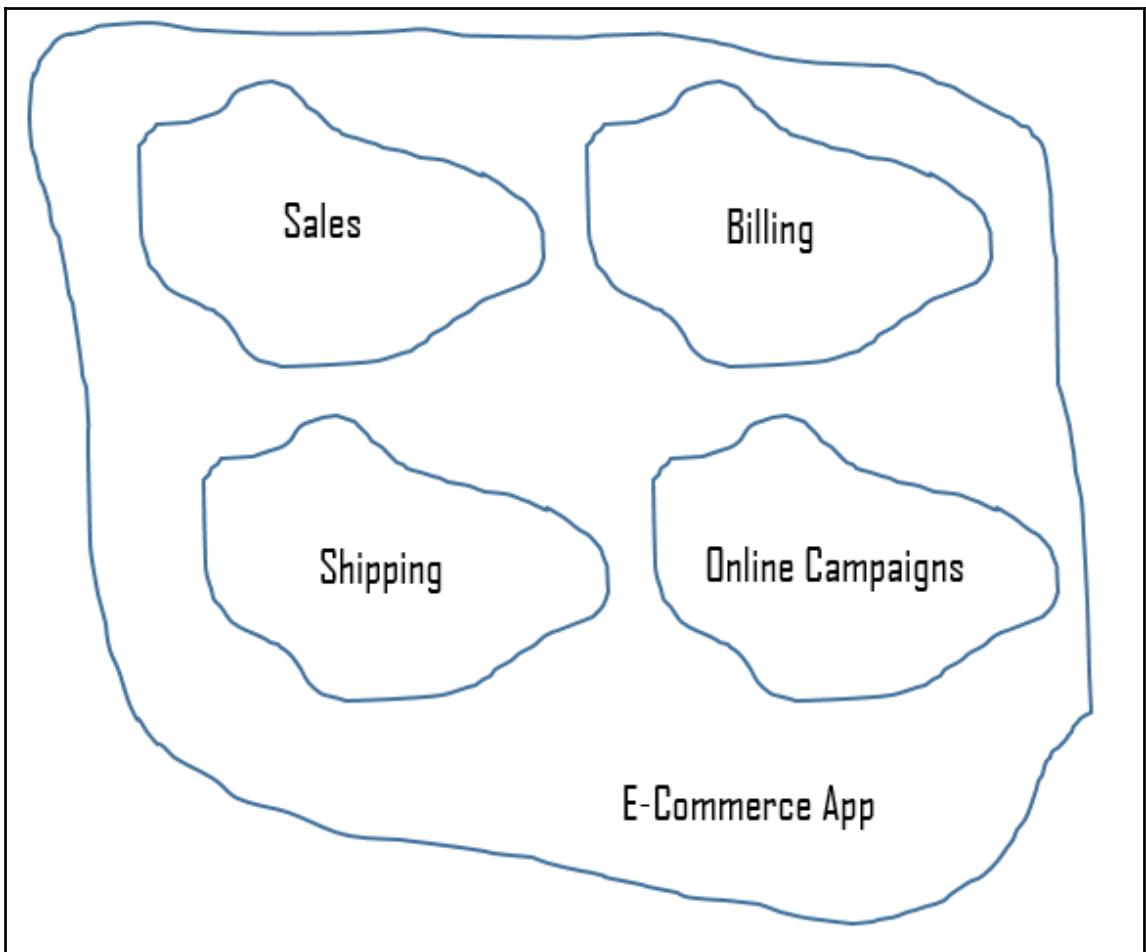


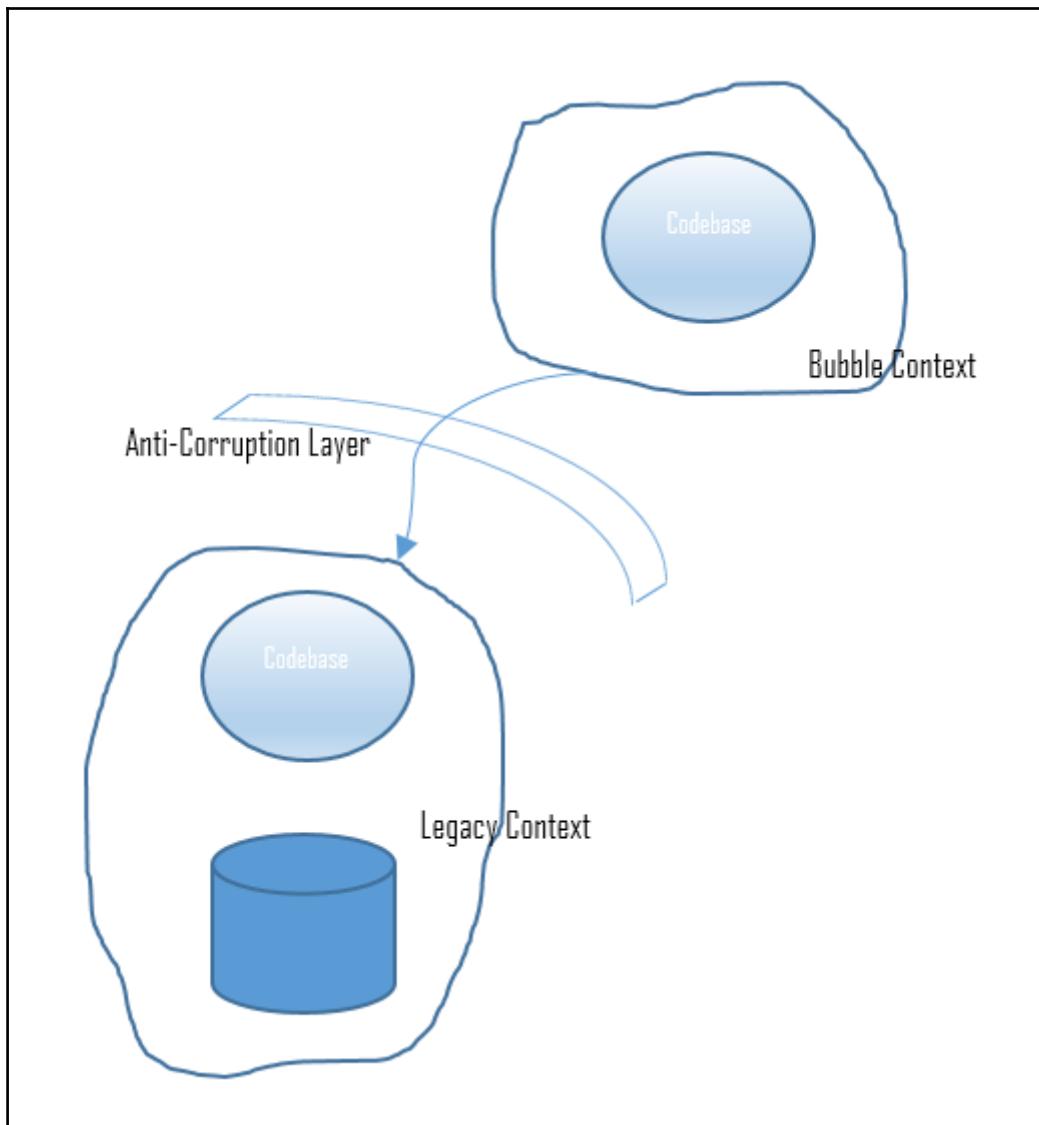
Chapter 5 - Domain-Driven Design (DDD) Principles and Patterns

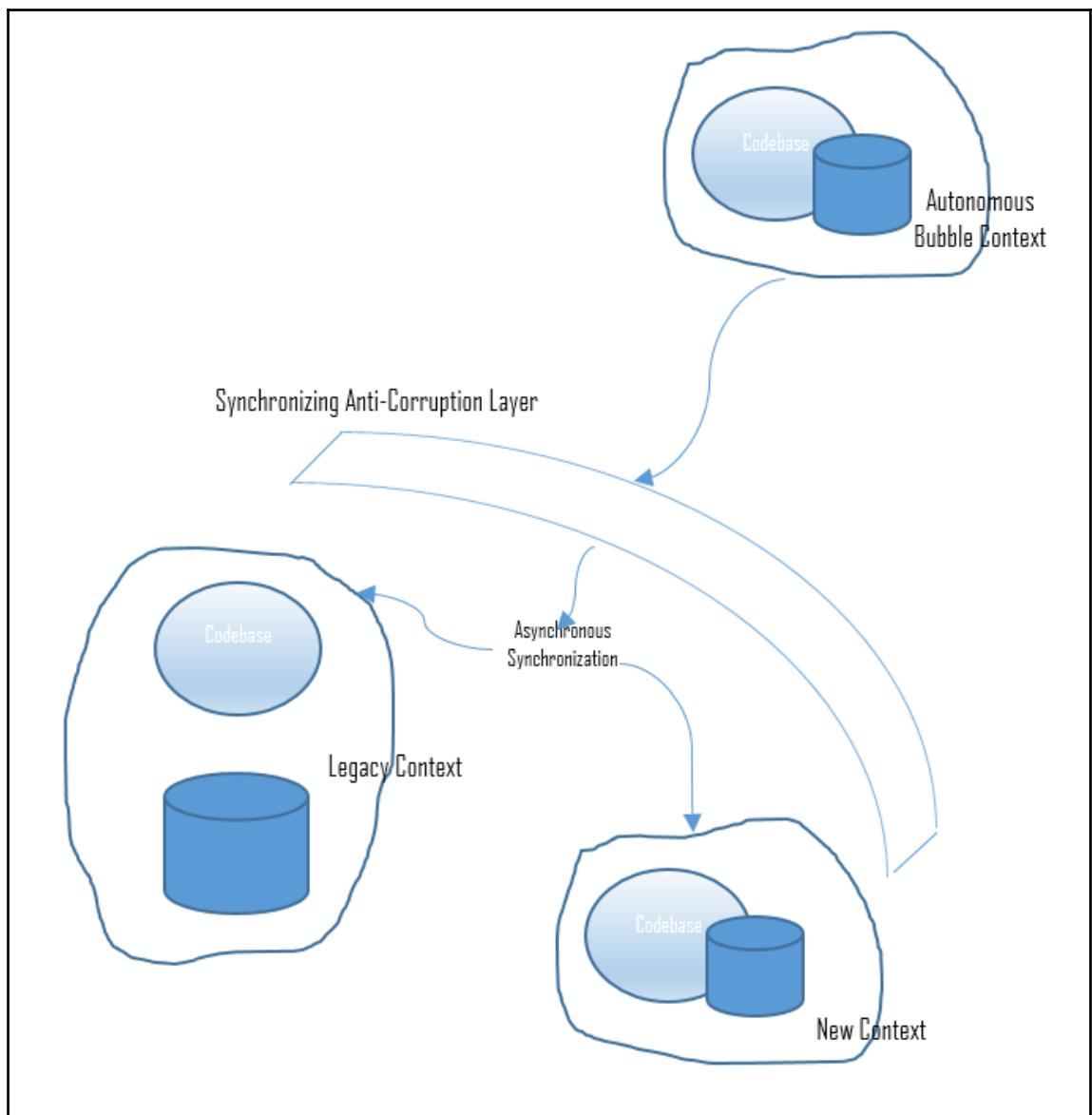


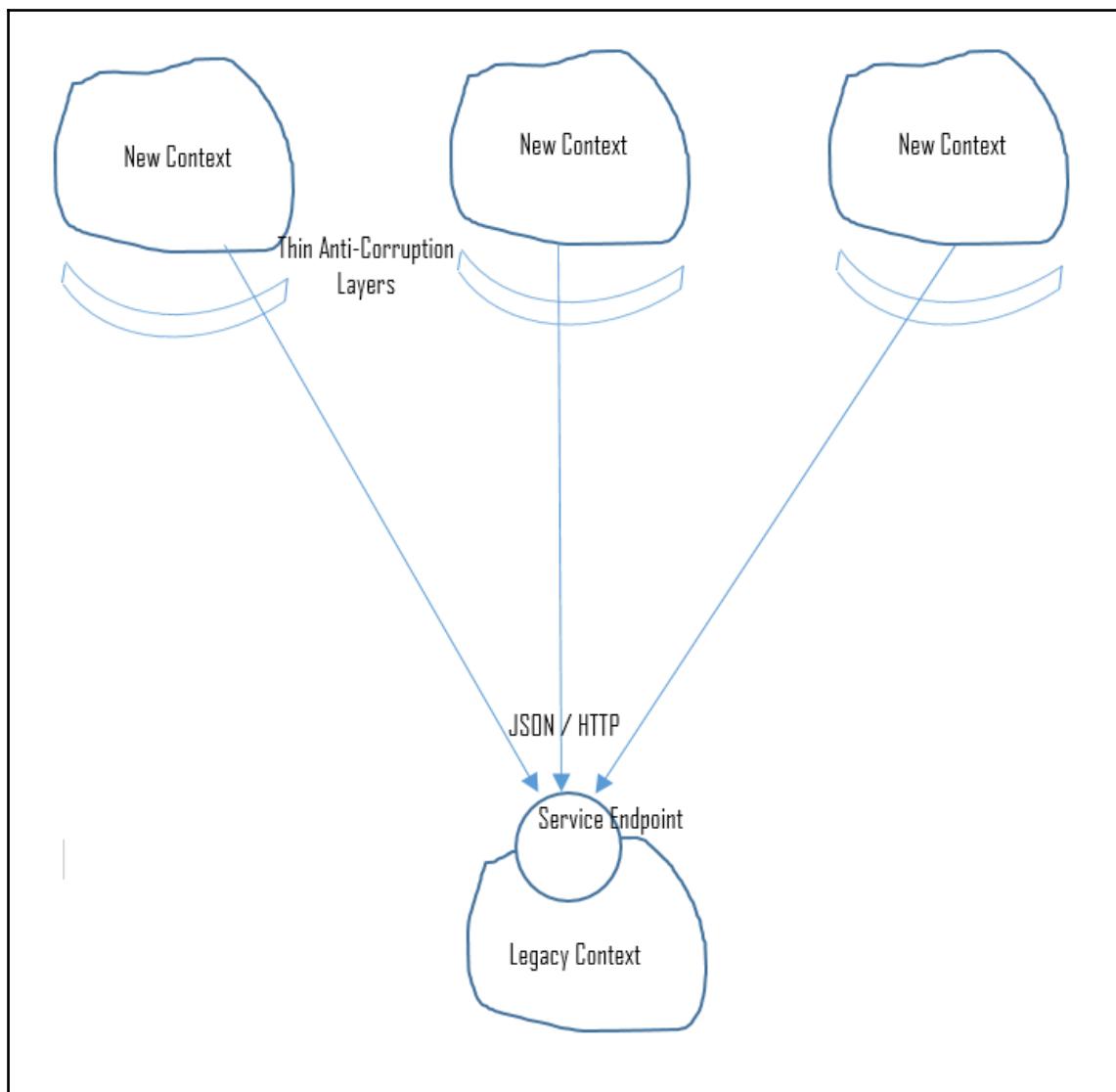


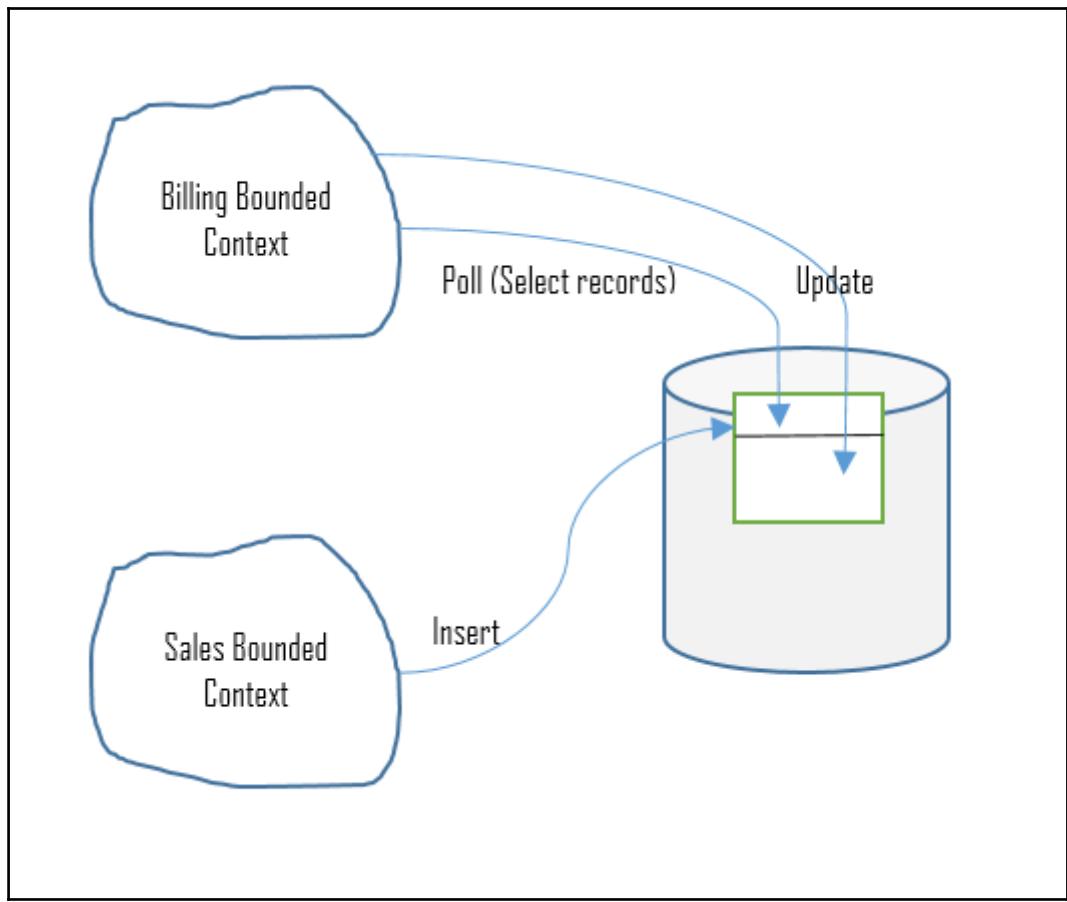


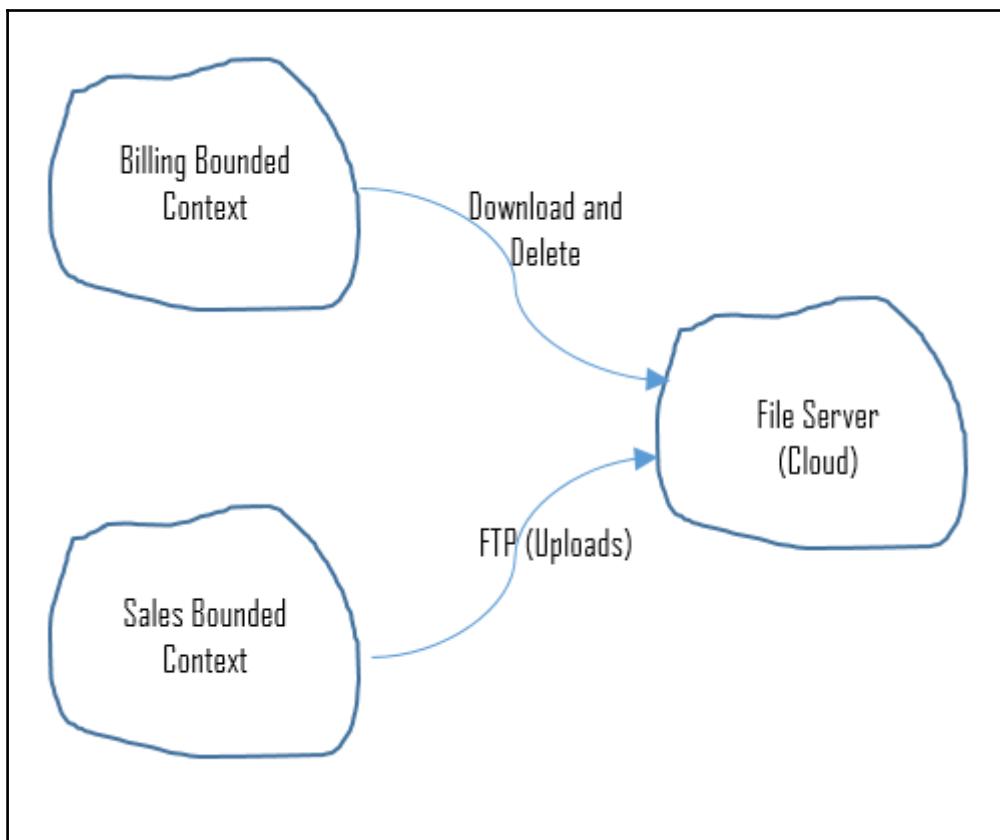


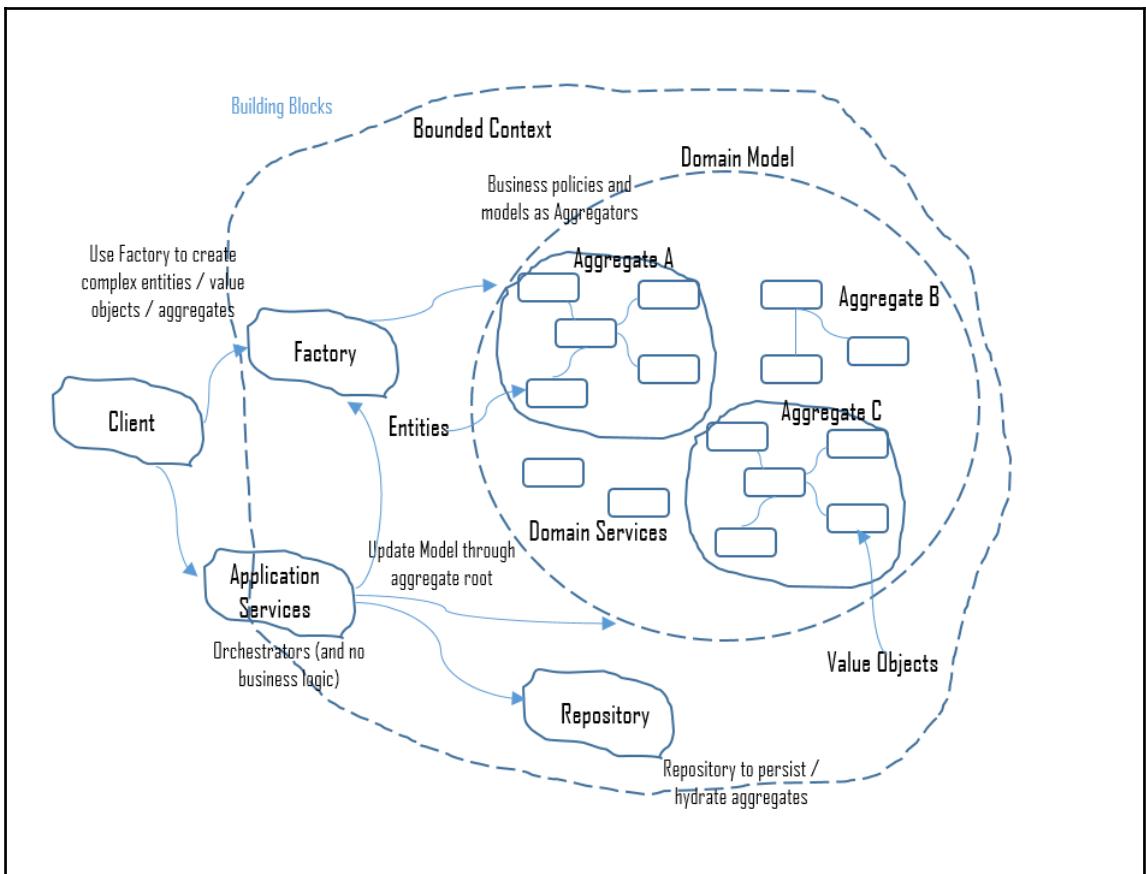


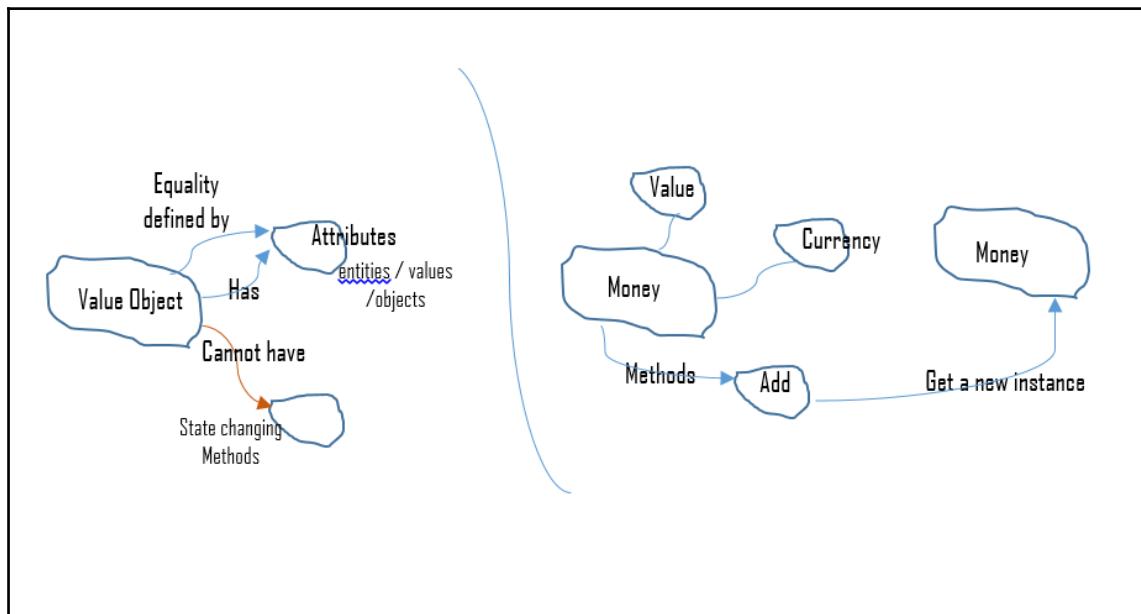
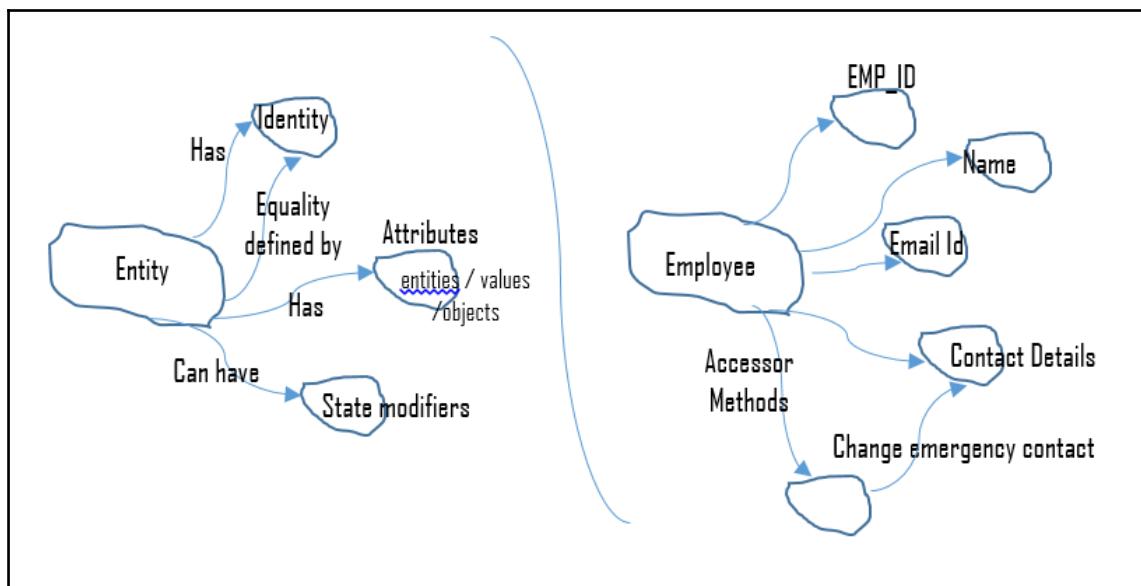


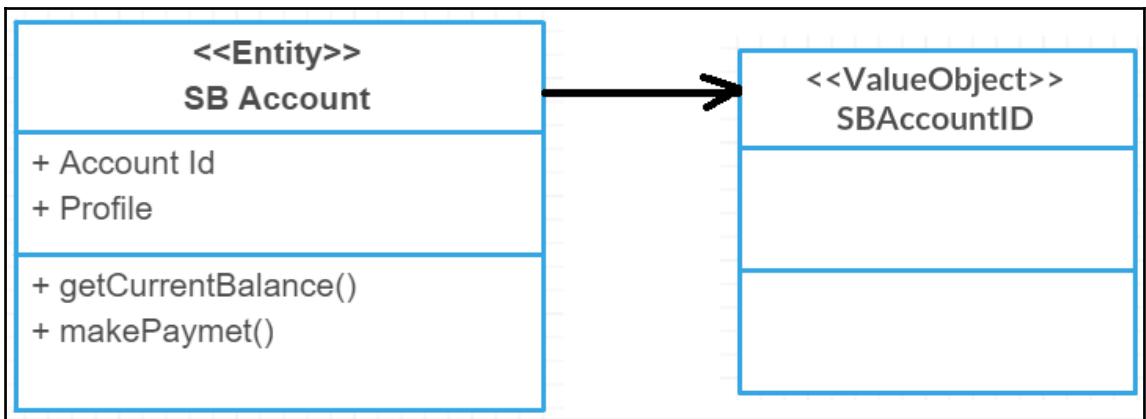


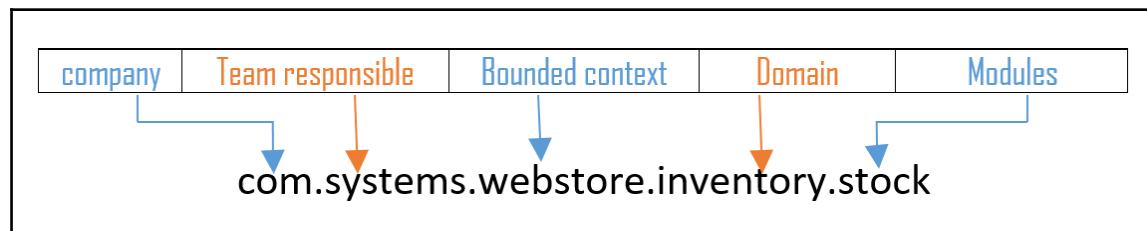
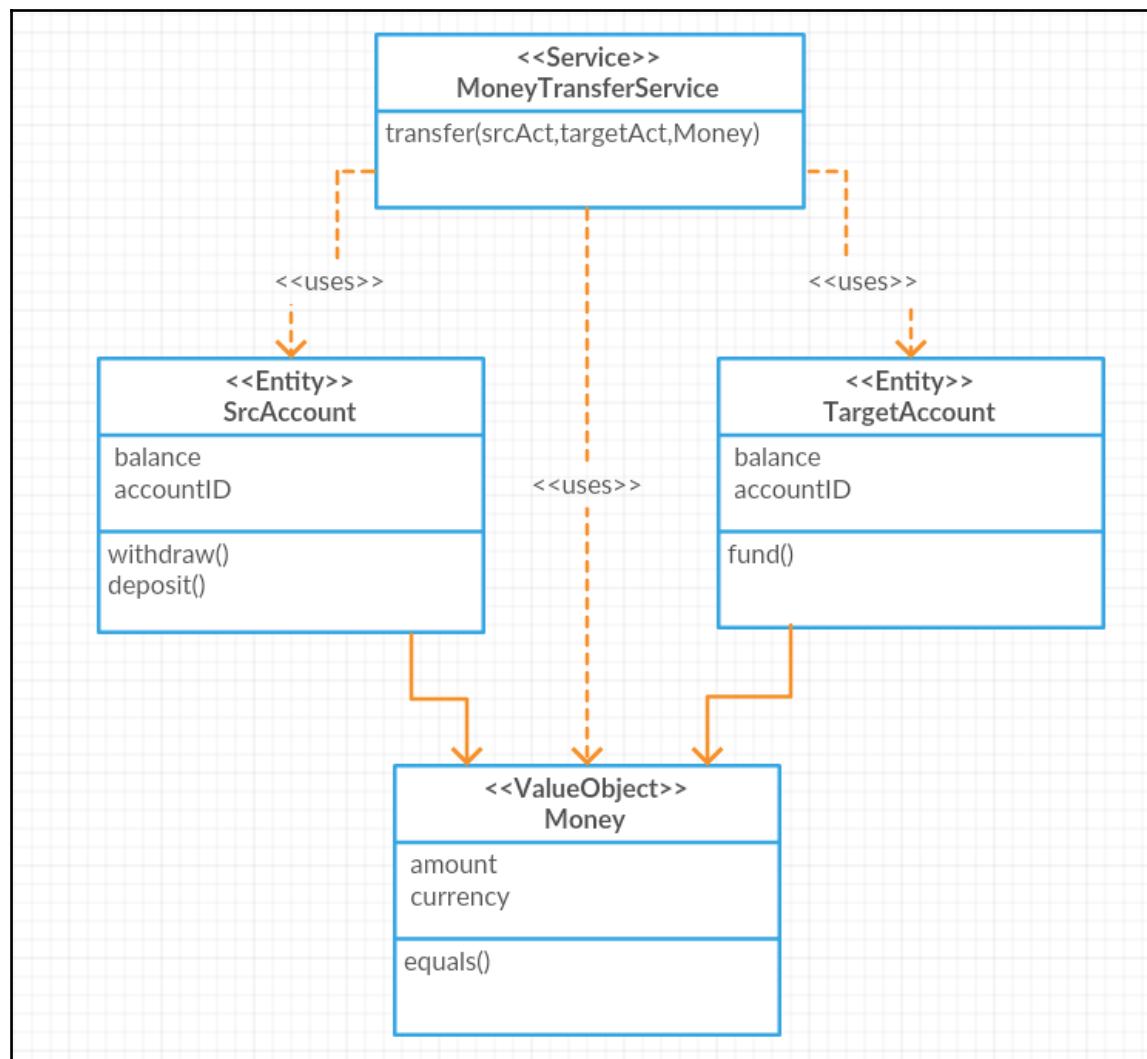


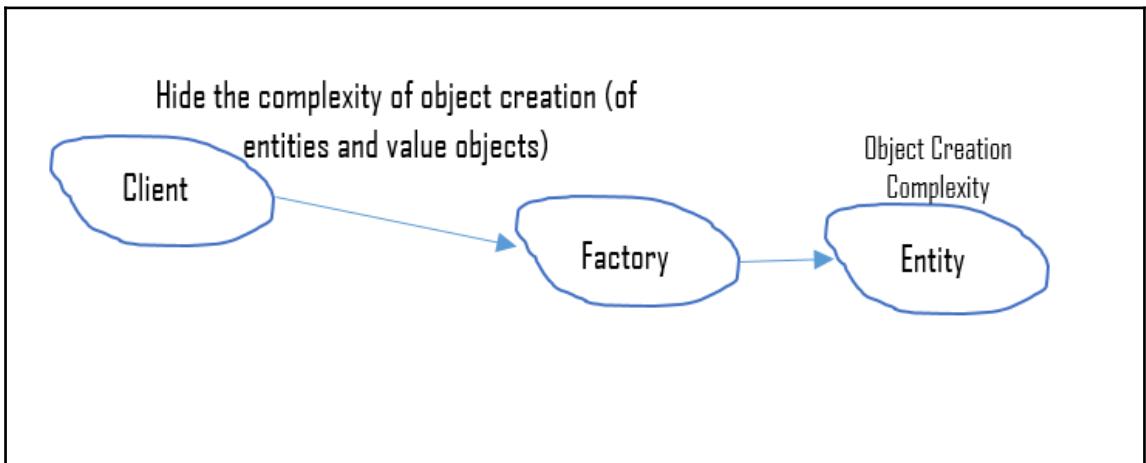
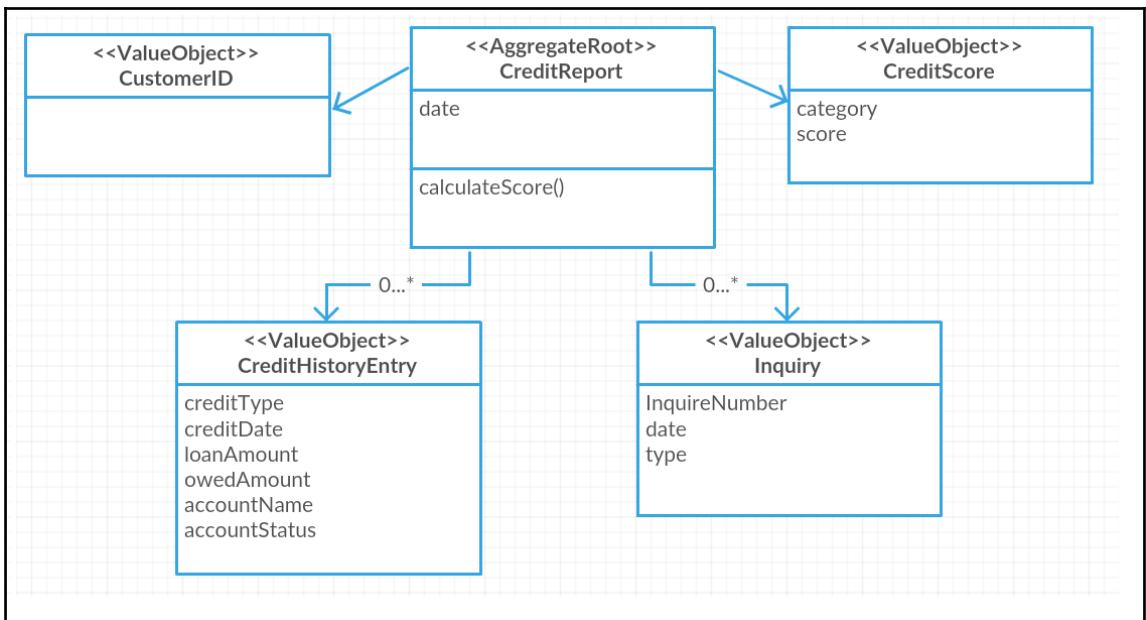


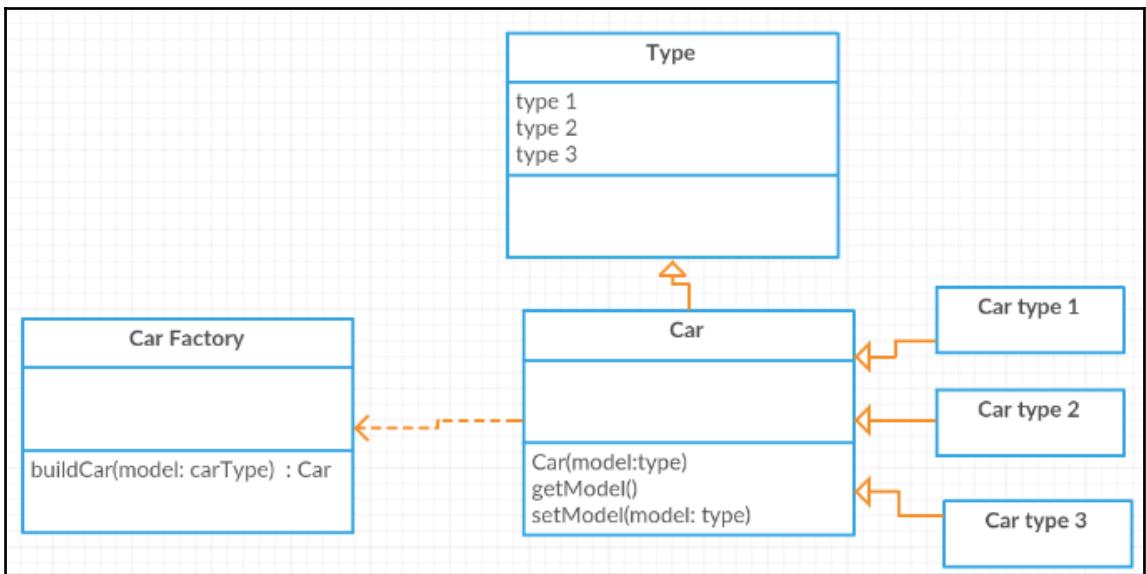


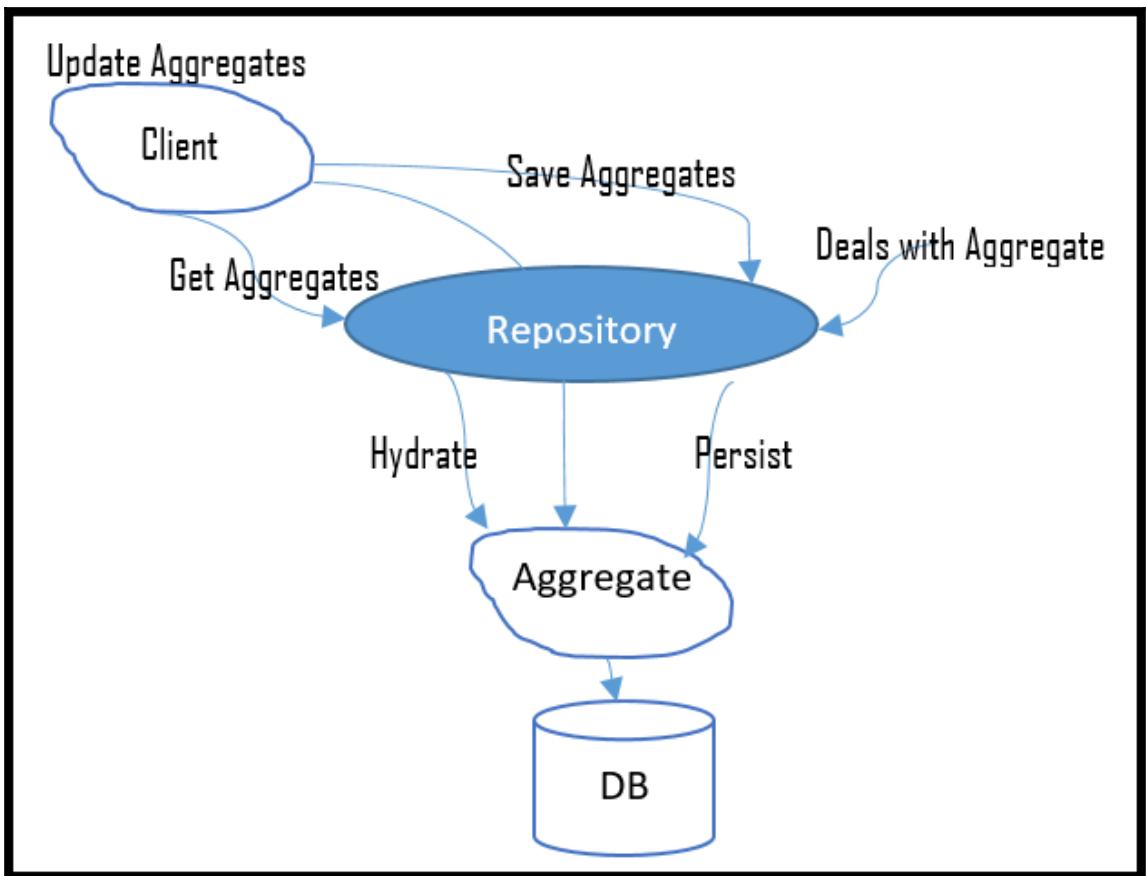


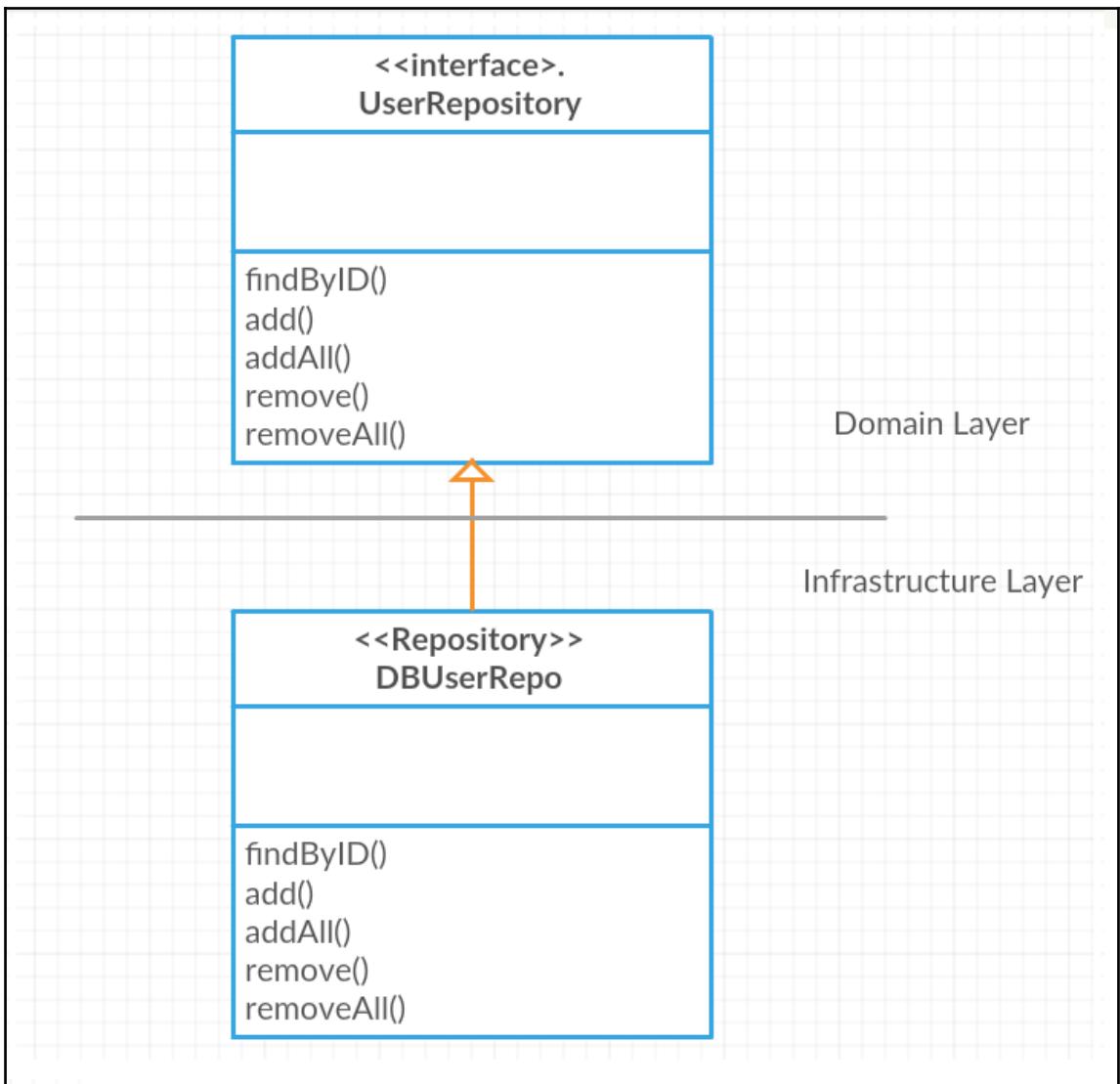


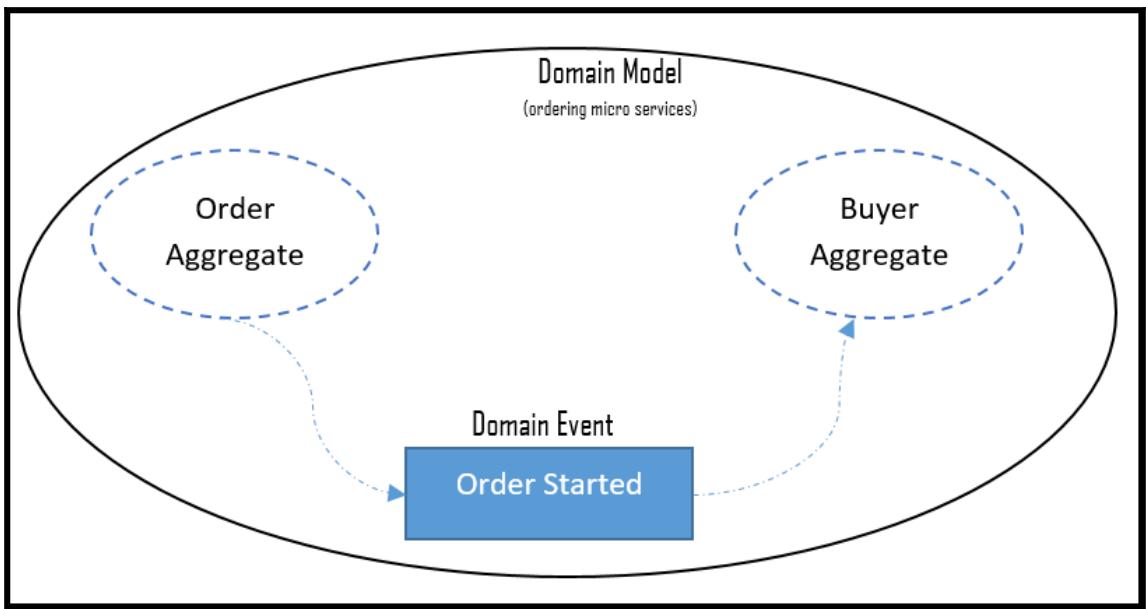




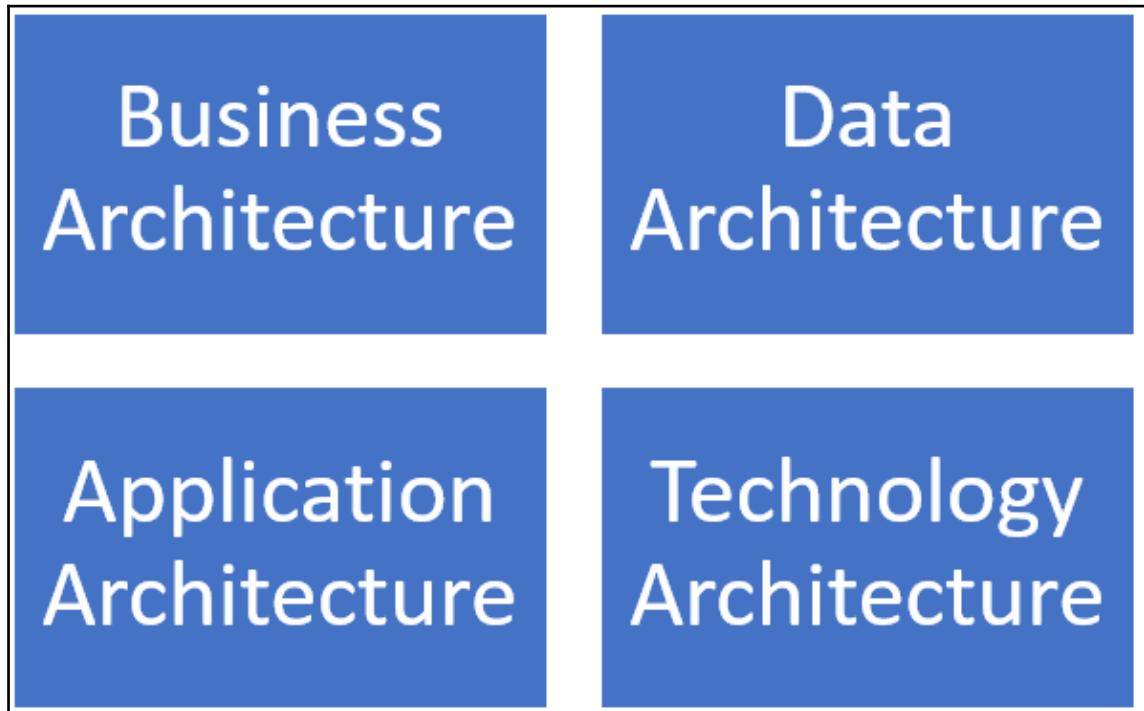








Chapter 6 - Enterprise Architecture Platforms and Tools

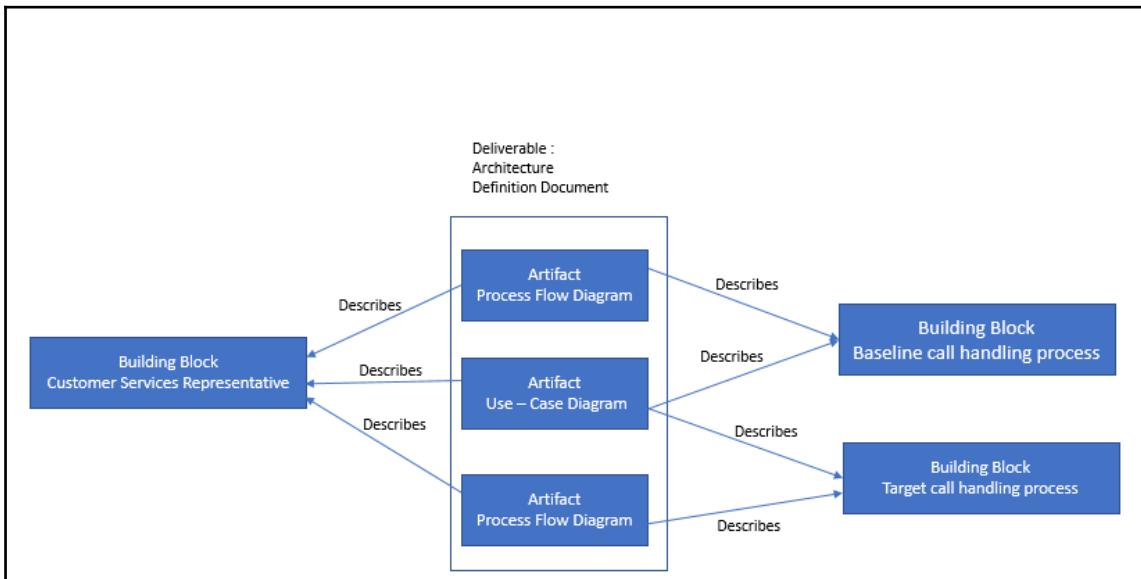
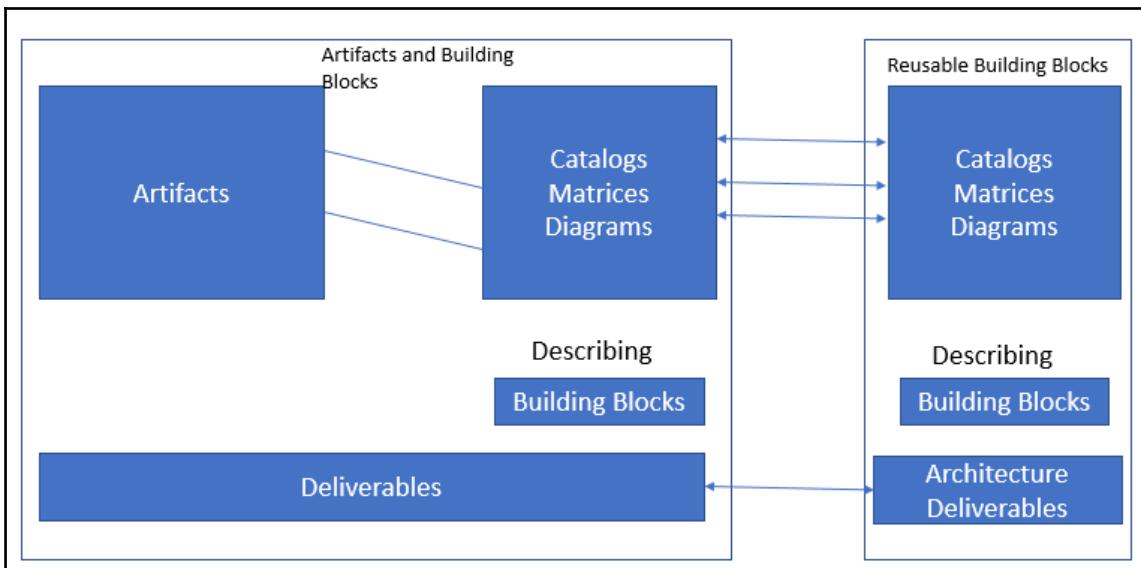


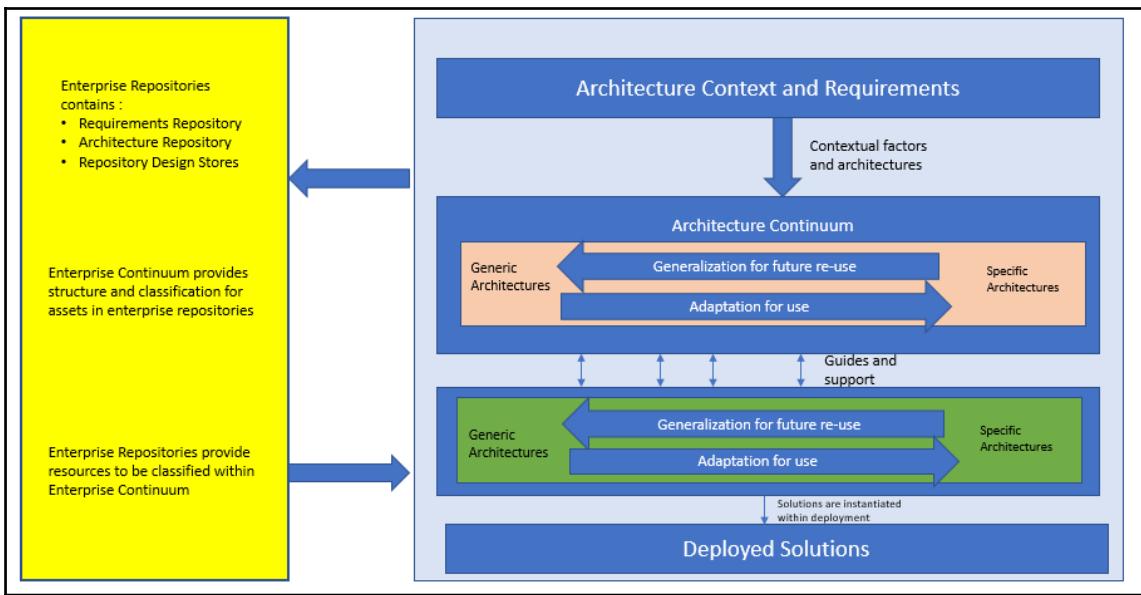


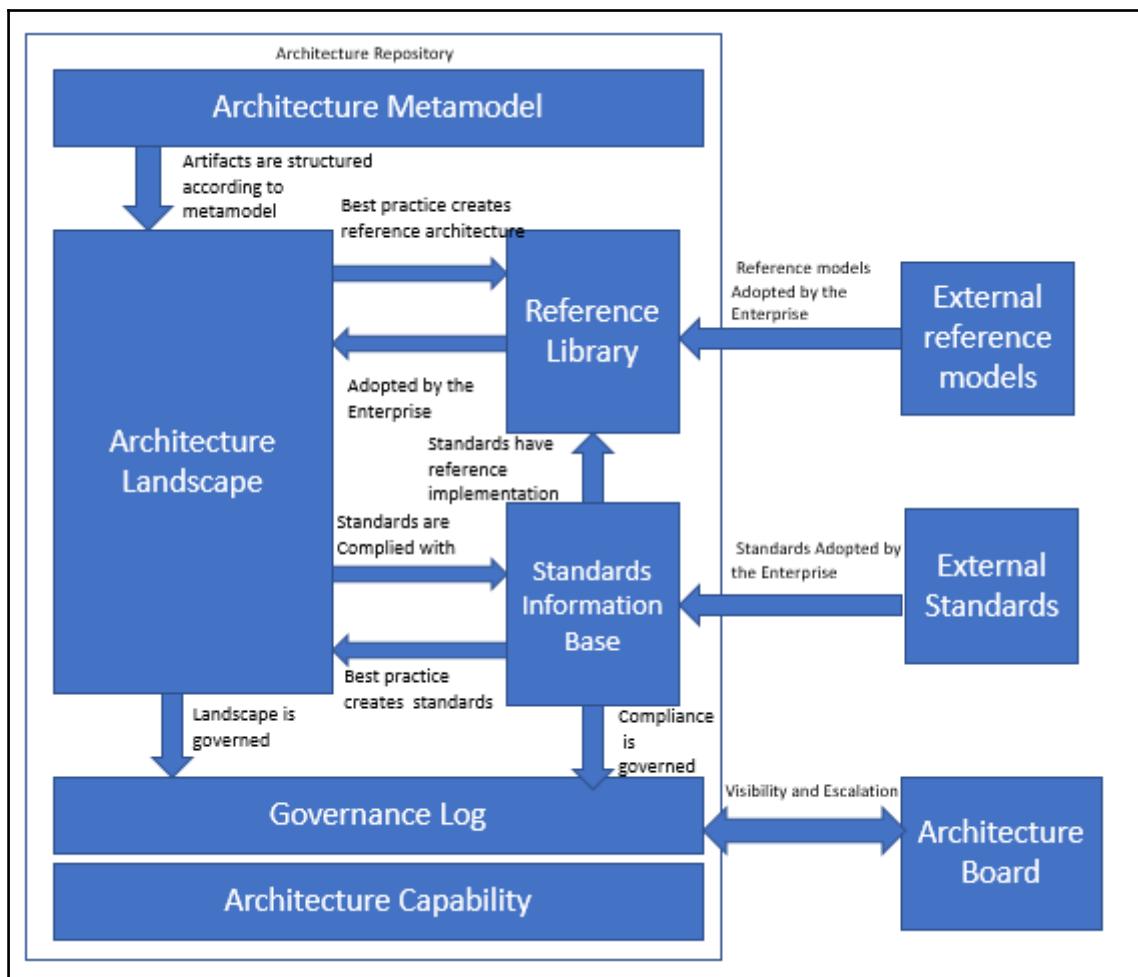
Deliverable

Artifact

Building Block

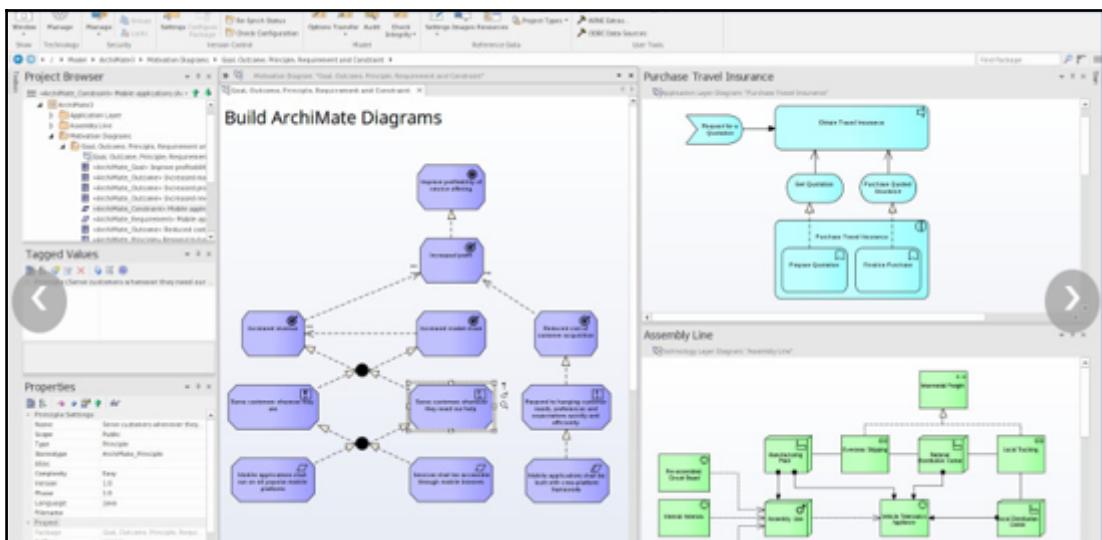




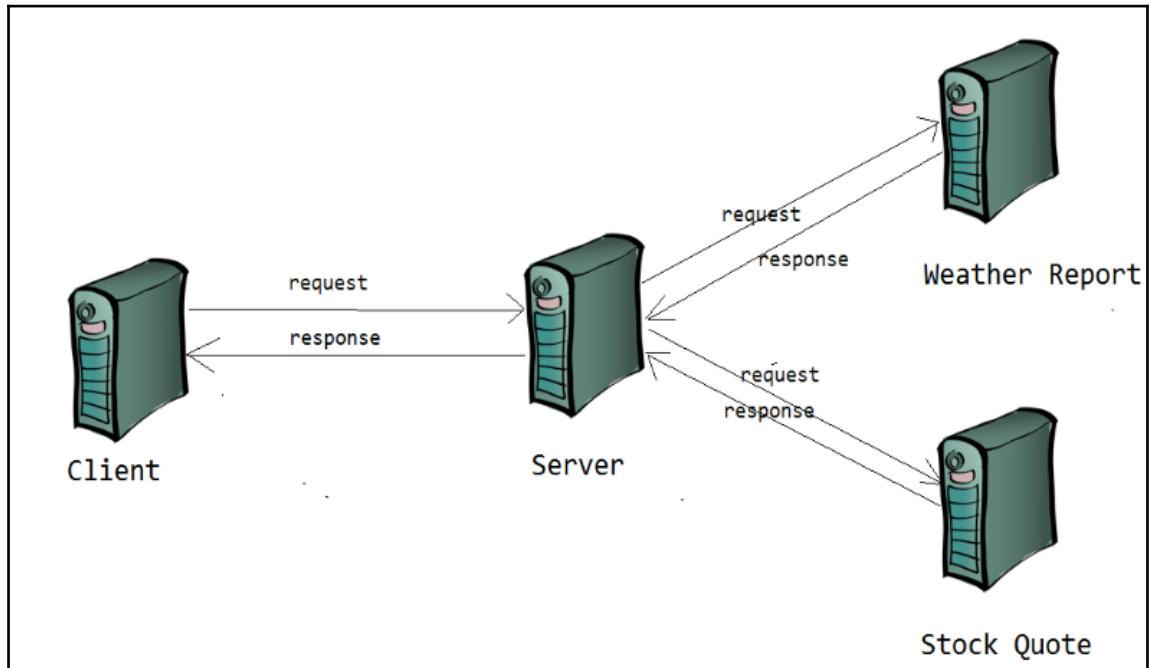


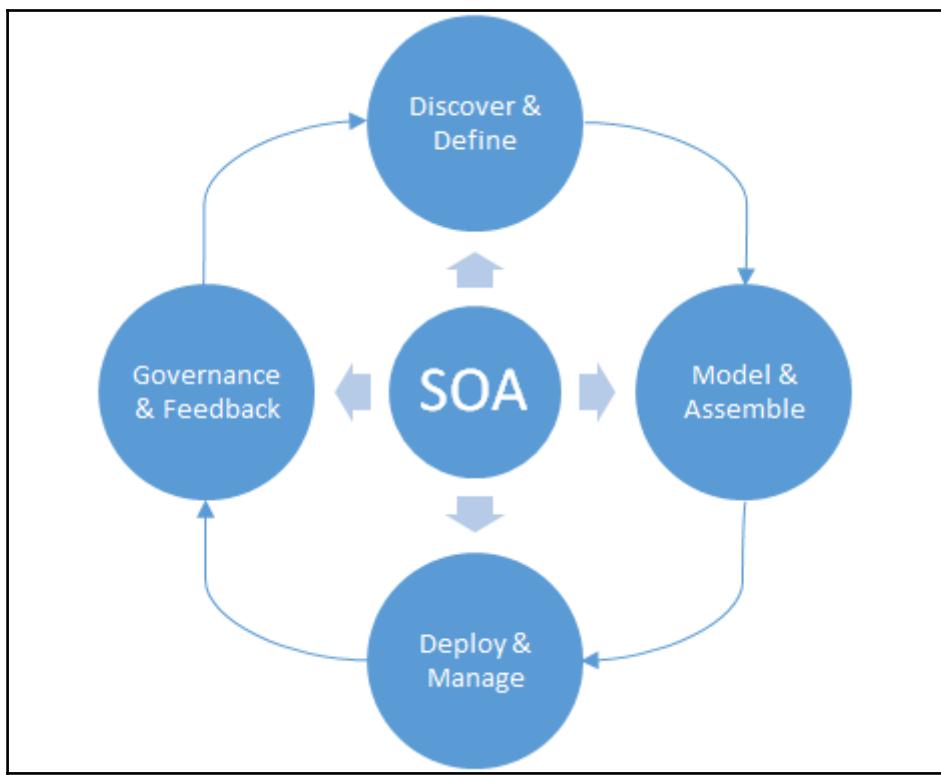
	What	How	Where	Who	When	Why	
Planner							Scope
Owner							Concepts
Designer							Logic
Builder							Physics
Implementer							Technology
Operator							Product
	Material	Process	Geometry	Instructions	Timing	Objectives	

Criteria	Zachman	TOGAF
Taxonomy Completeness	4	2
Process Completeness	1	4
Reference model guidance	1	3
Practice Guidance	1	2
Maturity Model	1	1
Business Focus	1	2
Governance Guidance	1	2
Partitioning Guidance	1	2
Prescriptive Catalog	1	2
Vendor Neutrality	2	4
Information Availability	2	4
Time to Value	1	3



Chapter 7 - Service-Oriented Architecture (SOA)

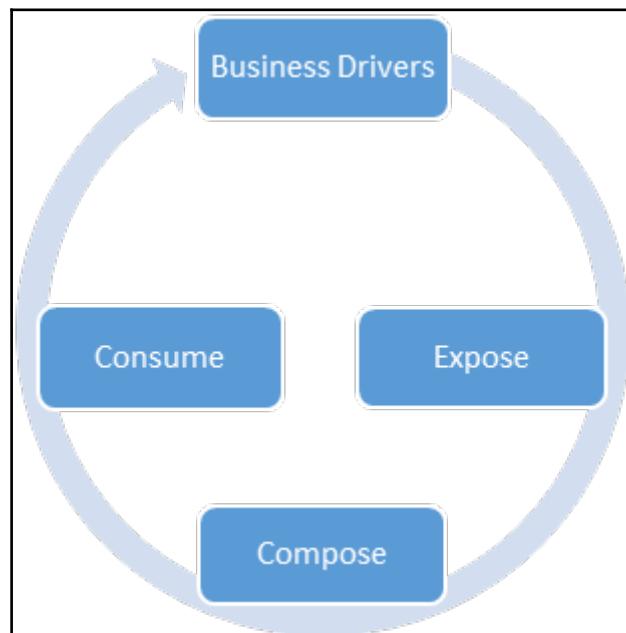


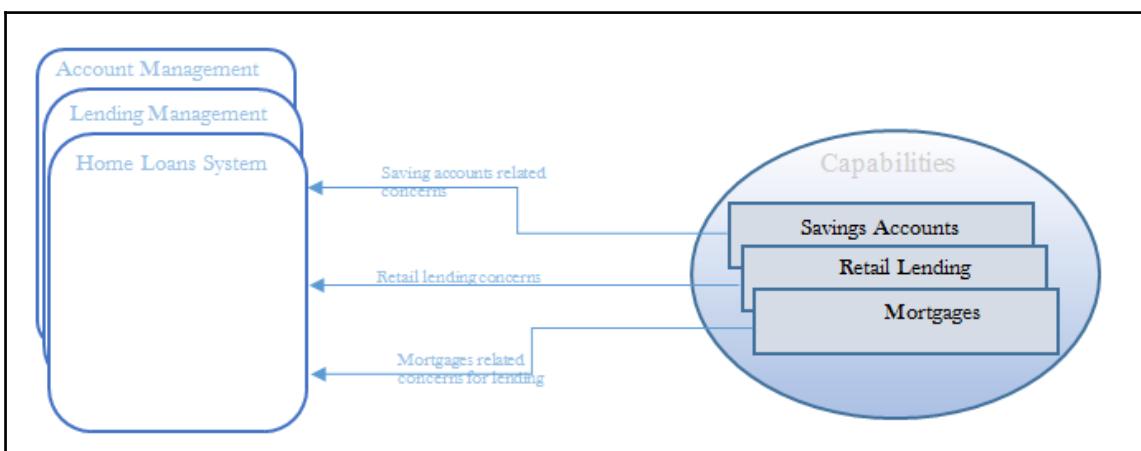
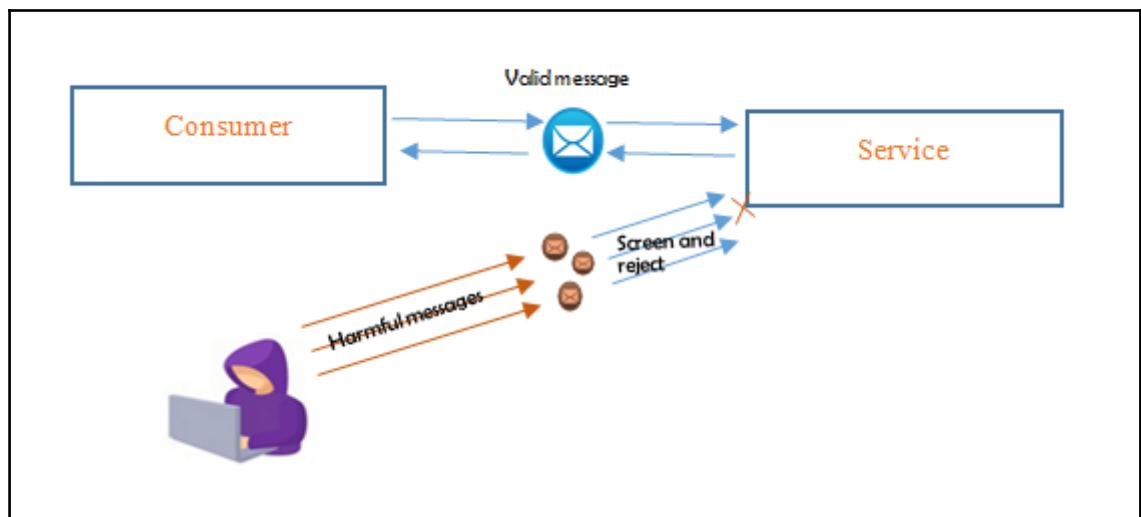


```

▼<definitions xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd=
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="http://graph
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsdl="http://schemas.xmls
  targetNamespace="http://graphical.weather.gov/xml/DWMLgen/wsdl/ndfdXML.wsdl">
  ▼<types>
    ▼<xsd:schema targetNamespace="http://graphical.weather.gov/xml/DWMLgen/wsdl/ndf
      <xsd:import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
      <xsd:import namespace="http://schemas.xmlsoap.org/wsdl/" />
      ▼<xsd:complexType name="weatherParametersType">
        ▶<xsd:all>...</xsd:all>
      </xsd:complexType>
    </xsd:schema>
  </types>
  ▼<message name="NDFDgenRequest">
    <part name="latitude" type="xsd:decimal"/>
    <part name="longitude" type="xsd:decimal"/>
    <part name="product" type="xsd:string"/>
    <part name="startTime" type="xsd:dateTime"/>
    <part name="endTime" type="xsd:dateTime"/>
    <part name="Unit" type="xsd:string"/>
    <part name="weatherParameters" type="tns:weatherParametersType"/>
  </message>
</definitions>

```

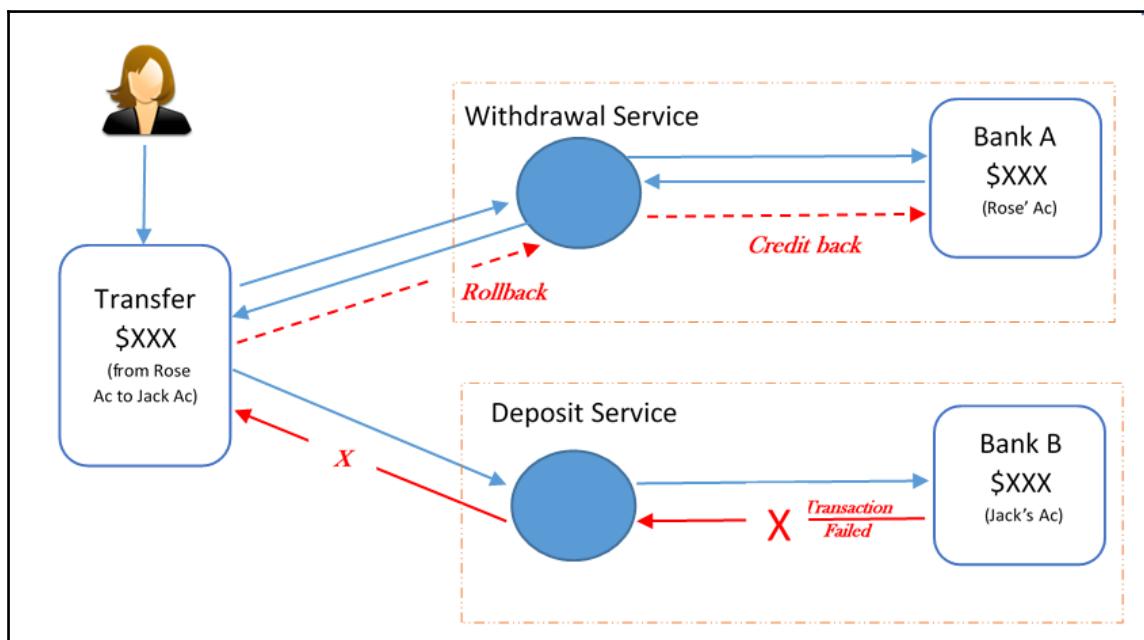


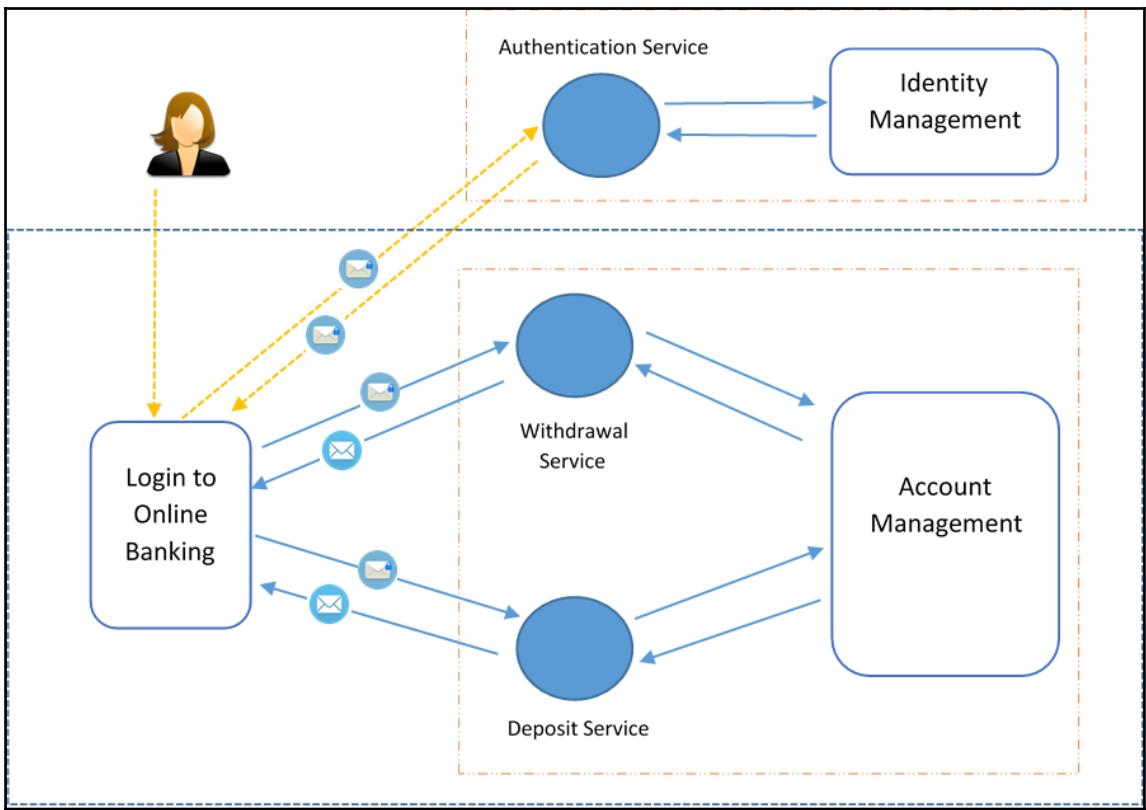


```

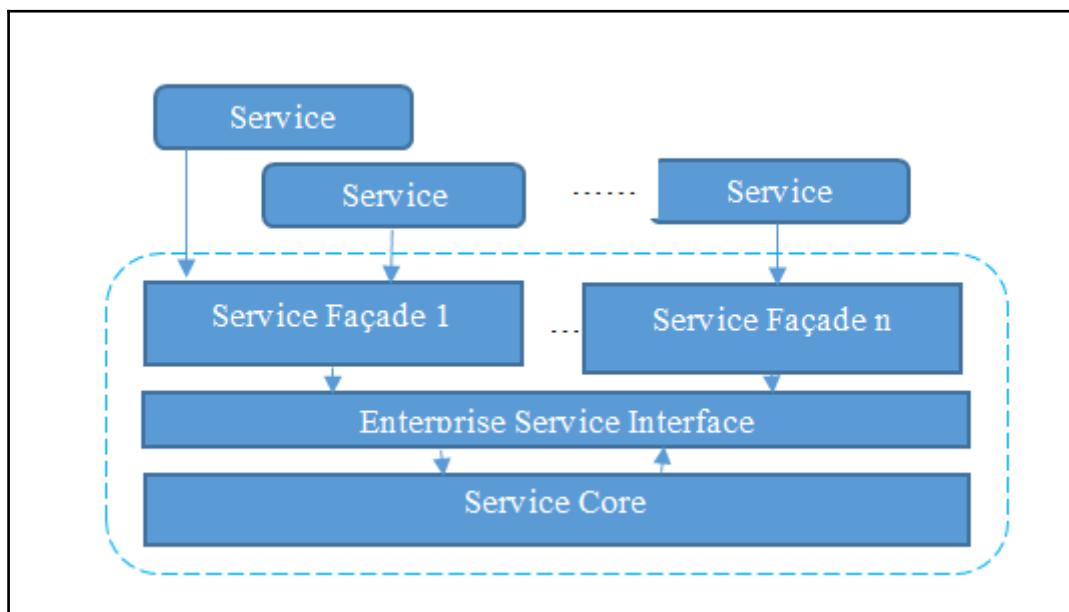
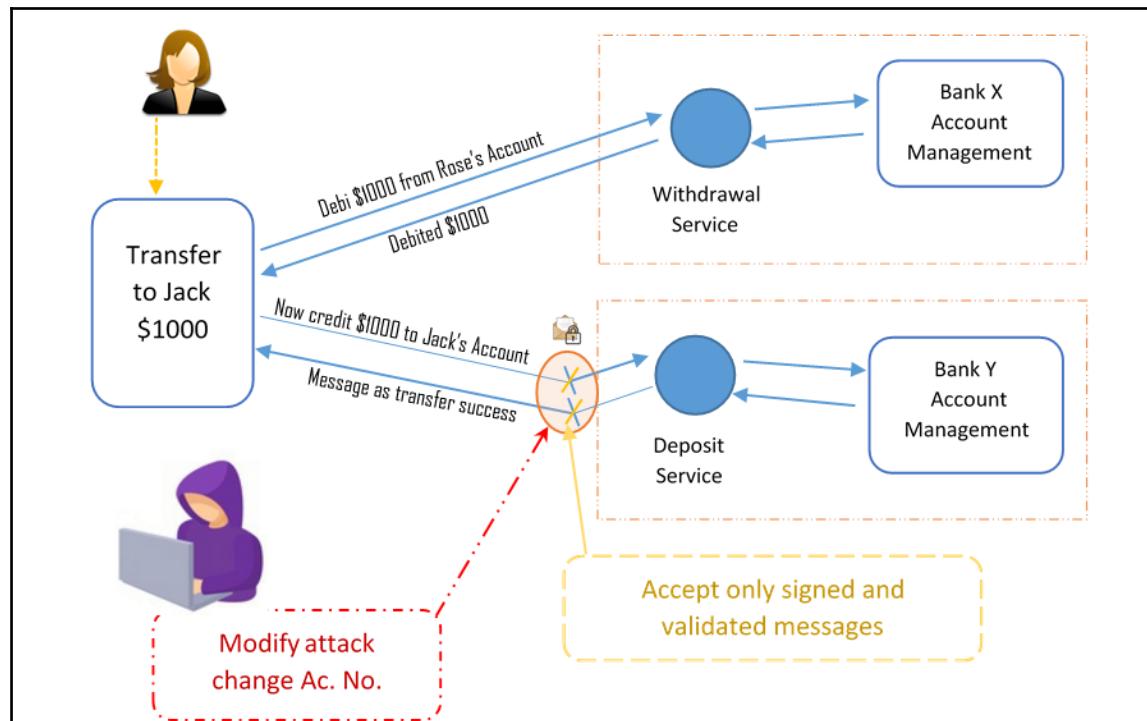
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
    <soap:Body>
        <GetMortgagePayment xmlns="http://www.webserviceX.NET/">
            <Years>15</Years>
            <Interest>12</Interest>
            <LoanAmount>2000000</LoanAmount>
            <AnnualTax>5</AnnualTax>
            <AnnualInsurance>1000</AnnualInsurance>
        </GetMortgagePayment>
        <GetLoanAccountProfile>
            <AccountNumber>1203493</AccountNumber>
            <AccountType>SB/LA</AccountType>
        </GetLoanAccountProfile>
    </soap:Body>
</soap:Envelope>

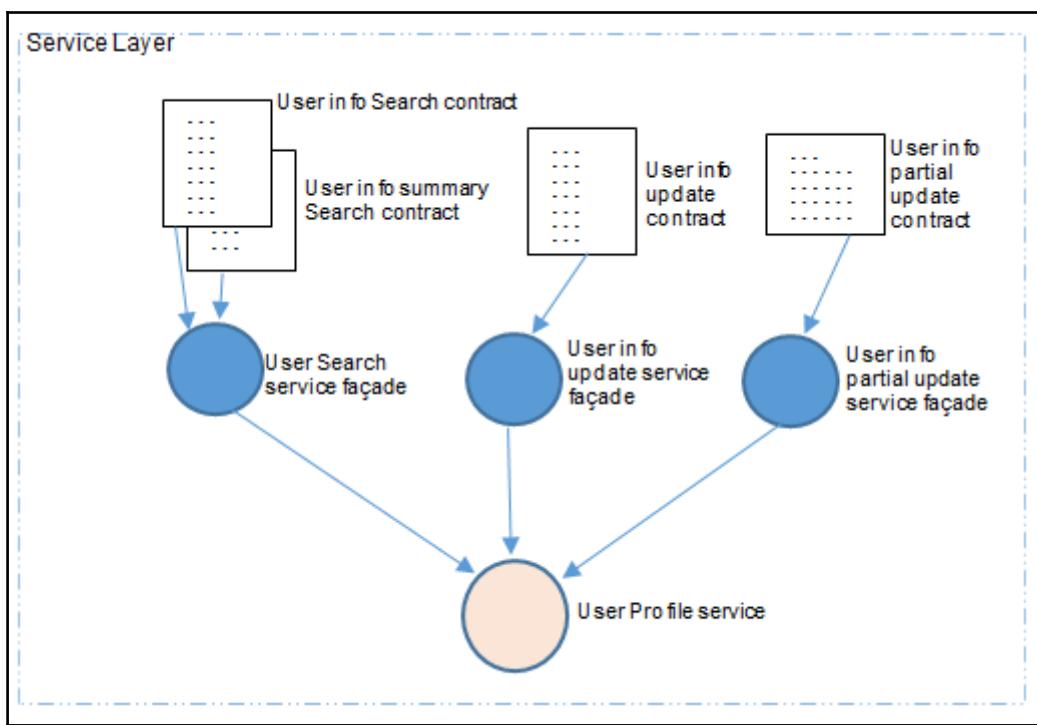
```

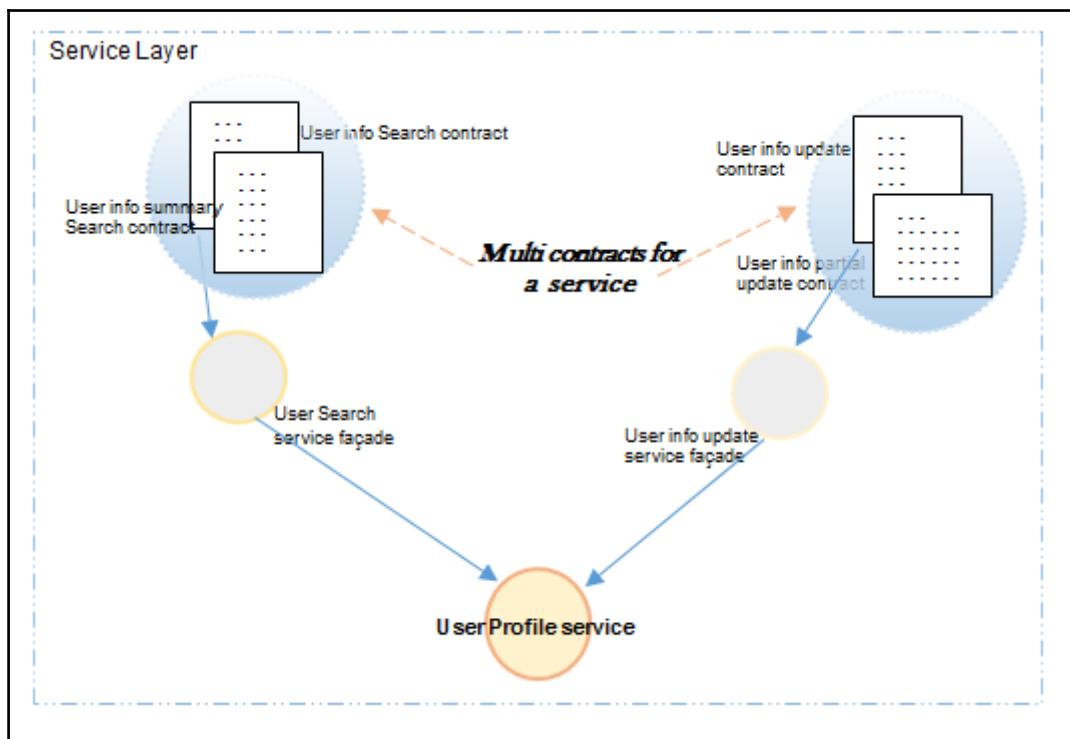




```
<Assertion ID="ed9f29bd-52ca-4237-95c1-eb53f70fe8e5" IssueInstant="2017-11-06T00:45:30.593Z"
Version="2.0" xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
<Issuer>ADatum</Issuer>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:SignedInfo>
<ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
<ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
<ds:Reference URI="#ed9f29bd-52ca-4237-95c1-eb53f70fe8e5">
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
<ds:DigestValue>.....
```







CALL ME BACK (web application)

Enter your 10 digit mobile number or your landline number including the STD code and click **Call me**. Our customer service representative will call you back on the number you enter.



Your number **9748573983**

When to call **within 30 mins**

Call me



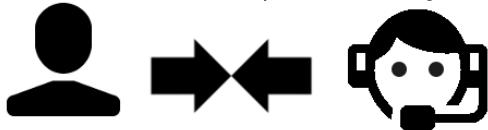
Long Wait Time

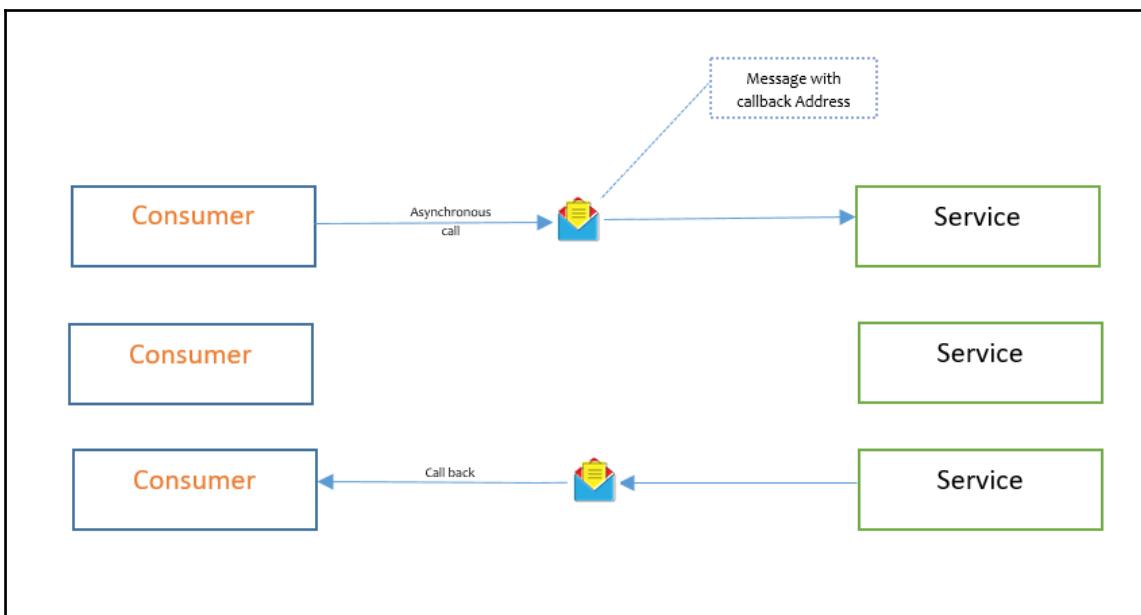
**Contact Center
Closed**



Callback App

Calls Agent and Customer at the requested time and joins the call.

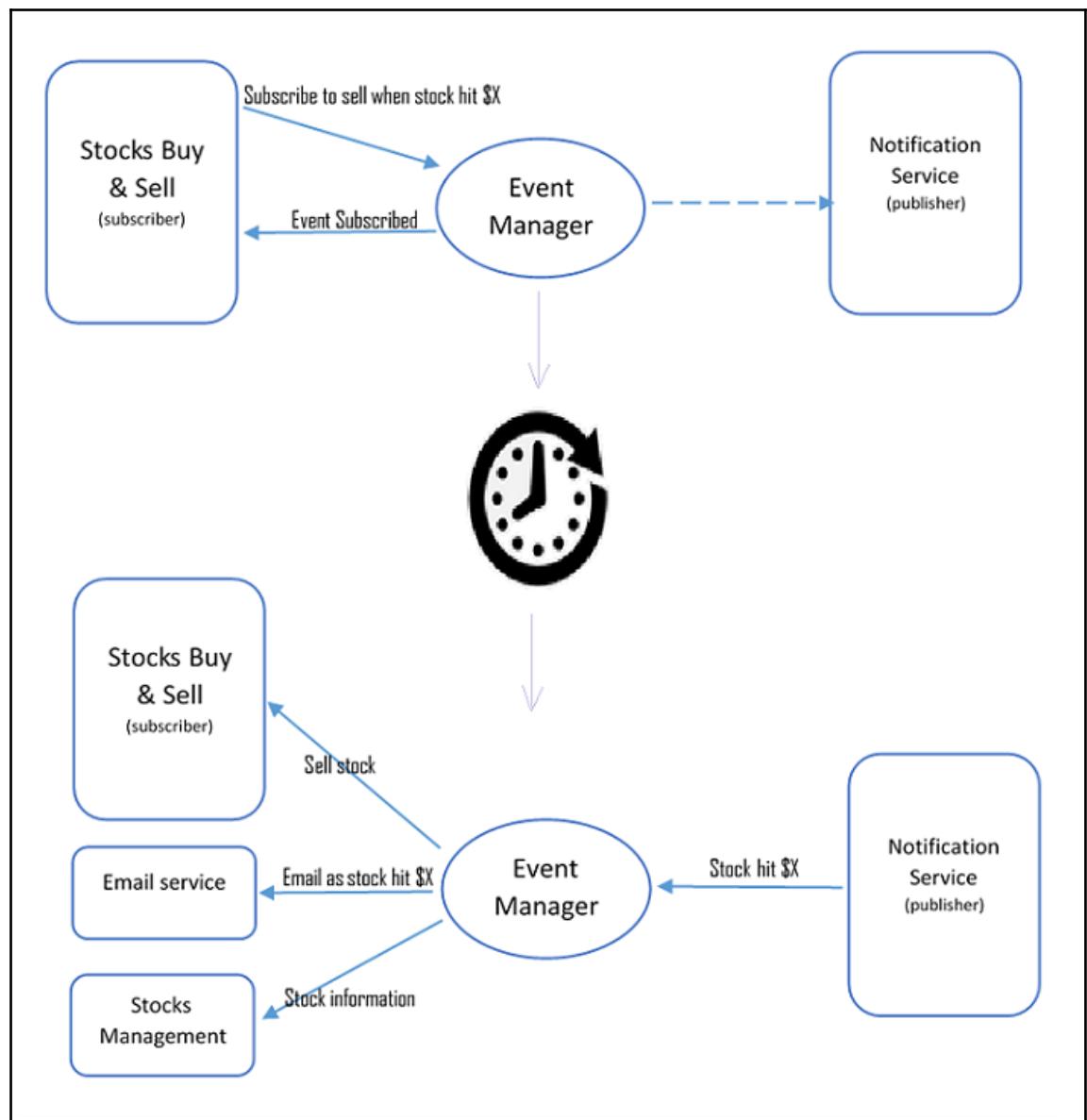




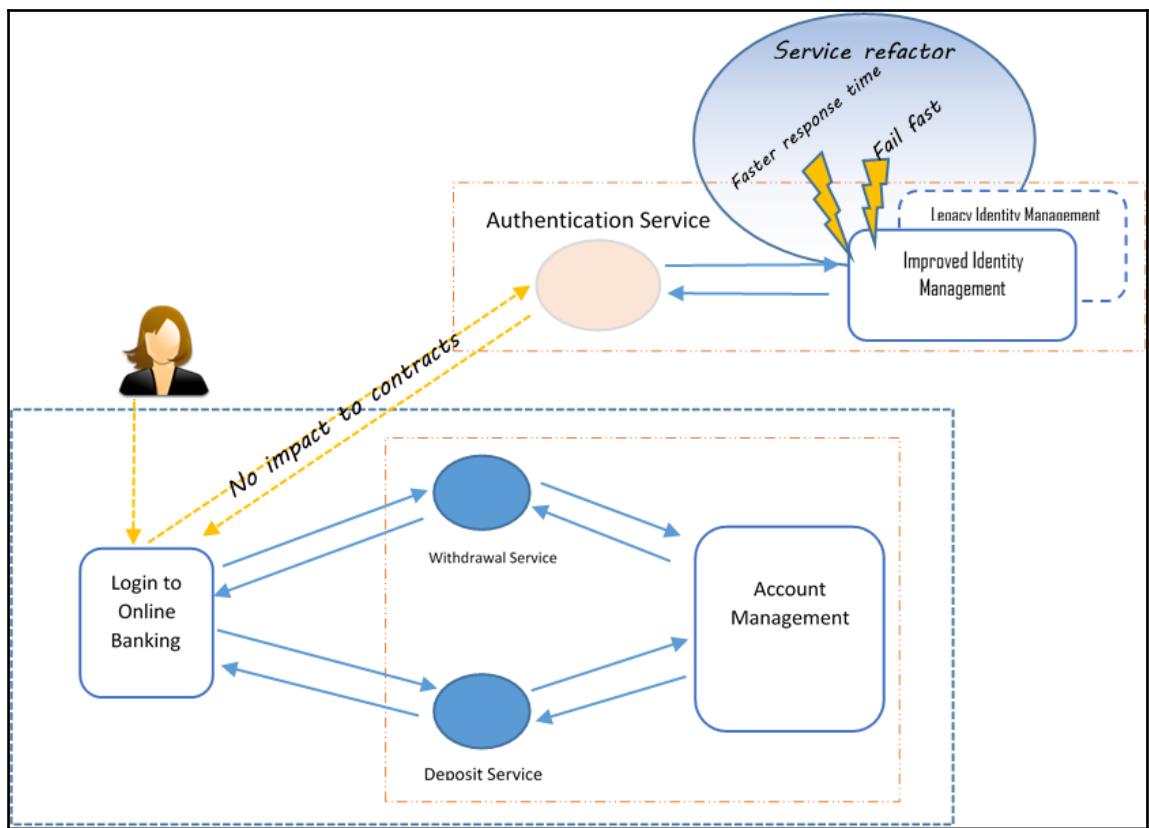
```

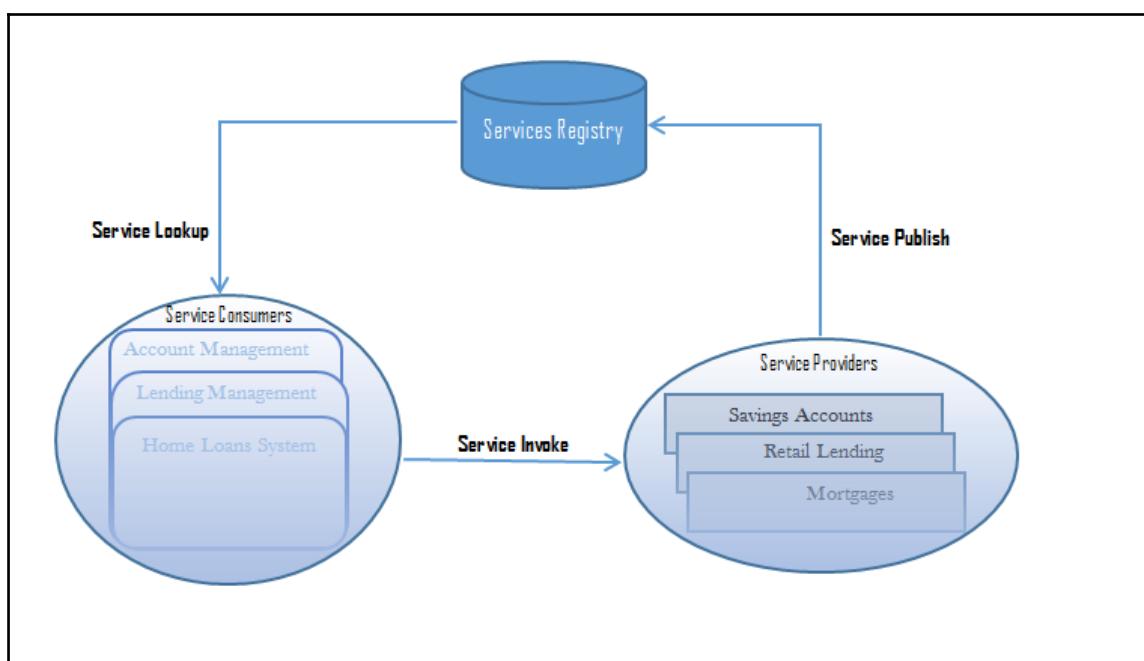
<soapenv:Envelope soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    .....
    .....
    <soapenv:Header>
        <con:StartHeader soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
            soapenv:mustUnderstand="0"
            xmlns:con="http://www.openuri.org/2002/04/soap/conversation/">
            <con:conversationID>[123456]</con:conversationID>
            <con:callbackLocation>
                http://myServer/myAppNotifyCallback
            </con:callbackLocation>
        </con:StartHeader>
    </soapenv:Header>
    <soapenv:Body>
        .....
        .....
        .....
    </soapenv:Body>
</soapenv:Envelope>
    
```

A red oval highlights the `<con:callbackLocation>` element, which contains the URL `http://myServer/myAppNotifyCallback`.



```
.....
<edms:events>
    <edms:event name="StockOrderEvent" uid="StockOrderEventUID" version="1.0">
        <edms:header name="cbe" element="cbe:CommonBaseEvent" use="optional"/>
        <edms:body name="order" element="exs:Order" use="required"/>
        <edms:body name="details" element="exs:ProcessingDetails" use="optional"/>
        <edms:fault name="orderFault" element="exs:OrderFault"/>
        <edms:fault name="processingFault" element="exs:OrderProcessingFault"/>
    </edms:event>
</edms:events>
.....
```





```
<Catalog ....>
```

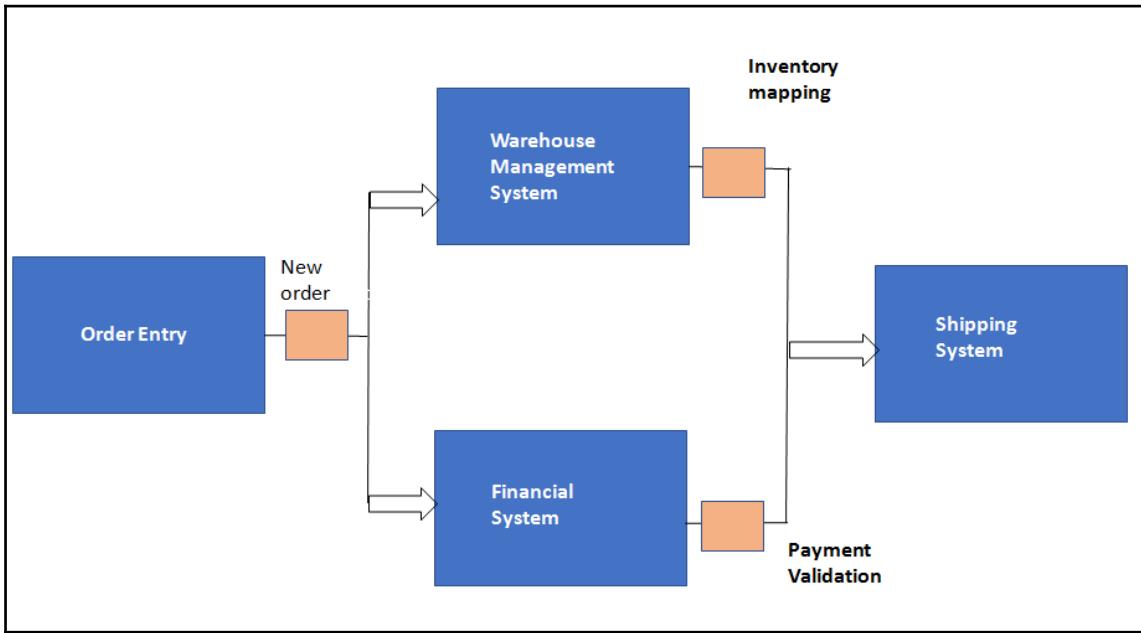
```
  <system systemId="http://sample.org/sb?wsdl" uri="SBSERVICE.wsdl" />
```

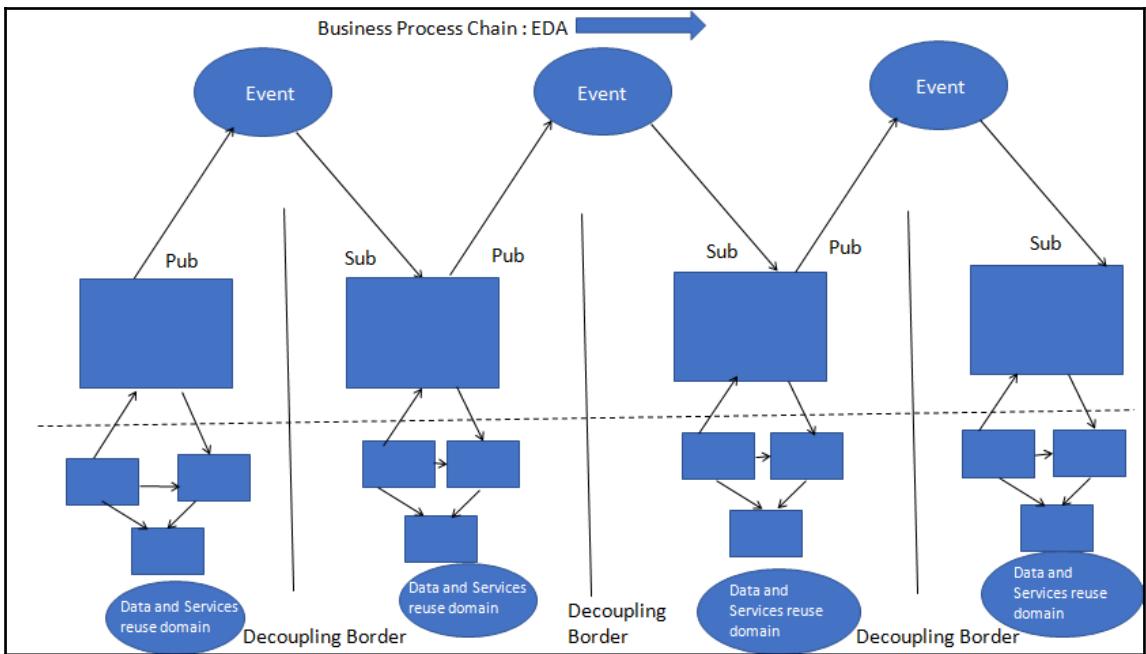
```
  <public publicId="....." uri="wsdl/sb/sbApp.wsdl"/>
```

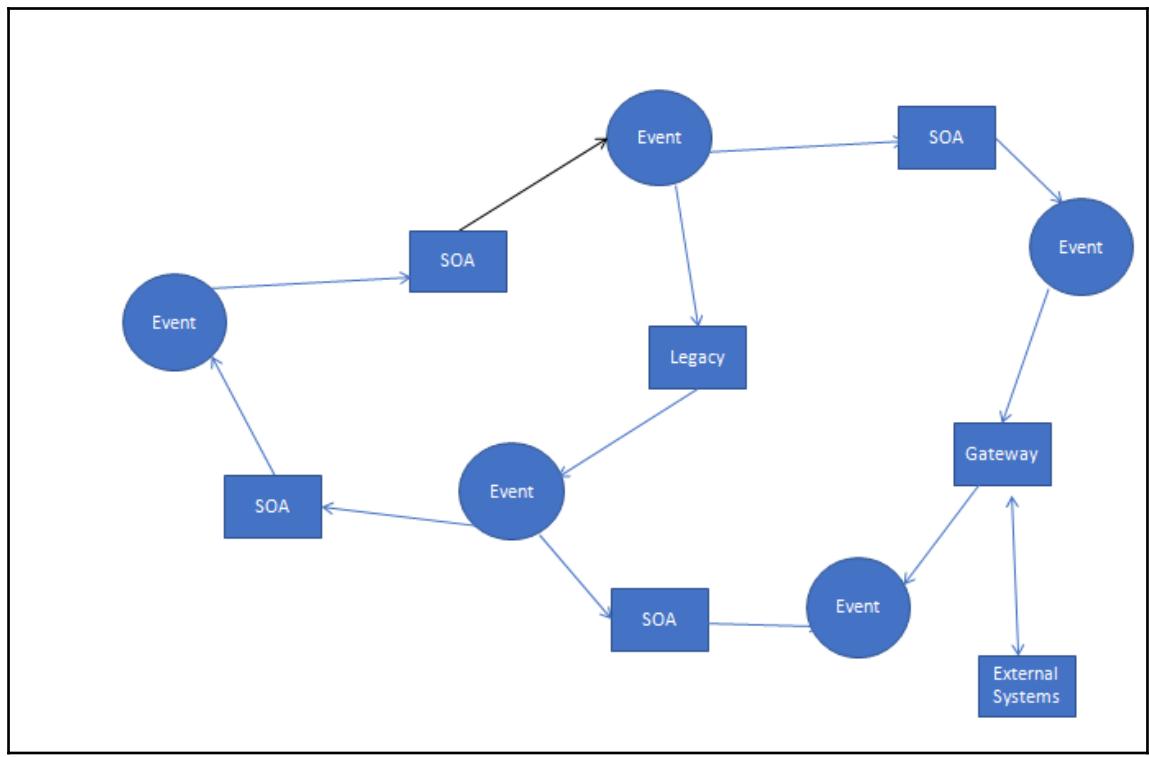
```
</Catalog>
```

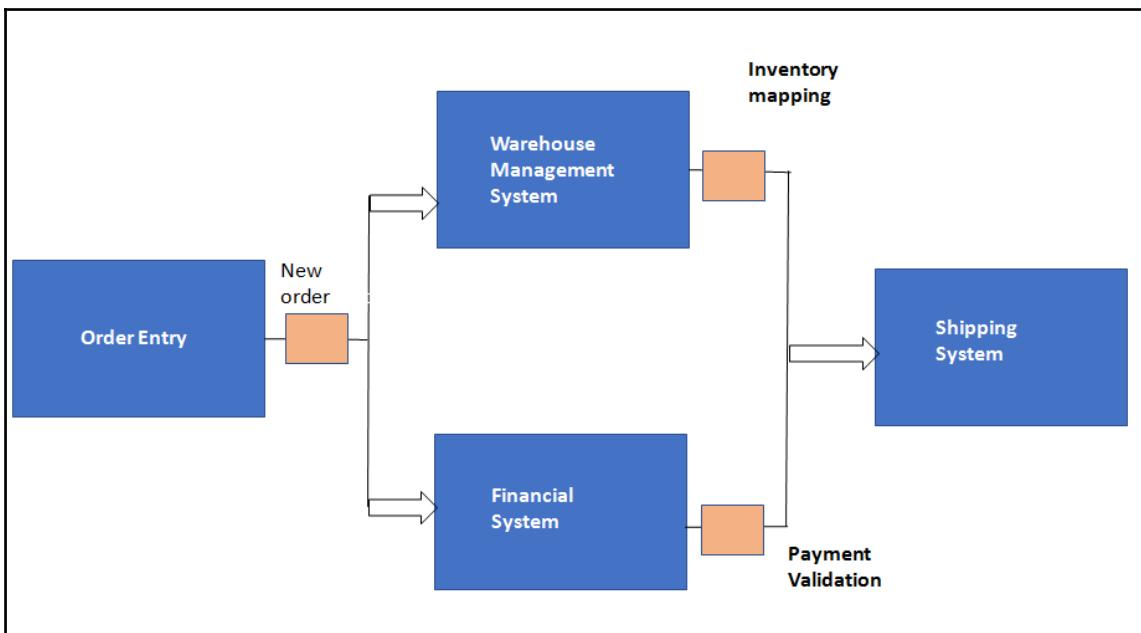
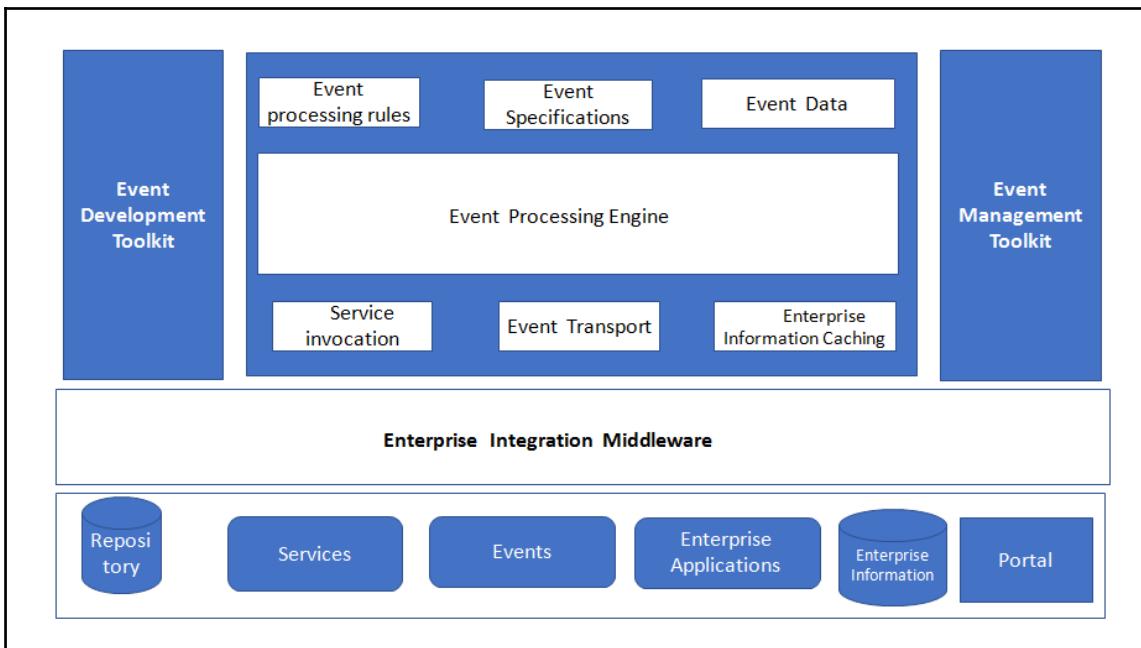
```
<businessEntity businessKey="...." operator=".../uddi" authorizedName=" ... ">
<discoveryURLs> <discoveryURL useType="businessEntity">.....</discoveryURL> </discoveryURLs>
    <name>XMethods</name>
    <description xml:lang="en">service resources </description>
<businessServices> <businessService serviceKey="...." businessKey="....">
    <name>Stock Quotes</name>
    <description xml:lang="en">stock quotes</description>
    <bindingTemplates> <bindingTemplate bindingKey=".." serviceKey="..">
        <description xml:lang="en">stock quote binding </description>
        <accessPoint URLType="http">http://.../soap </accessPoint>
    </bindingTemplate> </bindingTemplates>
</businessService> </businessServices>
</businessEntity>
```

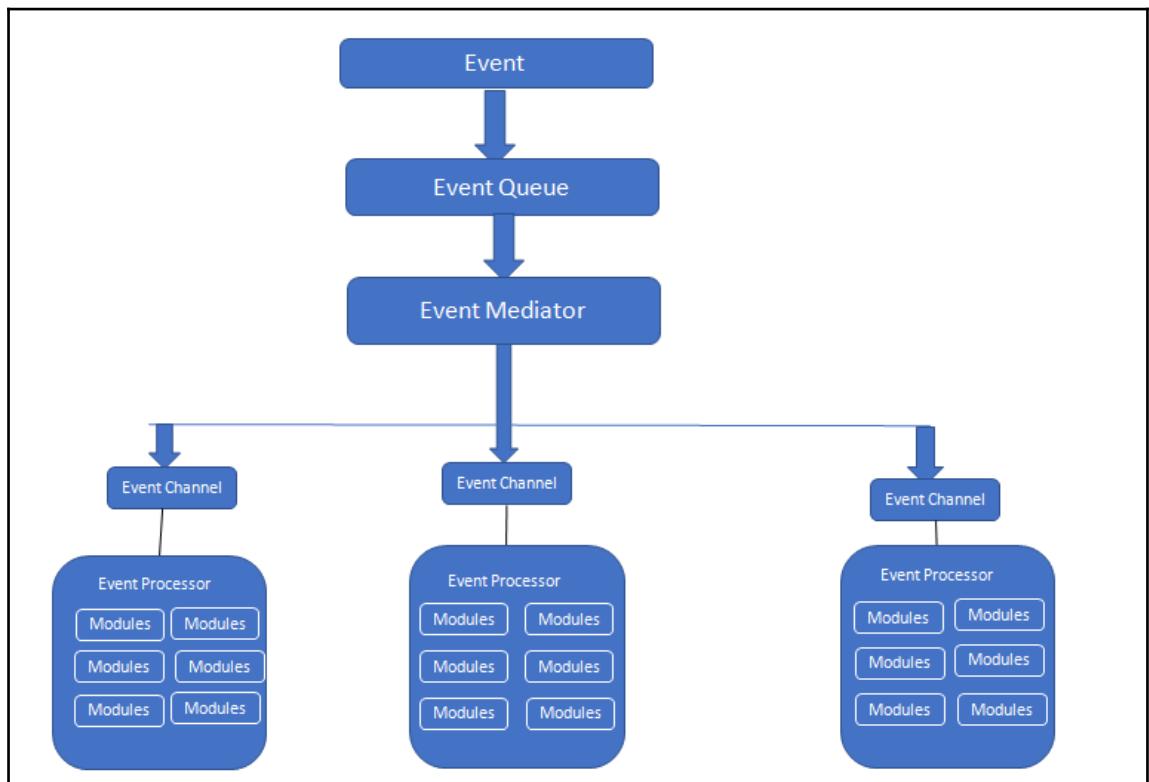
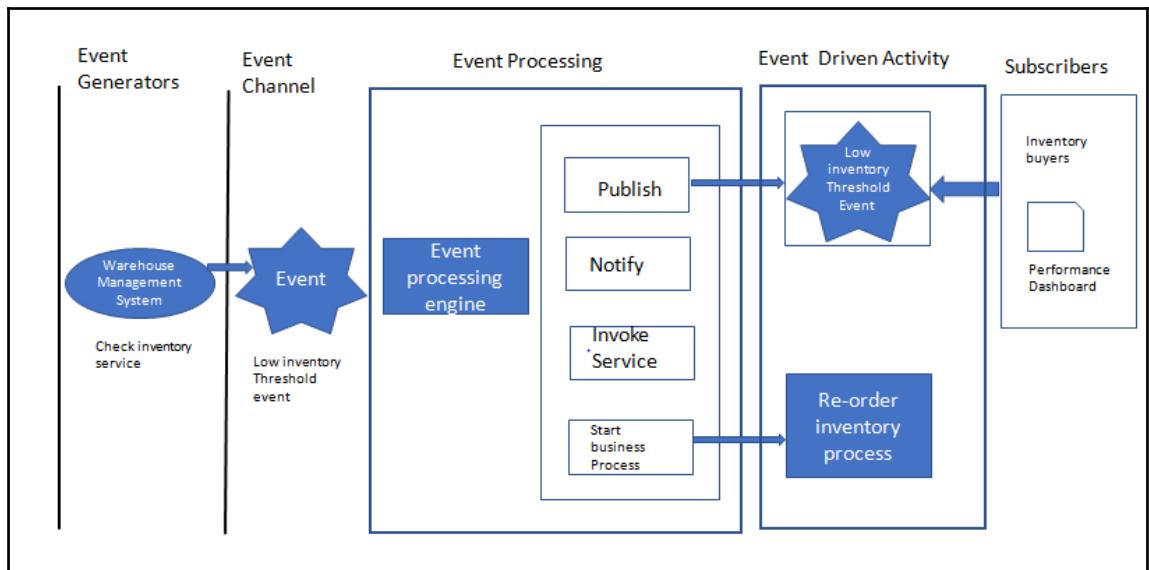
Chapter 8 - Event-Driven Architectural Patterns

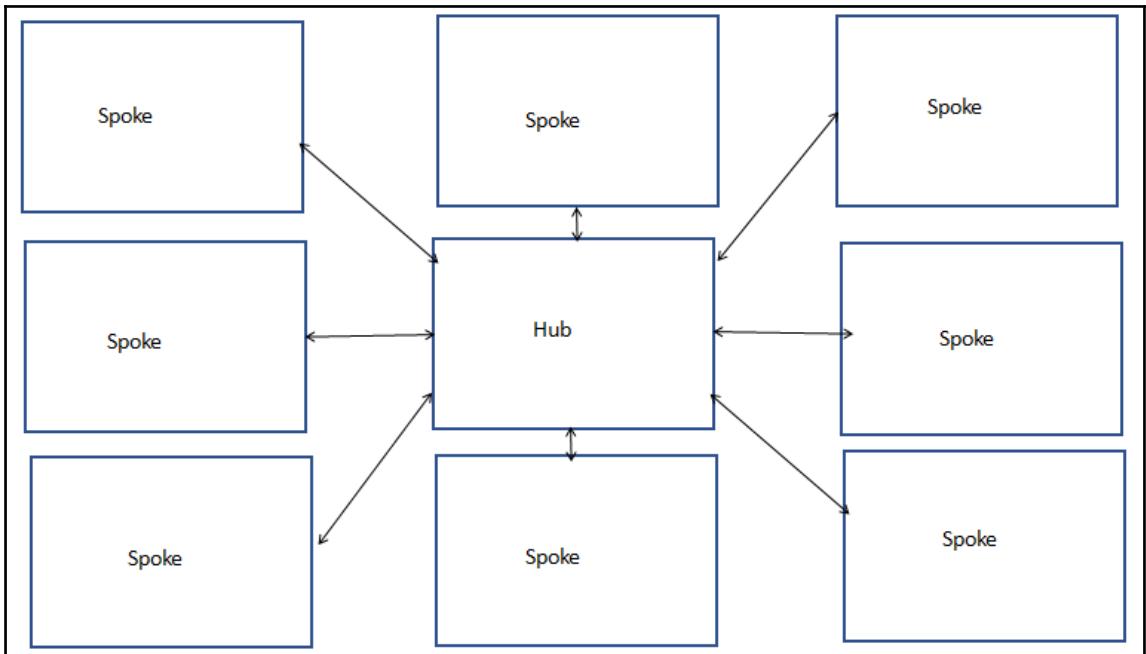
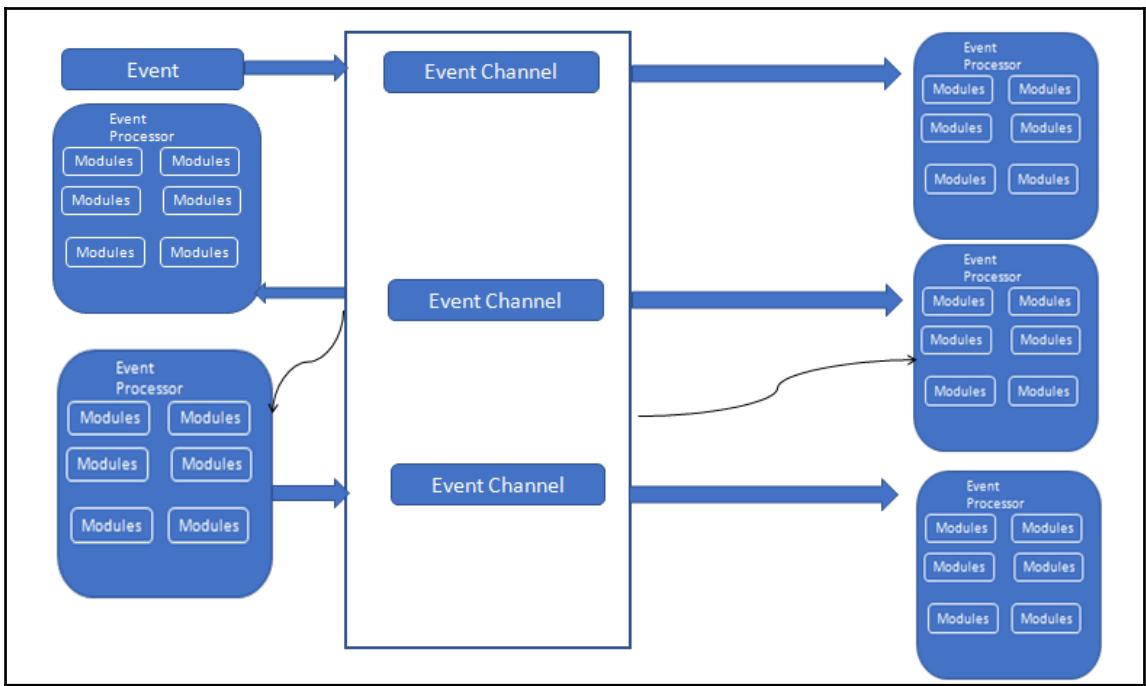


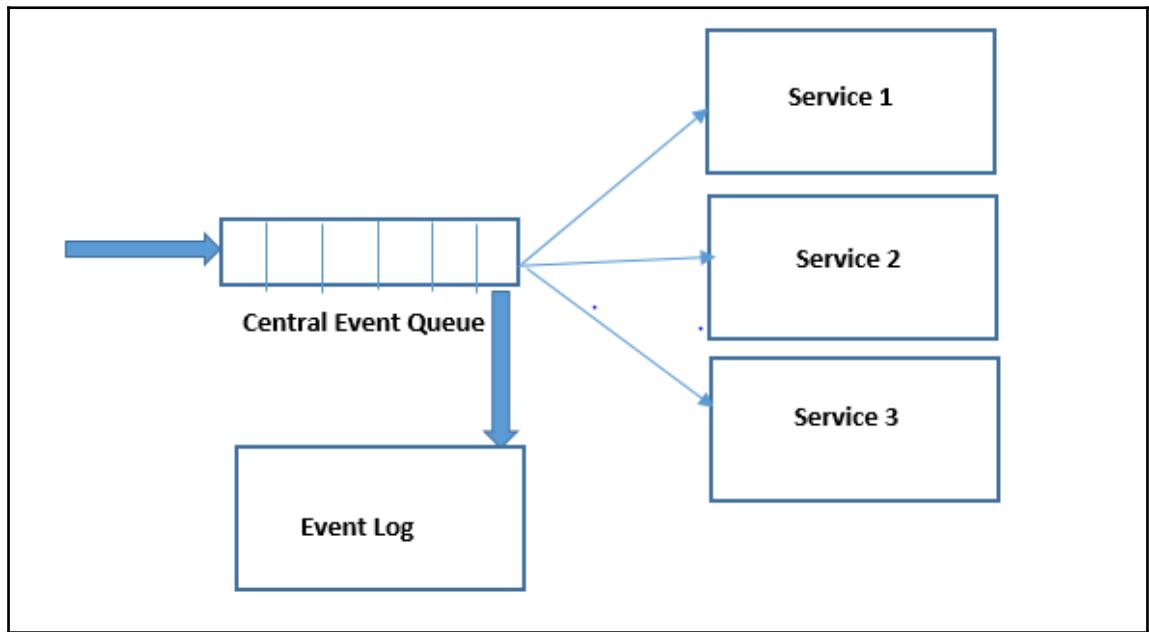
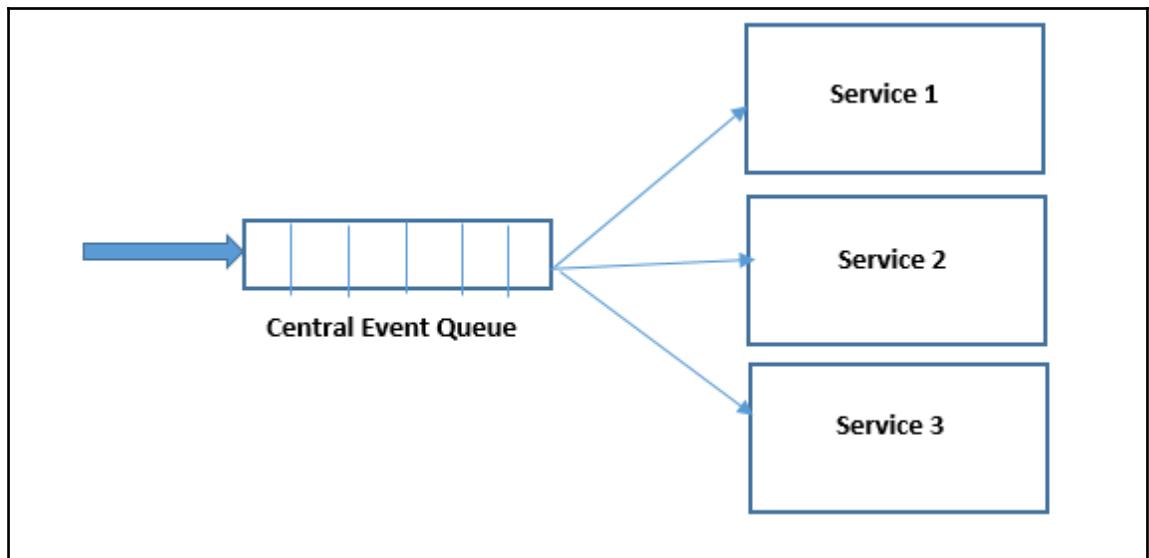


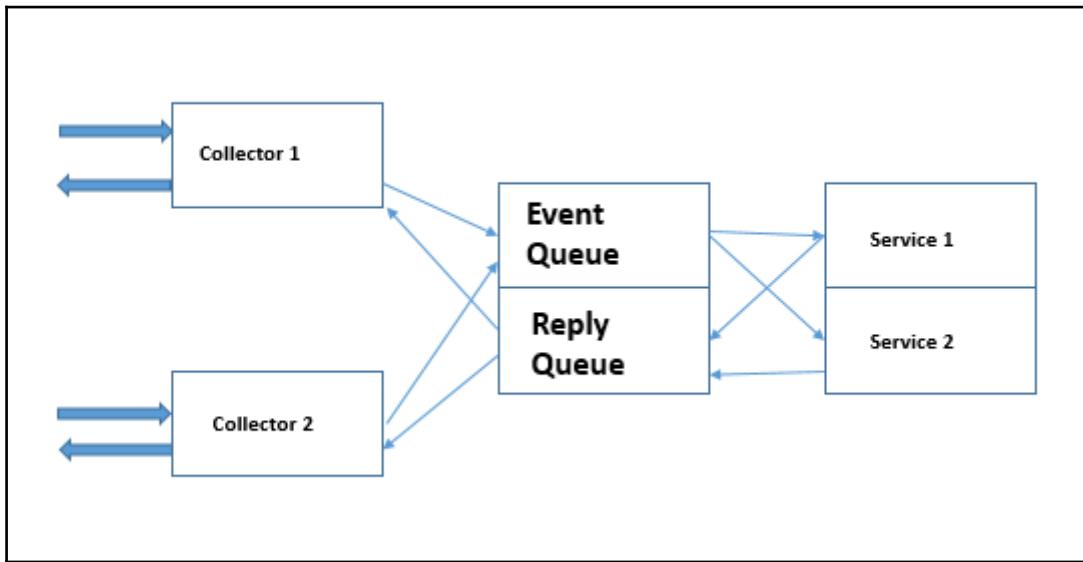
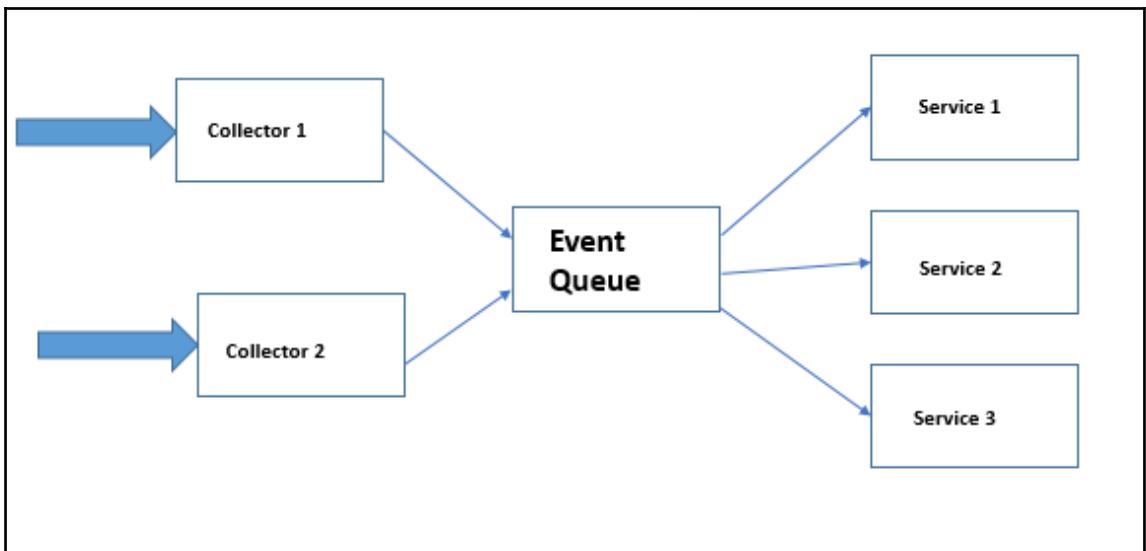


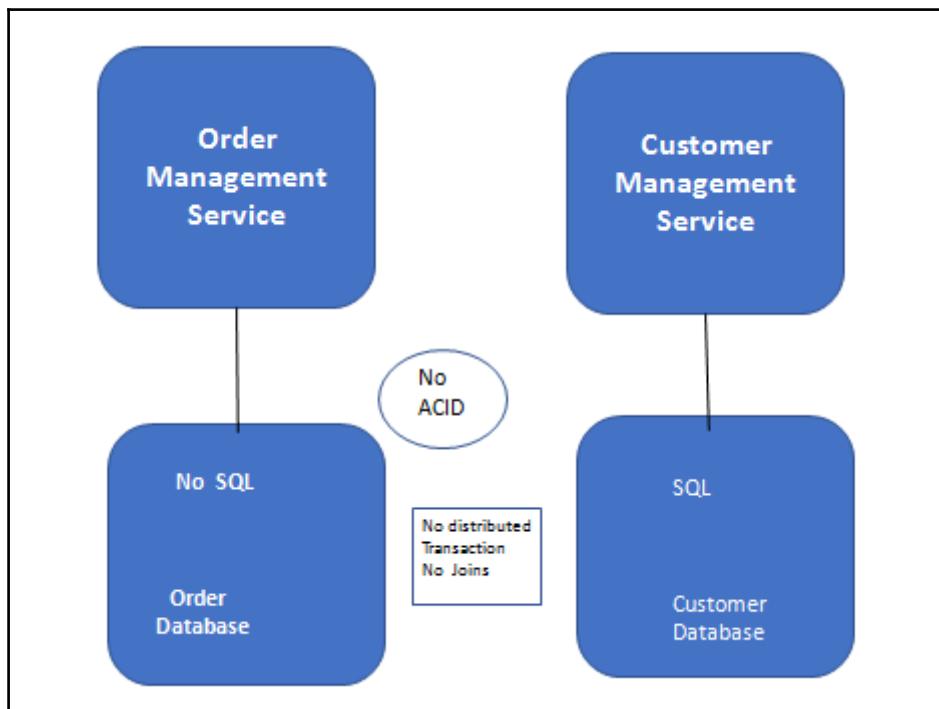
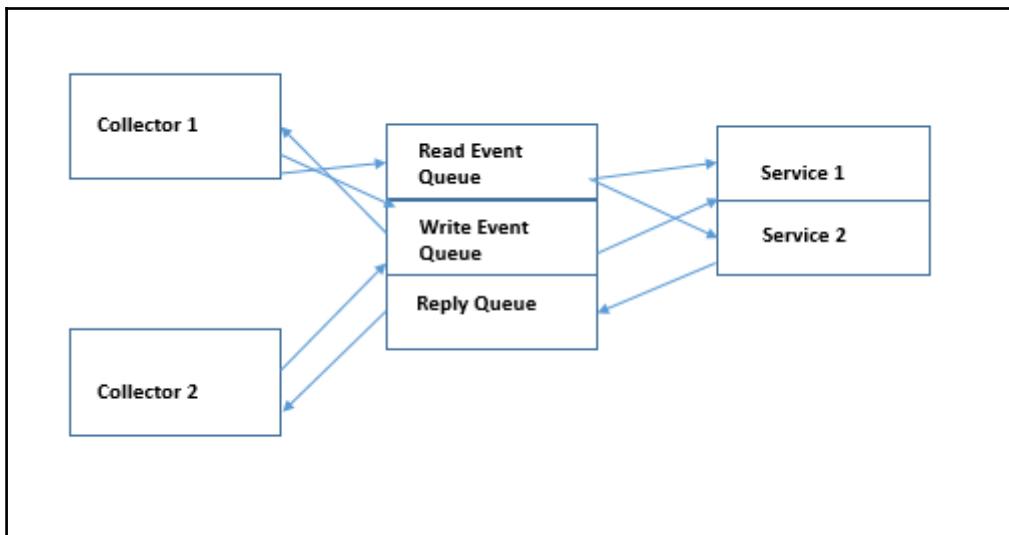


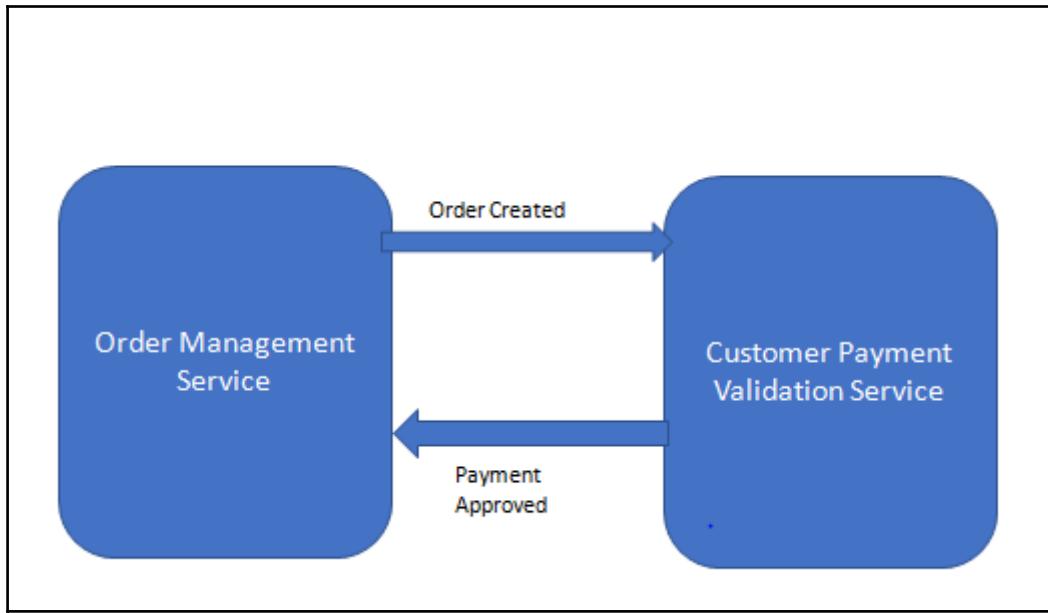




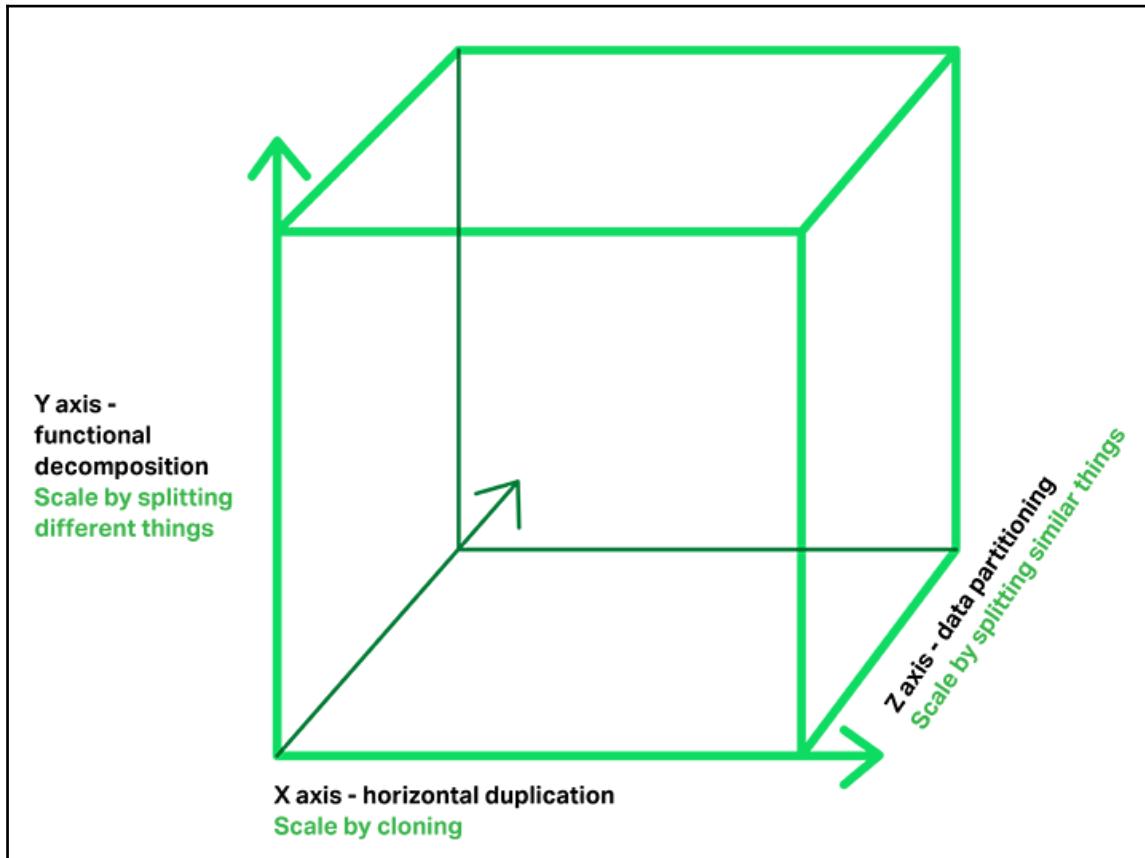


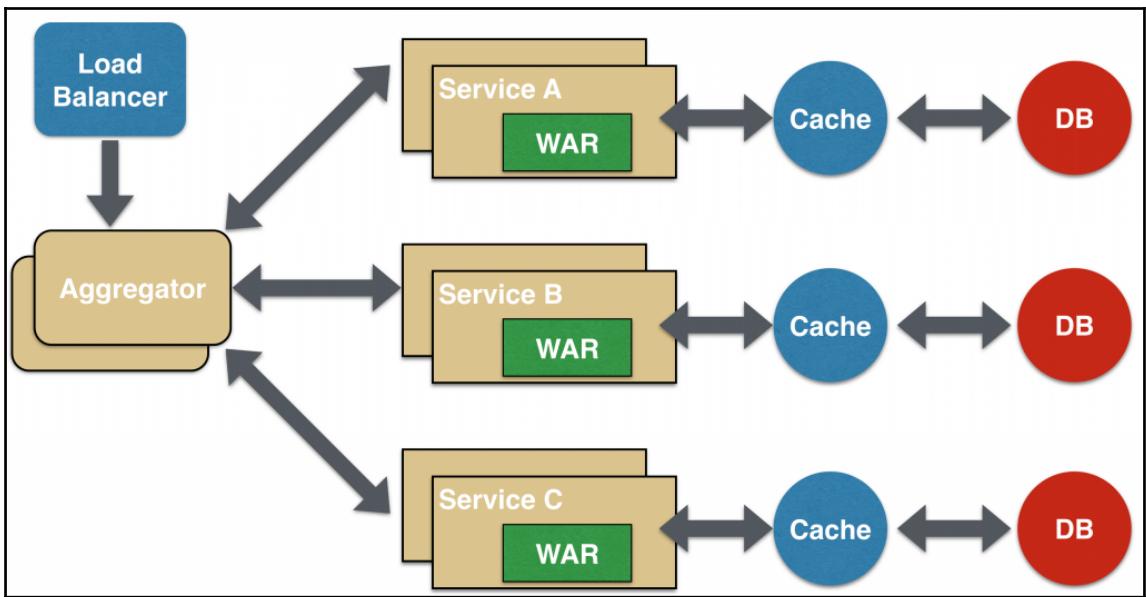




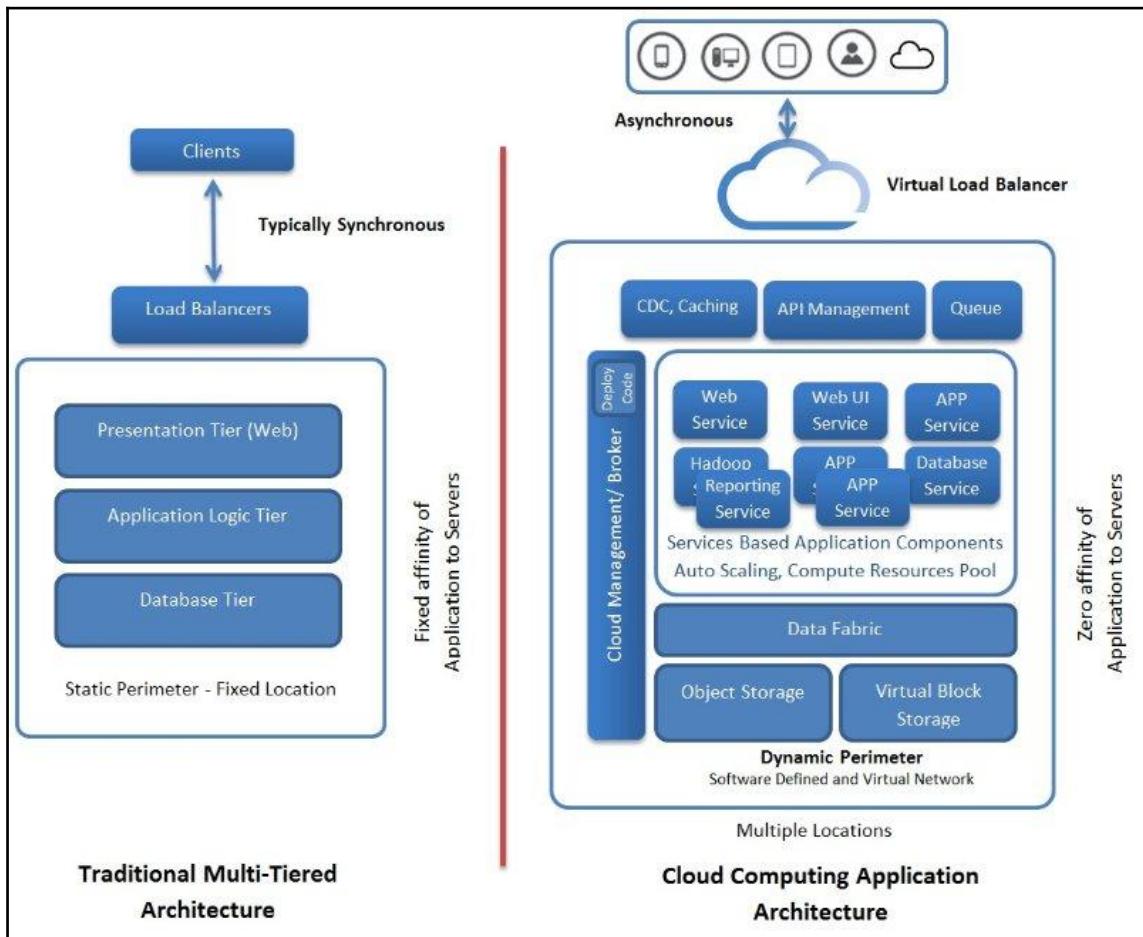


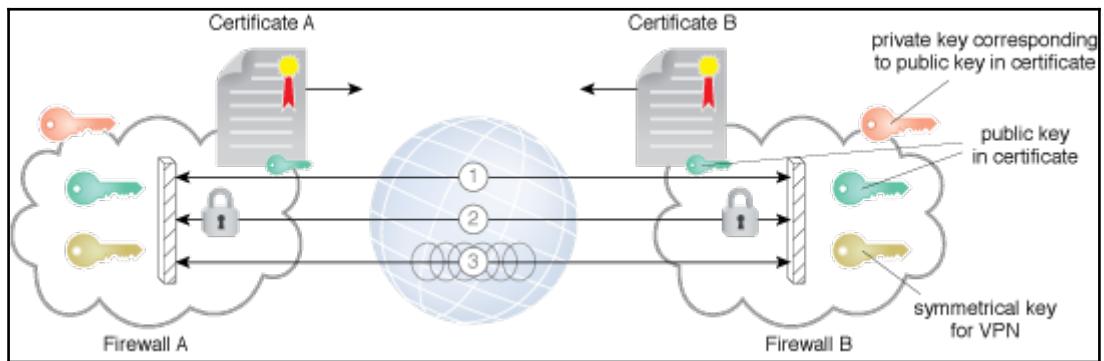
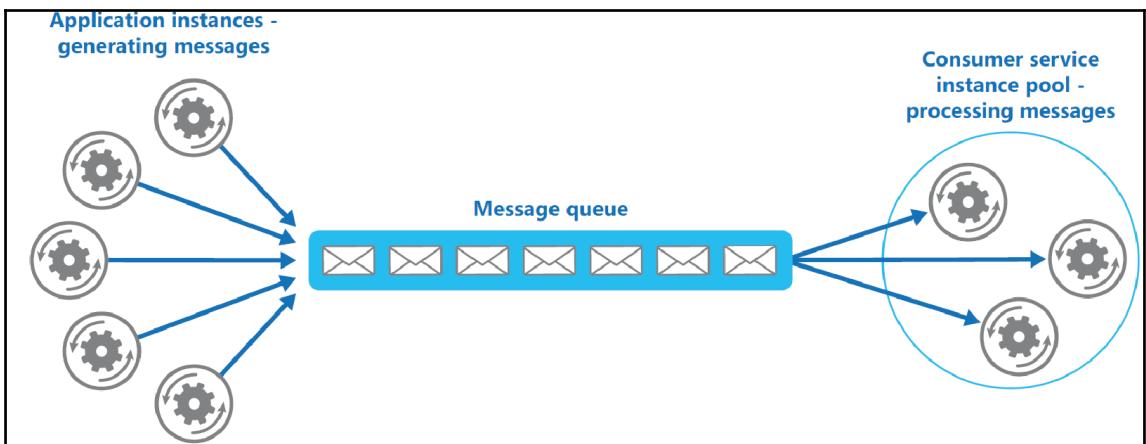
Chapter 9 - Microservices Architecture Patterns



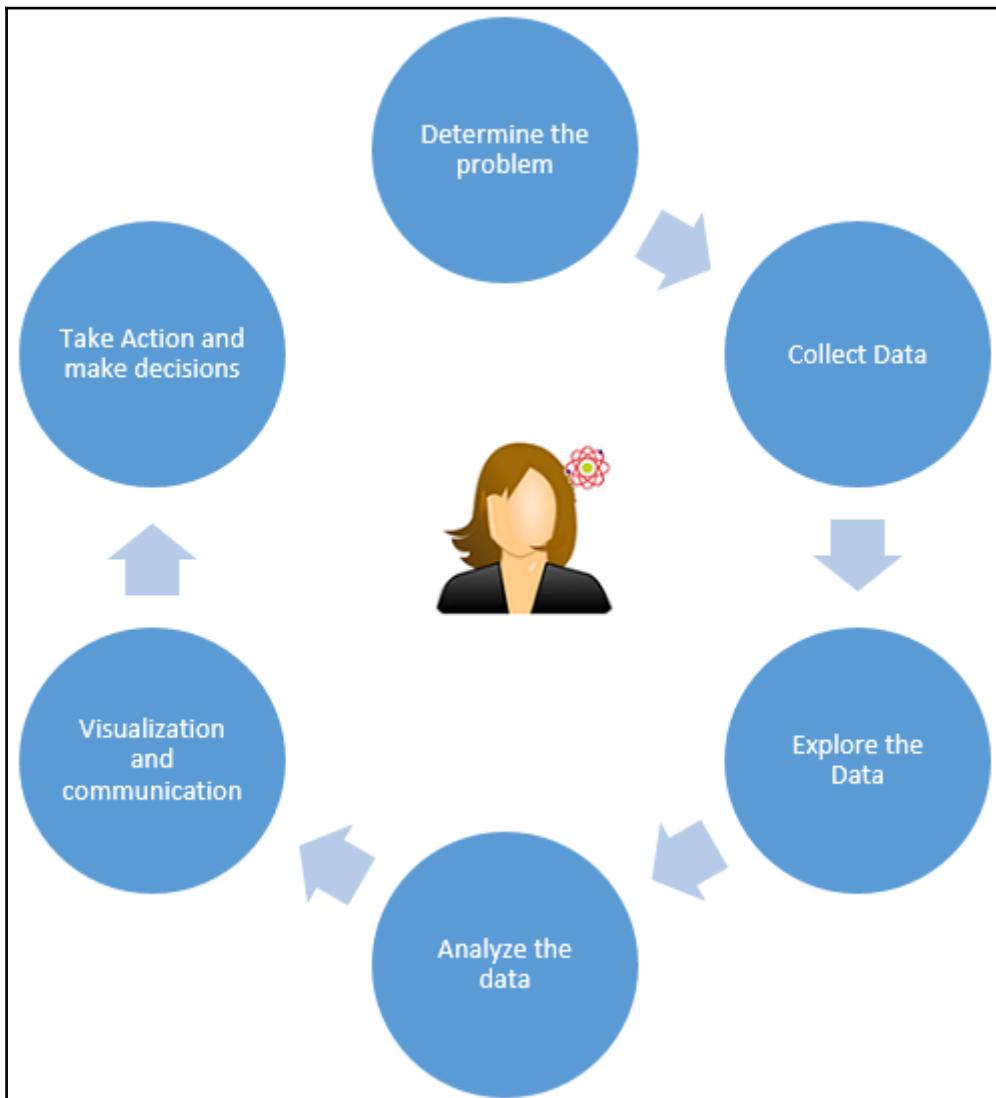


Chapter 11 - Software-Defined Clouds - the Architecture and Design Patterns





Chapter 12 - Big Data Architecture and Design Patterns



Hadoop / HDFS

Integration	Analysis	Visualization & Optimization	Security and Governance
Incorporate Discrete parts as a whole	Record and analyze the behavior of users	Present analyzed data in a simplified, graphical and statistical representations	Collection of data, retention policies, usages of data, data disclosures, privacy management, strict control policies, roles based access, Data protection and so on.

Splunk / Pentaho / Karmasphere / Talend / Tableau
and so on.

