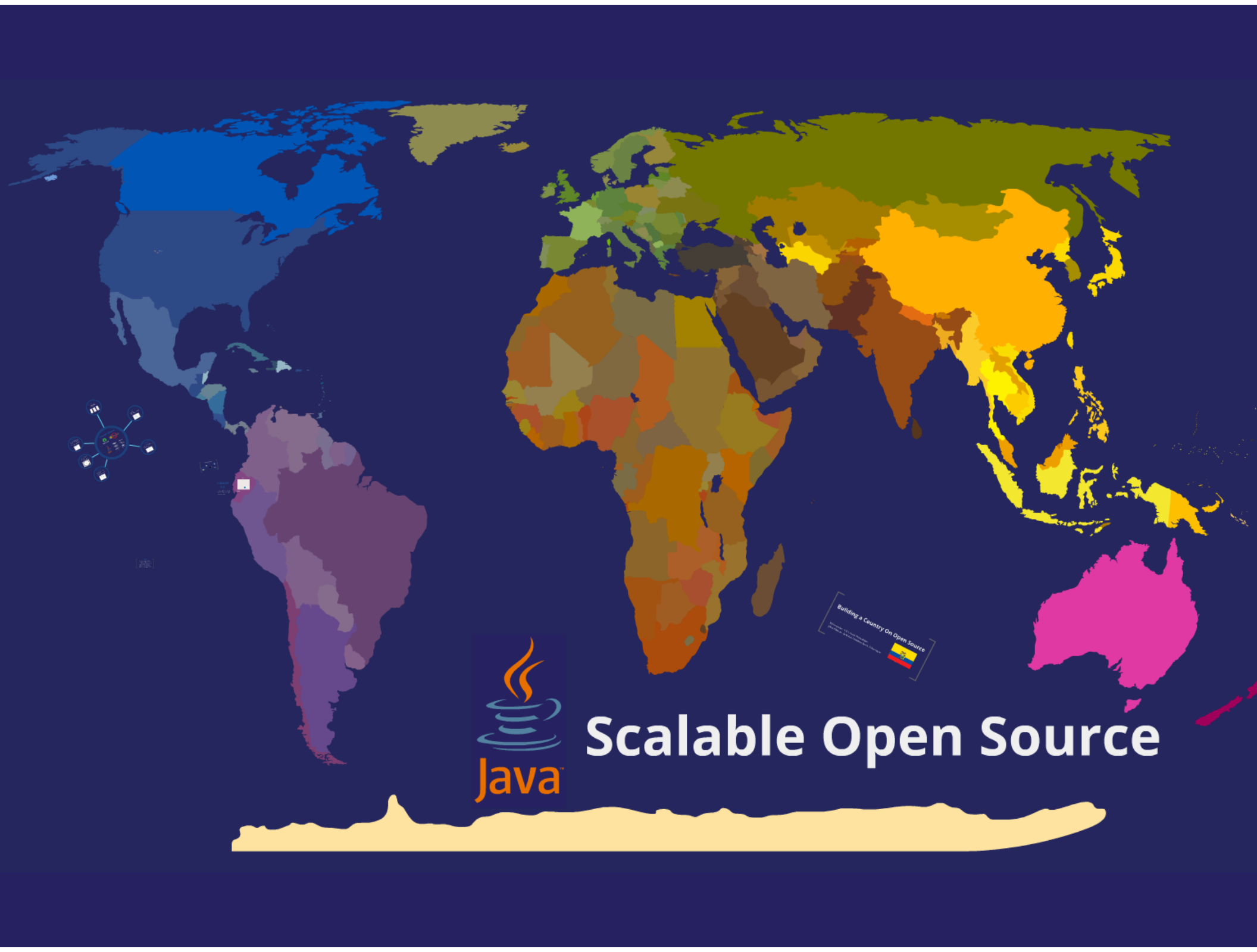
Building a Country On Open Source

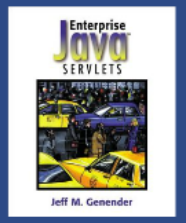
Jeff Genender - CTO, Savoir Technologies
Johan Edstrom - SOA Exec/Architect, Savoir Technologies

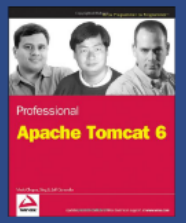

Building a Country On Open Source
Java
Scalable Open Source

- Denver

Jeff Genender

- Founder and CTO savoir TECHNOLOGIES
- Book Author





- Java Champion
- JSR 342 - Java EE 7 Expert Group
- Apache Member and Committer
  - ServiceMix Founder
  - CXF
  - Mina
  - Camel
  - Geronimo
  - TomEE
- Hobby - FireFighter for Evergreen

Johan Edstrom

- SOA Architect and Lieutenant

savoir TECHNOLOGIES

- Book Author

- Apache Committer
  - ServiceMix
  - Camel

- Foodie, Great Chef, and Sous Vide Cook
  (a must to be followed on Google+)

Ecuador S.A.

- Approximately 16 Million People
- Currency Based on the US Dollar
- Democratic Nation

- Quito

- Guayaquil

- Quito

- Guayaquil

What Is Dato Seguro?

Secure Data

Cedula

Goal?

History

Rafael Correa - 2007

"We will be producers of technology, and not simple consumers. We will be the owners of the source codes, and we can develop many products that can, with cooperation of the others, can be very useful to public and private companies in the region.

For that, everyone must use free software. The Ecuadorian government has already established this as a governmental and state policy. This will be an important step in the integration and, why not say, for the liberation of Latin America."

2011/2012

datoseguro.gob.ec

Trust O/S?

Nobody got fired for buying...

Microsoft

What They Built v1.0

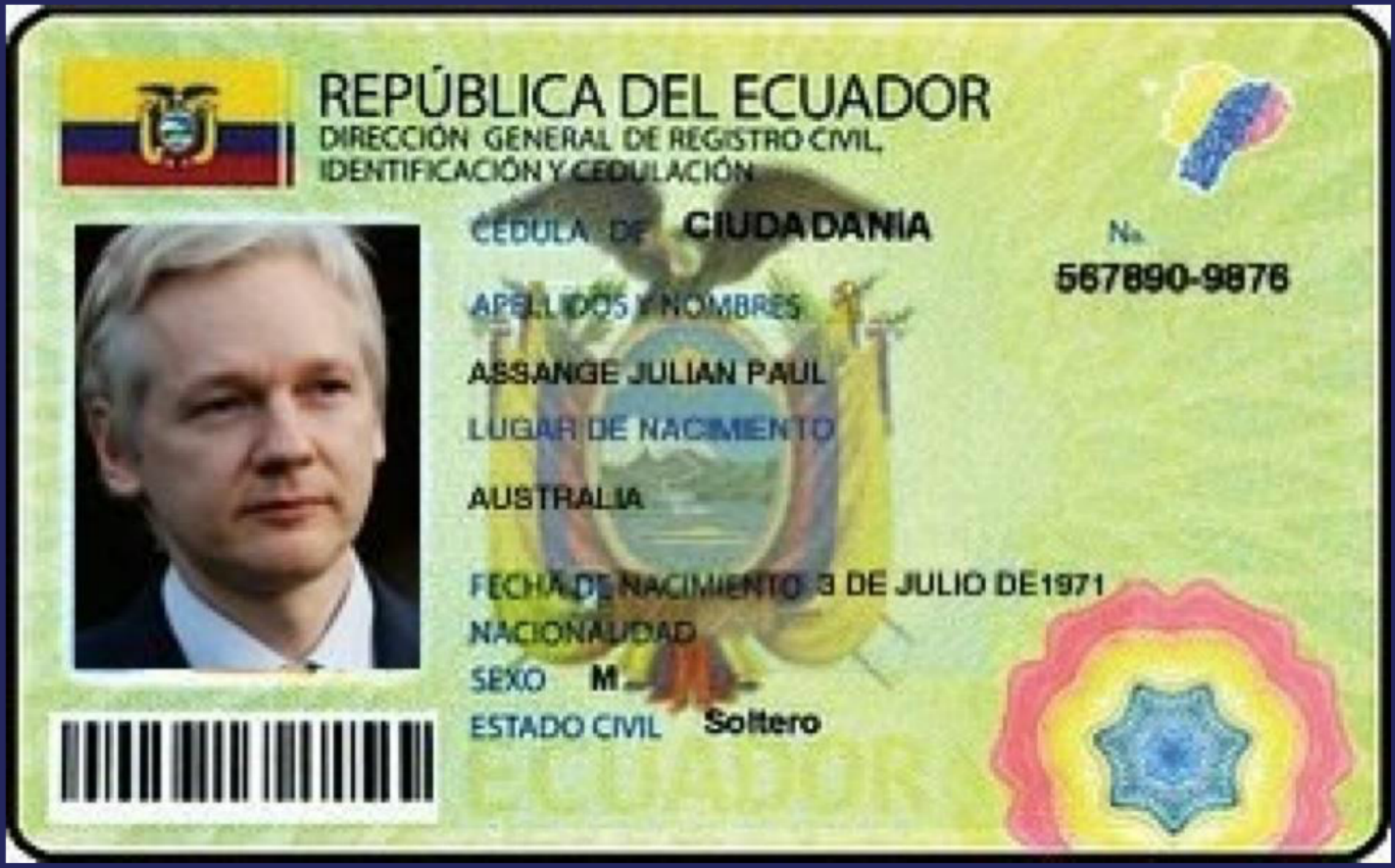
Problemas?

Licencia

Dato Seguro?

Secure Data

Cedula


REPÚBLICA DEL ECUADOR
DIRECCIÓN GENERAL DE REGISTRO CIVIL,
IDENTIFICACIÓN Y CEDULACIÓN
CÉDULA DE CIUDADANÍA
No.
567890-9876
APELLIDOS Y NOMBRES
ASSANGE JULIAN PAUL
LUGAR DE NACIMIENTO
AUSTRALIA
FECHA DE NACIMIENTO 3 DE JULIO DE1971
NACIONALIDAD
SEXO M
ESTADO CIVIL Soltero


Goal?


Registro Civil
Identificación y Cedulación
CNE
IESS
INCOP
INSTITUTO NACIONAL DE CONTRATACIÓN PÚBLICA
Ministerio de Relaciones Laborales
REGISTROS MERCANTILES
Registros de la Propiedad
SENESCYT
SRi
...le hace bien al país!
Agencia Nacional de Tránsito


Registro Civil
Identificación y Cedulación
CNE
IESS
INCOP
INSTITUTO NACIONAL DE CONTRATACIÓN PÚBLICA
Ministerio de Relaciones Laborales
REGISTROS MERCANTILES
Registros de la Propiedad
SENESCYT
SRi
...le hace bien al país!
Agencia Nacional de Tránsito

History

Rafael Correa - 2007

"We will be producers of technology, and not simple consumers. We will be the owners of the source codes, and we can develop many products that can, with cooperation of this effort, can be very useful to public and private companies in the region.

For that, everyone must use free software. The Ecuadoran government has already established this as a governmental and state policy. This will be an important step in the integration and, why not say, for the liberation of Latin America."

2011/2012

datoseguro .gob.ec

Trust O/S?

Nobody got fired for buying...

Microsoft

Rafael Correa - 2007

"We will b
consumers
and we can
cooperatio
private con

For that, ev
governmen
and state p
integration
America."

2011/2012

"We will be producers of technology, and not simple consumers. We will be the owners of the source codes, and we can develop many products that can, with cooperation of this effort, can be very useful to public and private companies in the region.

For that, everyone must use free software. The Ecuadoran government has already established this as a governmental and state policy. This will be an important step in the integration and, why not say, for the liberation of Latin America."

2011/2012

datoseguro.gob.ec

integration and, why not say, for the liberation of Latin America."

Trust O/S?

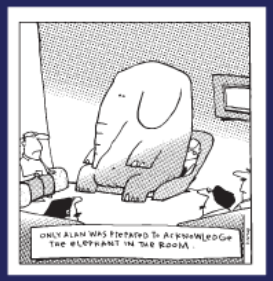
Nobody got fired for buying...

Microsoft

What They Built v1.0

SOAP WebServices

Problemas?



Licencia
(Licensing)

(In)Seguridad
((In) Security)

Arquitectura
(Architecture)

- Point to point - No message guarantee
- No failover capabilities - Infrastructure was unstable
- Not modular


ONLY ALAN WAS PREPARED TO ACKNOWLEDGE THE ELEPHANT IN THE ROOM.
harrop

Licencia

(Licensing)

(In)Seguridad

((In) Security)

DIARIO EL COMERCIO DEL ECUADOR **Policy**

Home | **News** | World | Quito | Sports | Opinion | Entertainment | Tech

POLICY | SECURITY | BUSINESS

News | Policy | Riobamba

Citizen arrested for 'bare the weaknesses' of Data Security

Photo of the detention of blogger Paul Moreno. Photo taken from the Twitter account @ pulpineitor

Photo of the detention of blogger Paul Moreno. Photo taken from the Twitter account @ pulpineitor

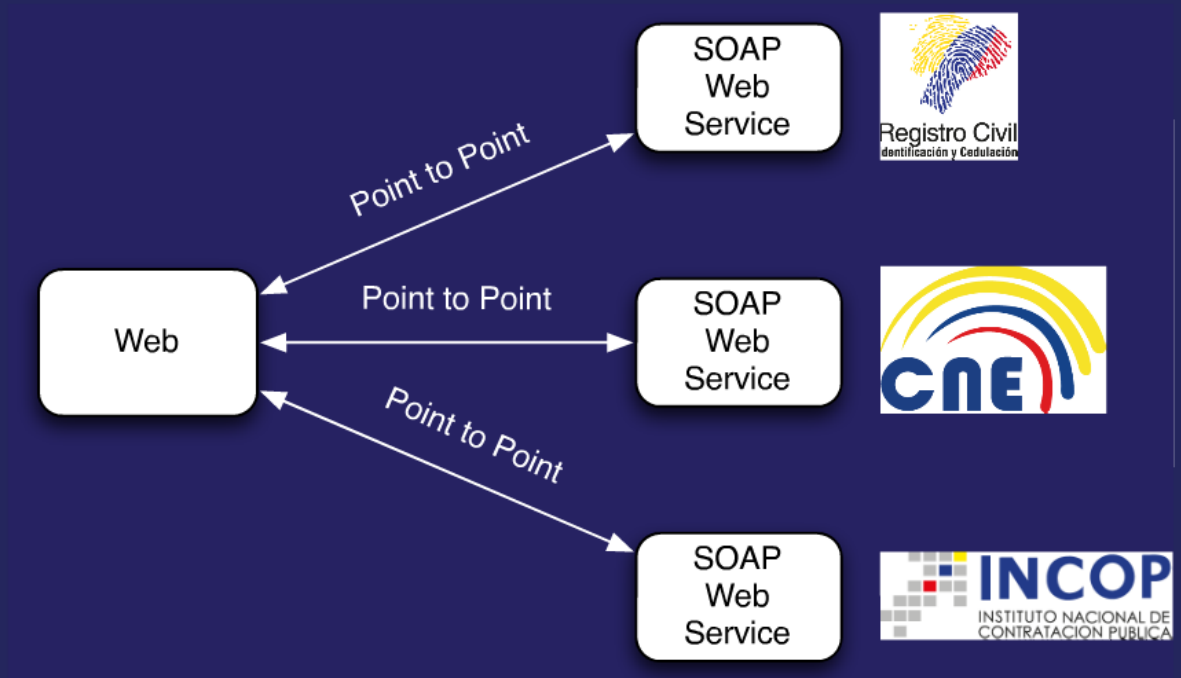
READING TIME: 3 '25" NO. OF WORDS: 571

Policy Drafting Saturday 01/12/2012

Access to information is Ecuadorian a click away. **Paul Moreno** , a blogger of **Riobamba** , he proved easily obtainable data **President** . Now in jail for it.

Yesterday, at 09:00, the police arrested the man, 42, for writing on his blog how he freely accessed personal information of the Head of State, which rests on the website **www.datoseguro.gob.ec.**

Arquitectura
(Architecture)



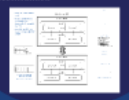
- Point to point - No message guarantee
- No failover capabilities - Infrastructure was unstable
- Not modular

Scalable Solution

La Solución

Apache Cassandra

- NoSQL Database that is Java-based
- Allowed system to have fault tolerant data access
- Extremely high performant and has cross-datacenter capabilitites and its decentralized



ServiceMix

- Apache Karaf
- OSGi Container - Provides Modularity
- Apache Camel and CXF Bundles

ActiveMQ

- Java Messaging Service (JMS)
- Persistent Messaging
- Removes Point To Point / Consumer Polling Strategy

Apache Camel (Adaptors)

- Each "company" has its own service
- Lives at its respective data center close to the "company"
- Mediates between SOAP and the back-end company datasink

Apache Camel (Routes)

- Used to mediate the JAXRS calls
- Ensured the calls went through, even on adaptor service failure

Scalable Solution

Apache ServiceMix

- Modular Deployments / OSGi
- Enterprise Service Bus - ESB

karaf

- Modular Deployments / OSGi
- Division of Labor

Apache

ActiveMQ

- Guaranteed Messaging/Communication
- Persistent Messaging
- Event and consumer based poller
- Removes Point To Point

Apache Camel

- Data / Communication Routing
- Data Transformation
- Endpoint container

Apache CXF

- Webservice Container
- Used SOAP/JAX-WS for external communication
- JAX-RS for internal ESB communication

- Fault Tolerant Data
- Used as a cache for service queries as a backup

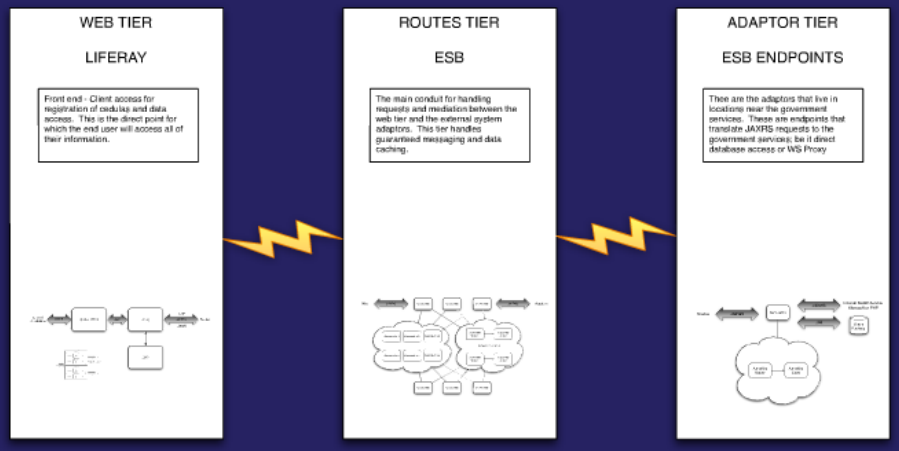
La Solución

(The Solution)

DIRECCIÓN NACIONAL DE REGISTRO DE DATOS PÚBLICOS

DINARDAP

SINARDAP - SISTEMA NACIONAL DE DATOS PÚBLICOS



La Solución

(The Solution)

latinus
The IT Company

DIRECCIÓN NACIONAL DE REGISTRO DE DATOS PÚBLICOS

DINARDAP

SINARDAP - SISTEMA NACIONAL DE DATOS PÚBLICOS

WEB TIER	ROUTES TIER	ADAPTOR TIER
LIFERAY	ESB	ESB ENDPOINTS

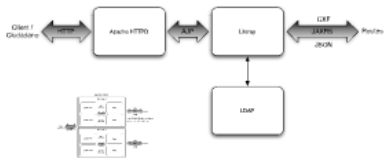
DINARDAP

SINARDAP - SISTEMA NACIONAL DE DATOS PÚBLICOS

WEB TIER

LIFERAY

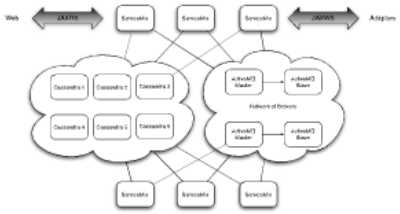
Front end - Client access for registration of cedulas and data access. This is the direct point for which the end user will access all of their information.



ROUTES TIER

ESB

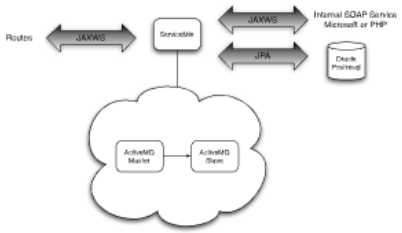
The main conduit for handling requests and mediation between the web tier and the external system adaptors. This tier handles guaranteed messaging and data caching.



ADAPTOR TIER

ESB ENDPOINTS

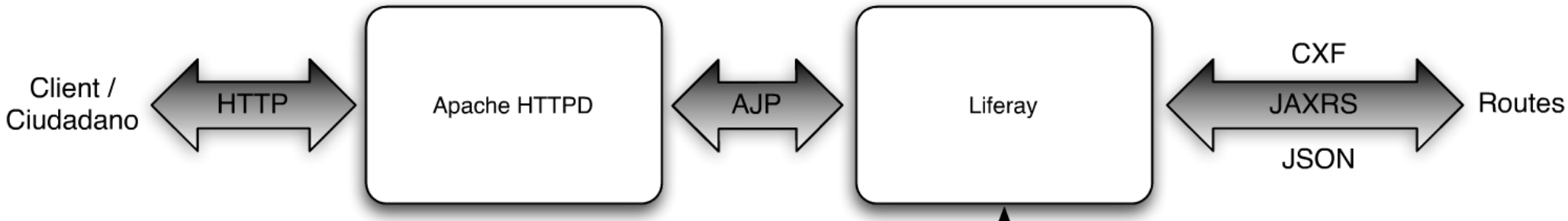
Thee are the adaptors that live in locations near the government services. These are endpoints that translate JAXRS requests to the government services; be it direct database access or WS Proxy

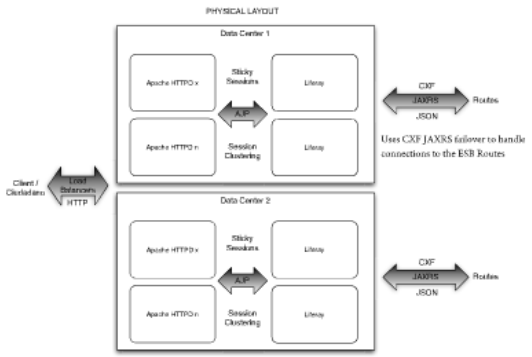


WEB TIER

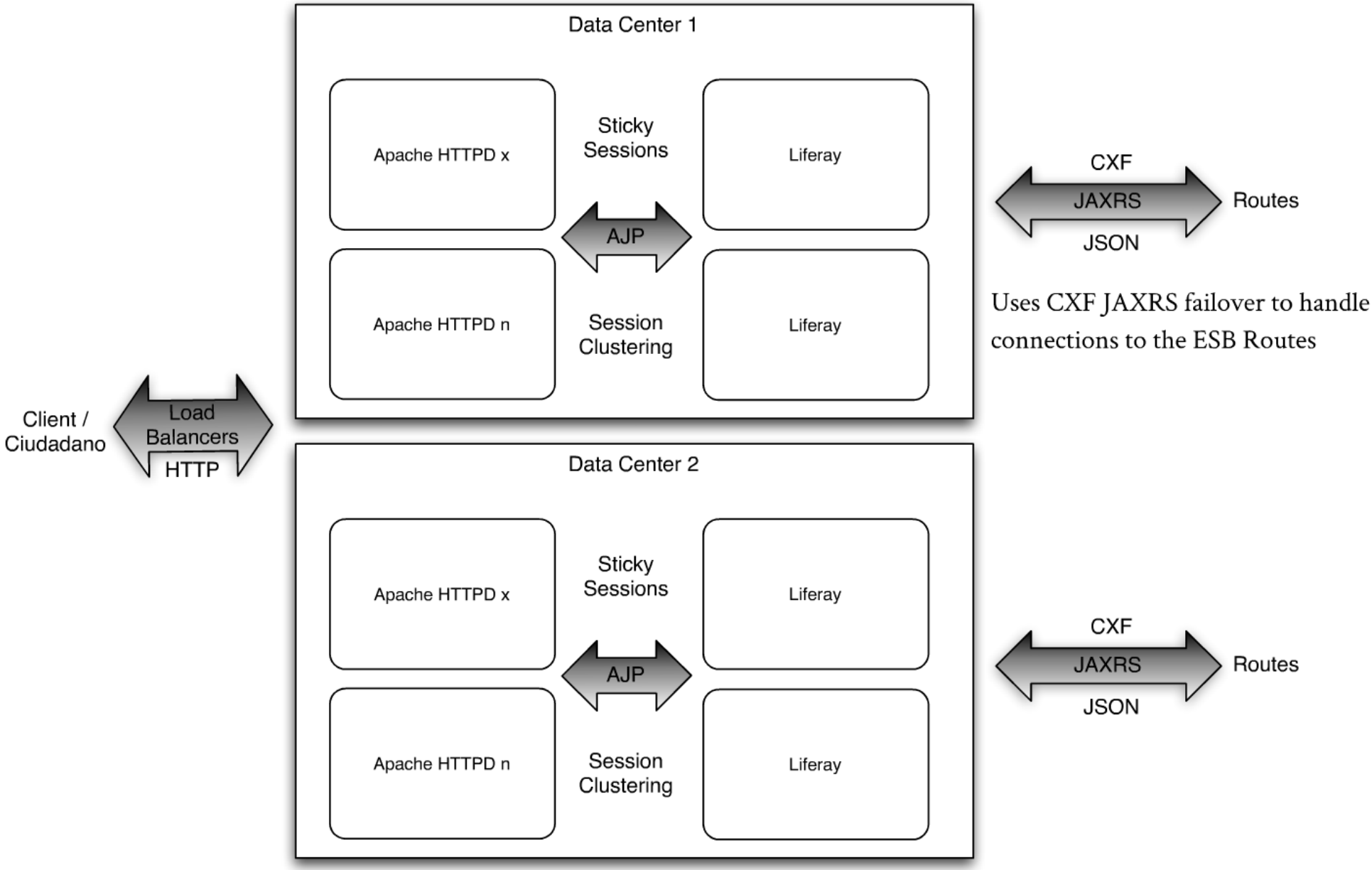
LIFERAY

Front end - Client access for registration of cedulas and data access. This is the direct point for which the end user will access all of their information.


Client / Ciudadano
HTTP
Apache HTTPD
AJP
Liferay
CXF
JAXRS
JSON
Routes
LDAP


PHYSICAL LAYOUT
Data Center 1
Apache HTTPD x
Sticky Sessions
Liferay
AJP
Apache HTTPD n
Session Clustering
Liferay
CXF
JAXRS
JSON
Routes
Uses CXF JAXRS failover to handle connections to the ESB Routes
Client / Ciudadano
Load Balancers
HTTP
Data Center 2
Apache HTTPD x
Sticky Sessions
Liferay
AJP
Apache HTTPD n
Session Clustering
Liferay
CXF
JAXRS
JSON
Routes

PHYSICAL LAYOUT

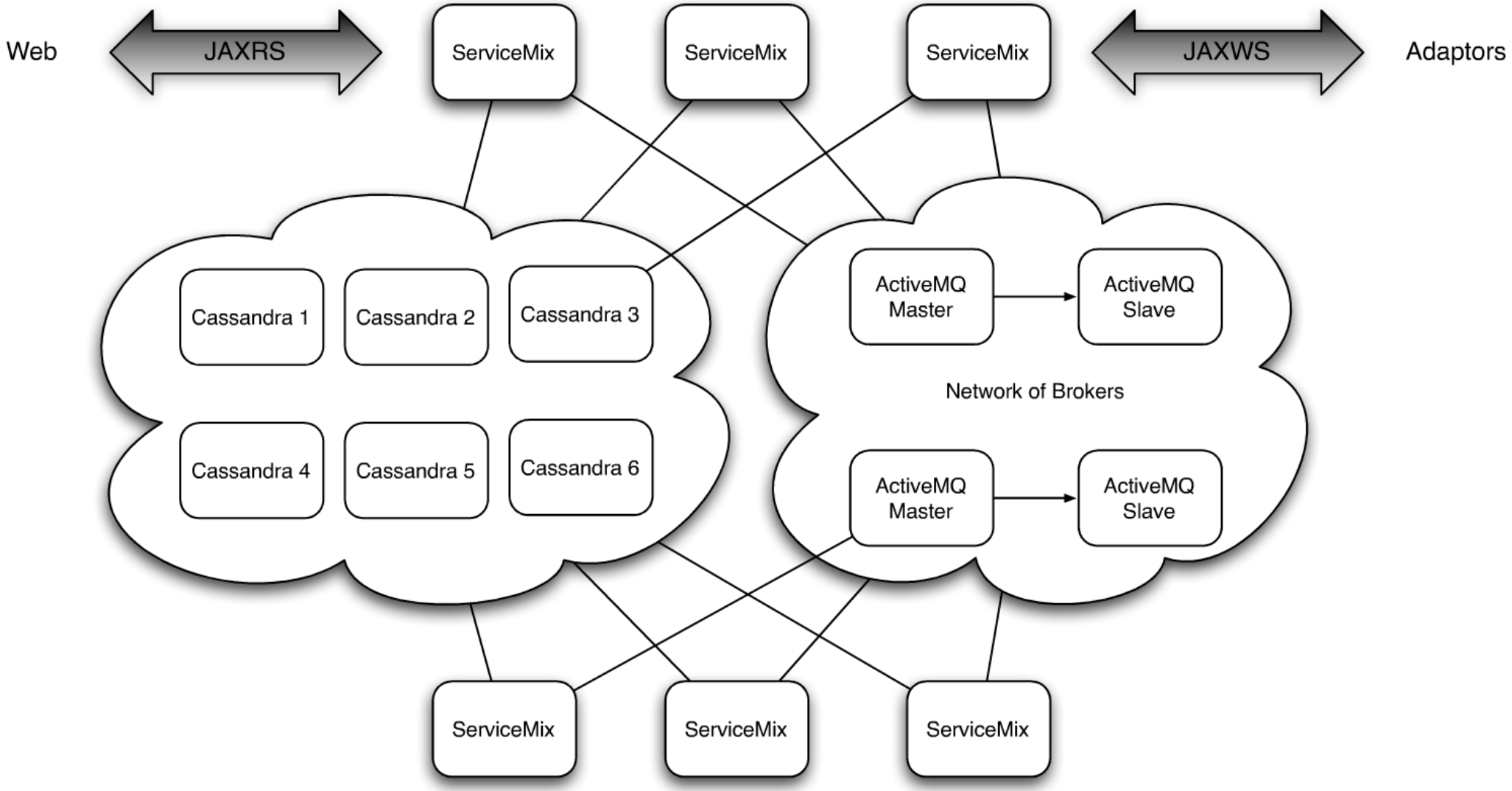


Uses CXF JAXRS failover to handle connections to the ESB Routes

ROUTES TIER

ESB

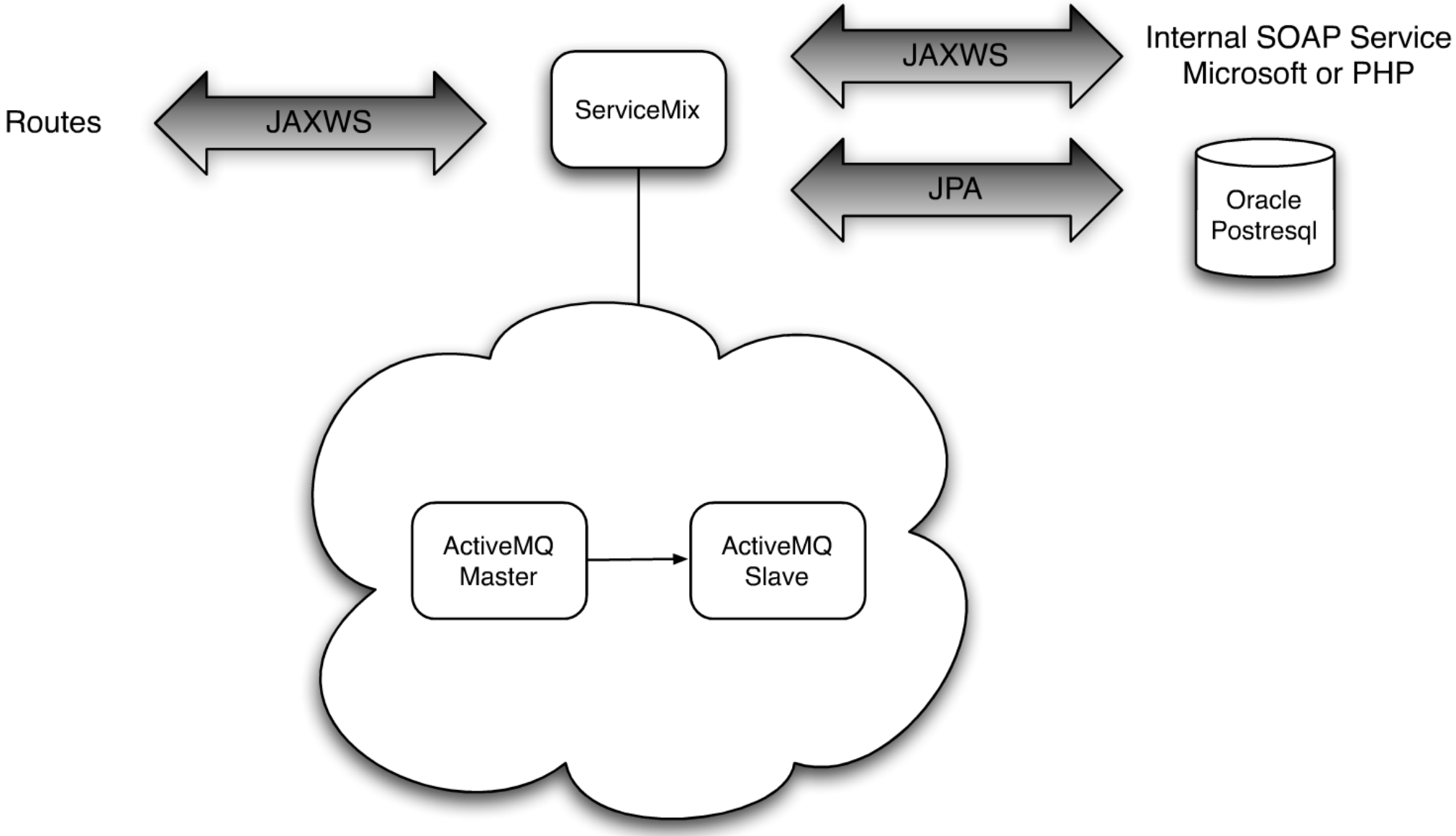
The main conduit for handling requests and mediation between the web tier and the external system adaptors. This tier handles guaranteed messaging and data caching.

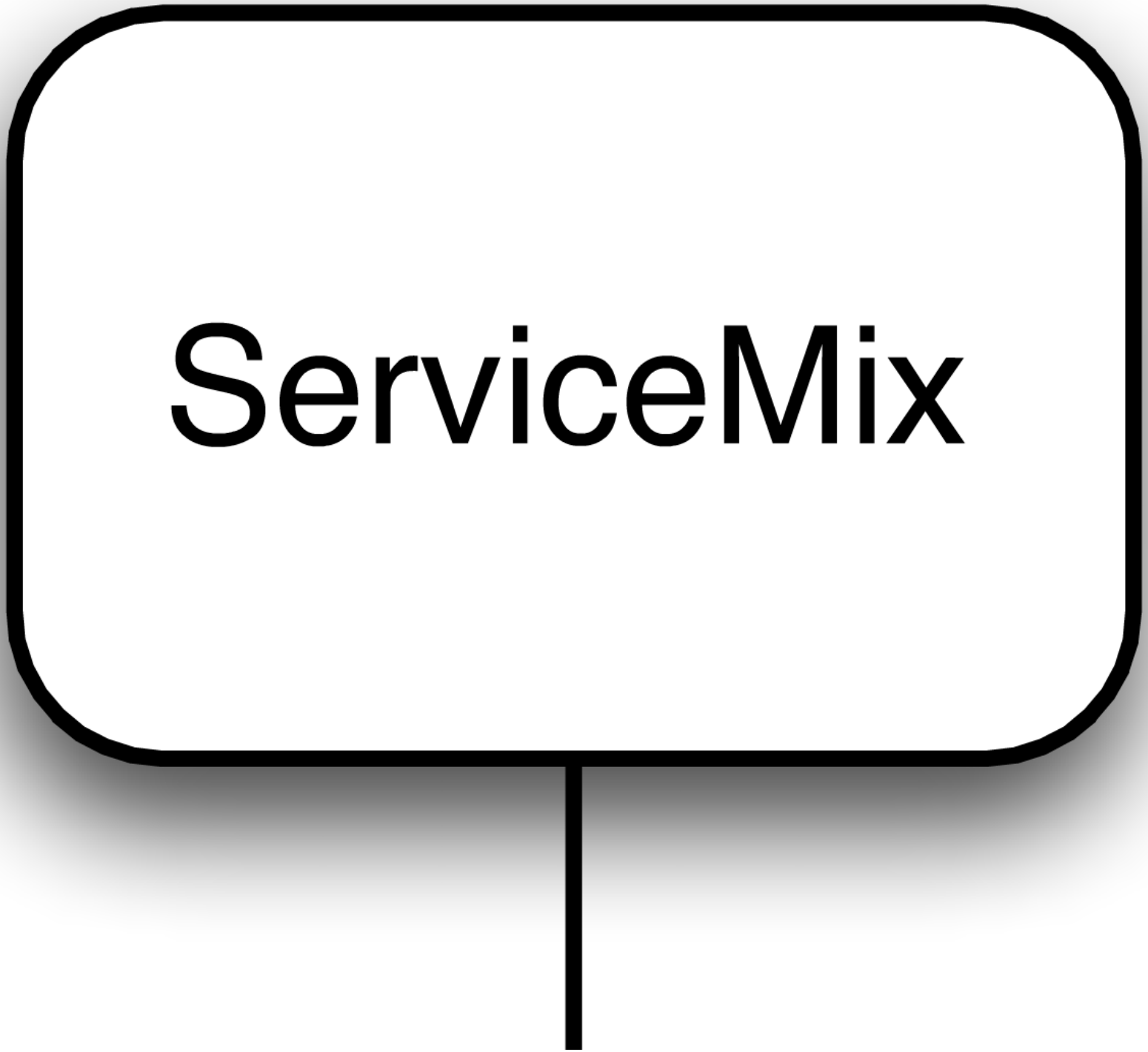

Web
JAXRS
ServiceMix
ServiceMix
ServiceMix
JAXWS
Adaptors
Cassandra 1
Cassandra 2
Cassandra 3
Cassandra 4
Cassandra 5
Cassandra 6
ActiveMQ Master
ActiveMQ Slave
Network of Brokers
ActiveMQ Master
ActiveMQ Slave
ServiceMix
ServiceMix
ServiceMix

ADAPTOR TIER

ESB ENDPOINTS

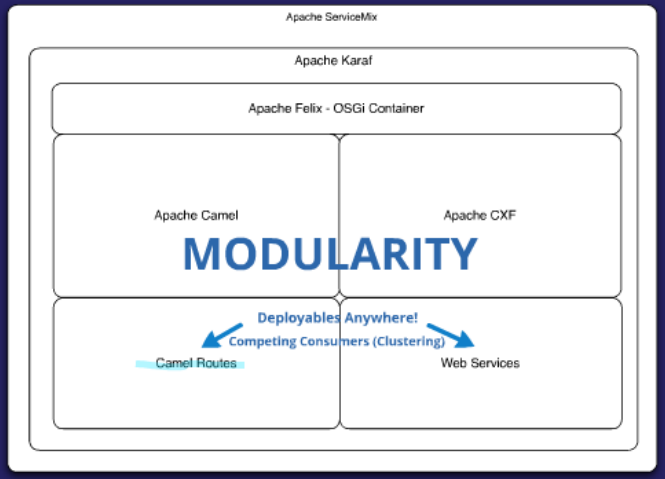
Thee are the adaptors that live in locations near the government services. These are endpoints that translate JAXRS requests to the government services; be it direct database access or WS Proxy

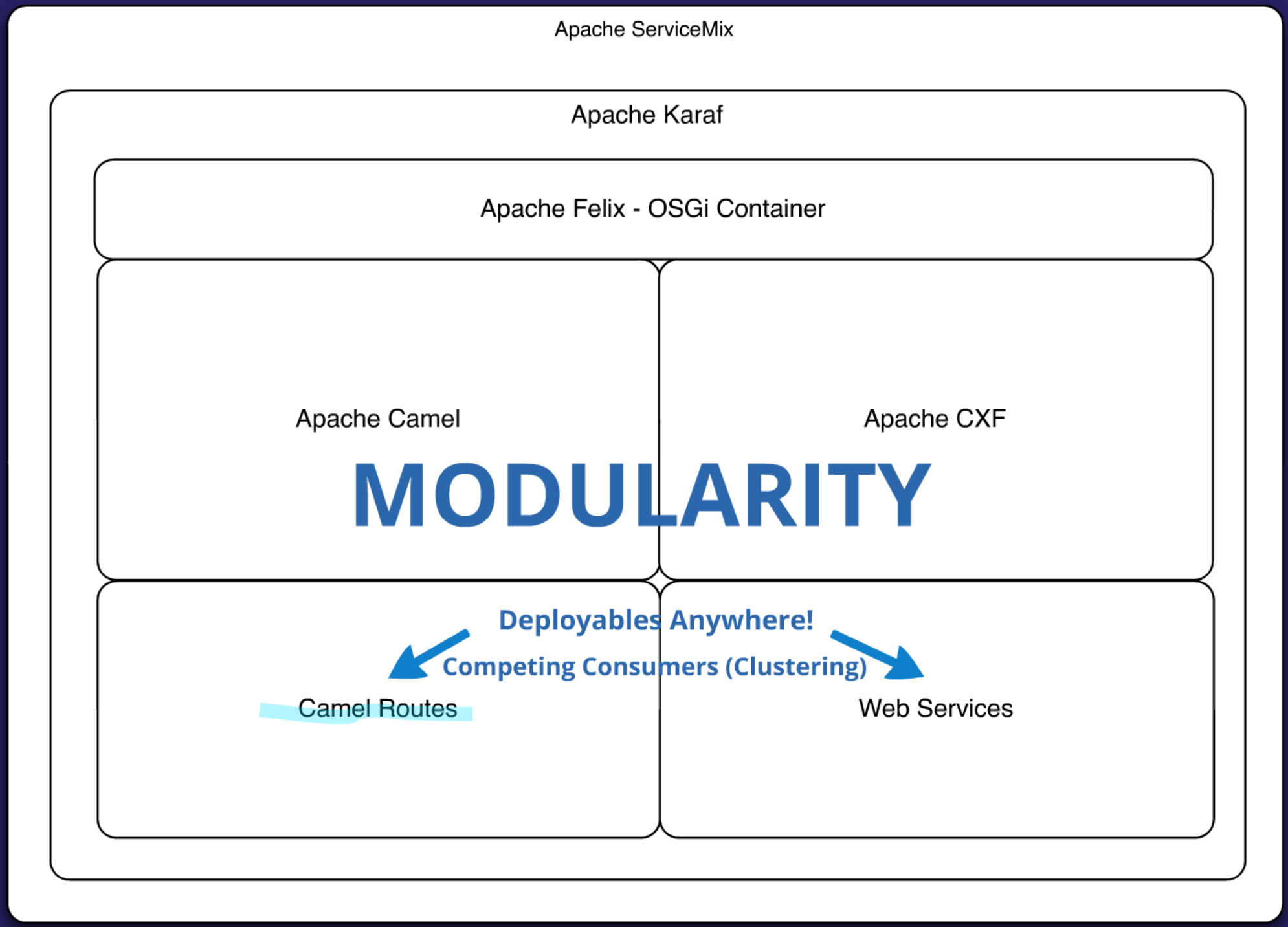

Routes
JAXWS
ServiceMix
JAXWS
Internal SOAP Service
Microsoft or PHP
JPA
Oracle
Postresql
ActiveMQ
Master
ActiveMQ
Slave


ServiceMix

ServiceMix

- Apache Karaf
- OSGi Container - Provides Modularity
- Apache Camel and CXF Bundles




Apache ServiceMix
Apache Karaf
Apache Felix - OSGi Container
Apache Camel
Apache CXF
MODULARITY
Deployables Anywhere!
Competing Consumers (Clustering)
Camel Routes
Web Services

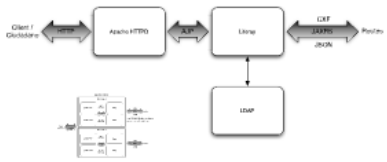
DINARDAP

SINARDAP - SISTEMA NACIONAL DE DATOS PÚBLICOS

WEB TIER

LIFERAY

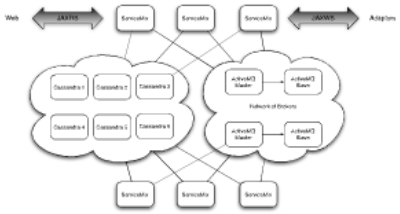
Front end - Client access for registration of cedulas and data access. This is the direct point for which the end user will access all of their information.



ROUTES TIER

ESB

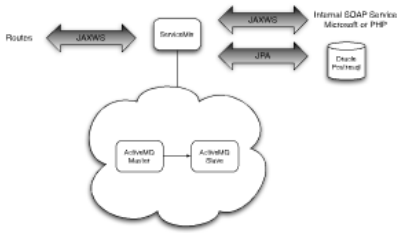
The main conduit for handling requests and mediation between the web tier and the external system adaptors. This tier handles guaranteed messaging and data caching.

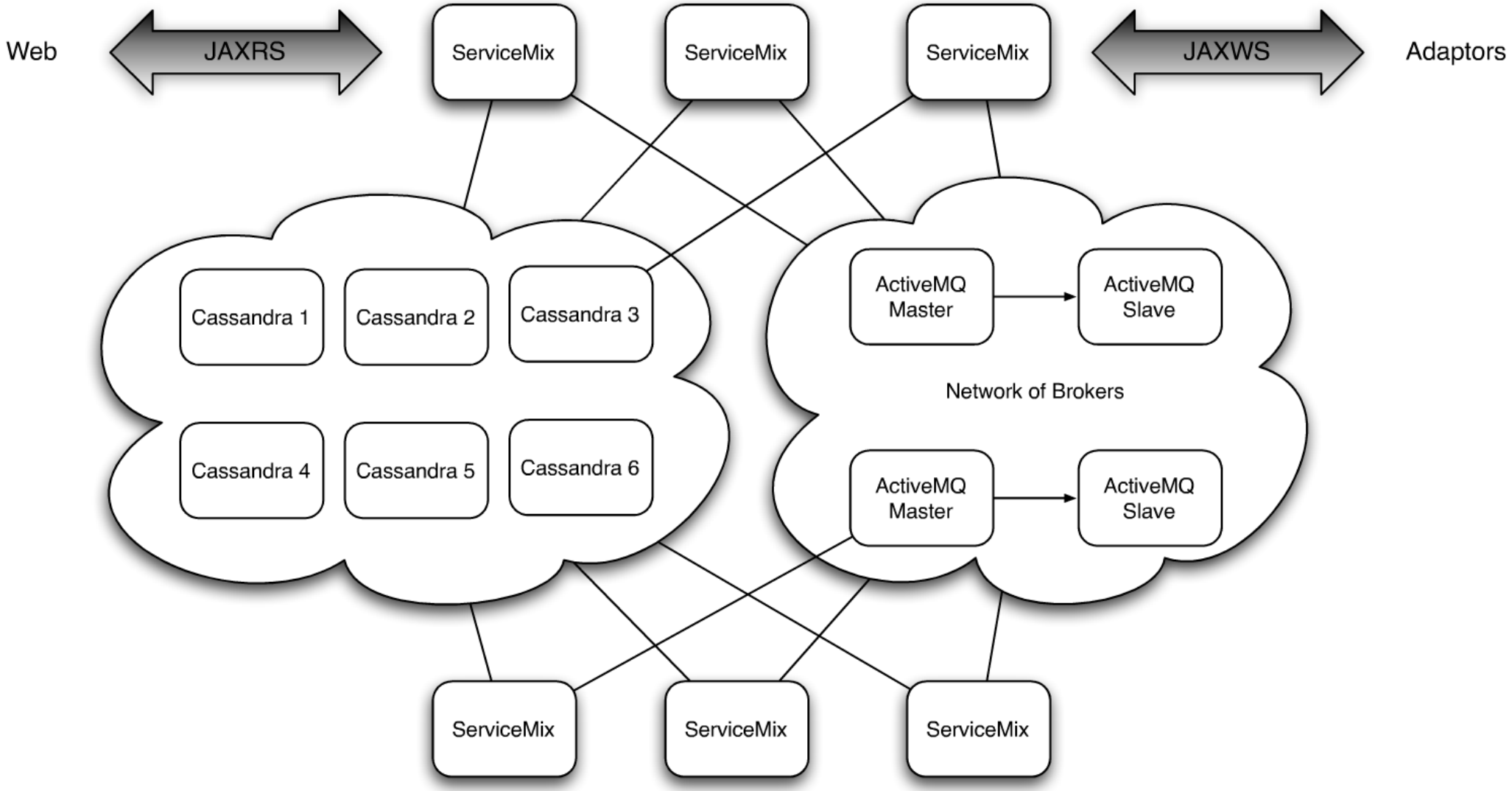


ADAPTOR TIER

ESB ENDPOINTS

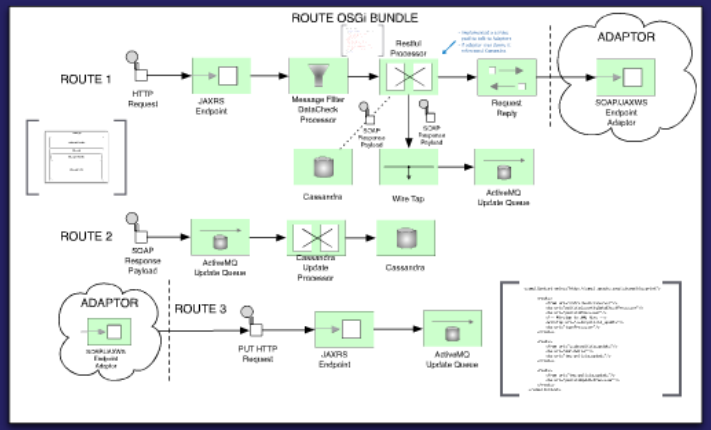
Thee are the adaptors that live in locations near the government services. These are endpoints that translate JAXRS requests to the government services; be it direct database access or WS Proxy

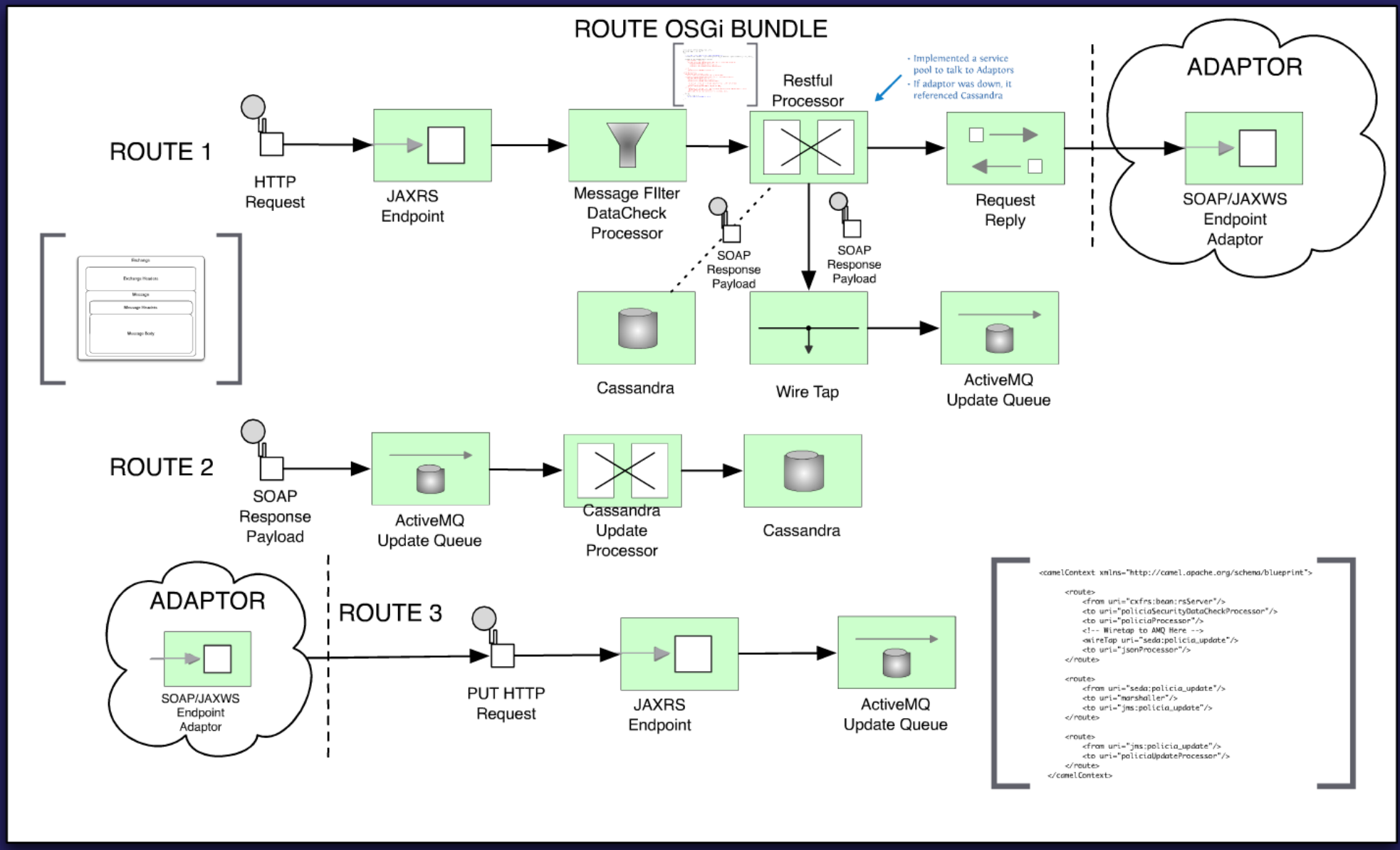


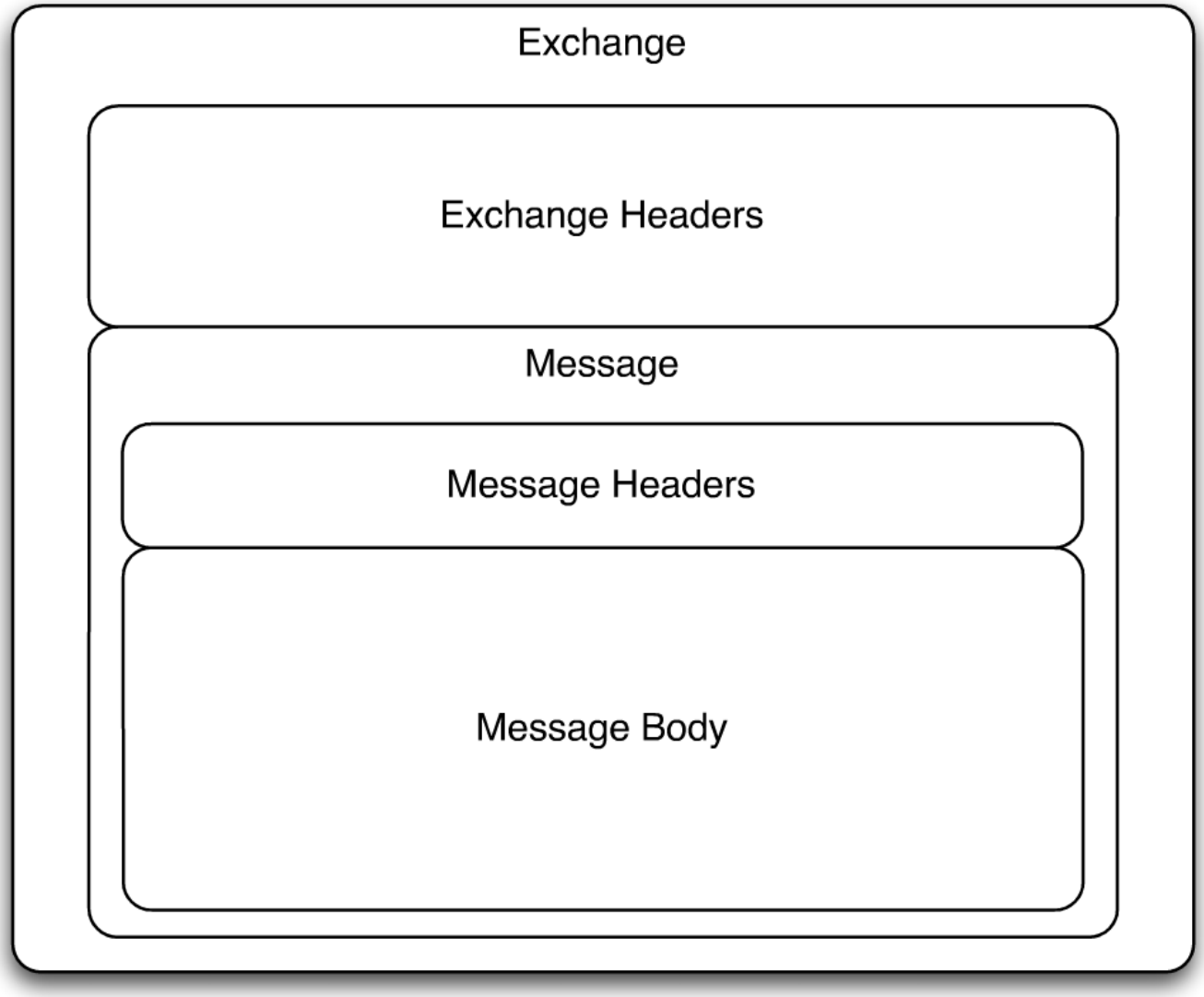

Web
JAXRS
ServiceMix
ServiceMix
ServiceMix
JAXWS
Adaptors
Cassandra 1
Cassandra 2
Cassandra 3
Cassandra 4
Cassandra 5
Cassandra 6
ActiveMQ Master
ActiveMQ Slave
Network of Brokers
ActiveMQ Master
ActiveMQ Slave
ServiceMix
ServiceMix
ServiceMix

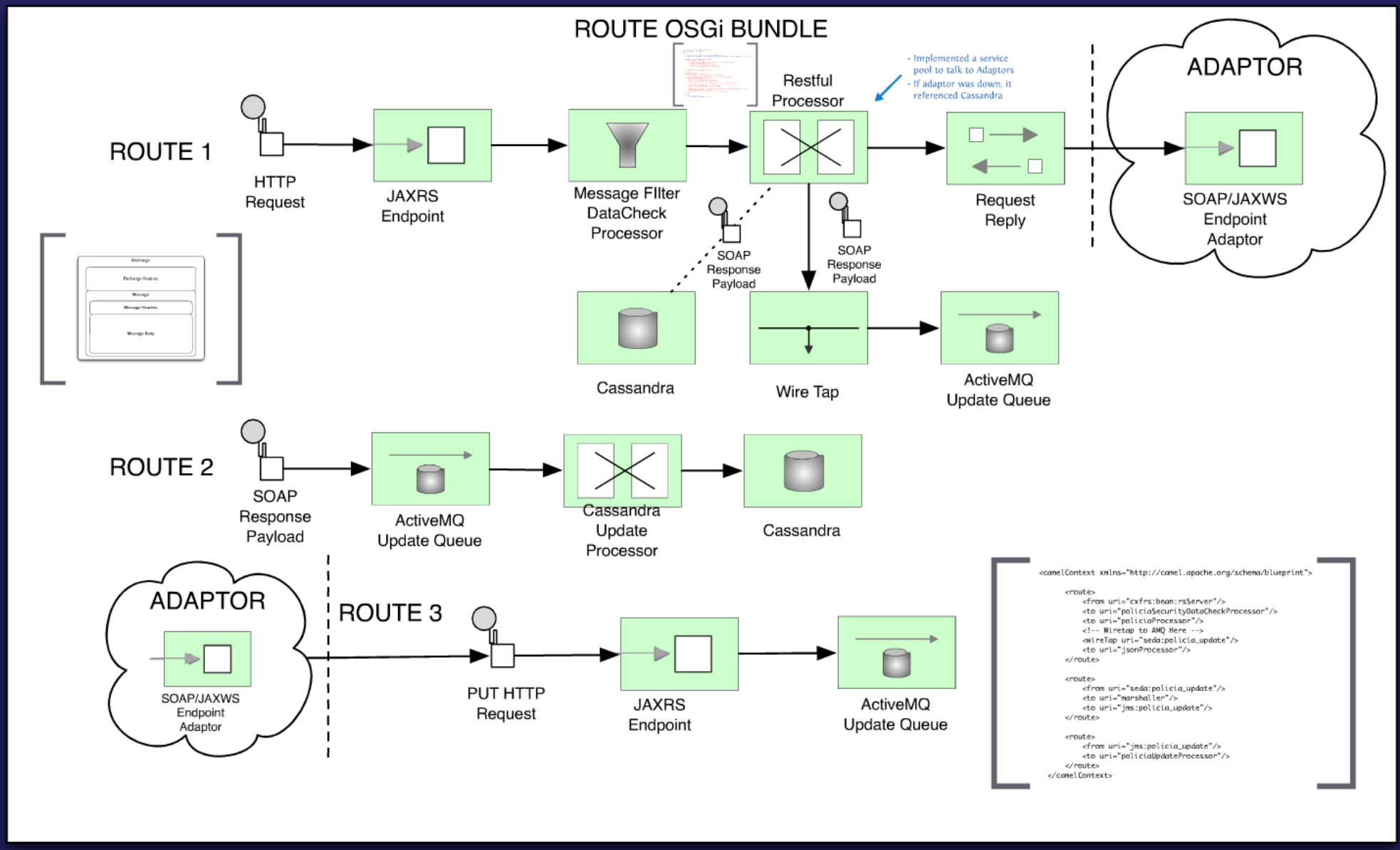
Apache Camel (Routes)

- Used to mediate the JAXRS calls
- Ensured the calls went through, even on adaptor service failure




ROUTE OSGi BUNDLE
ROUTE 1
HTTP Request
JAXRS Endpoint
Message FIlter DataCheck Processor
Restful Processor
- Implemented a service pool to talk to Adaptors
- If adaptor was down, it referenced Cassandra
Request Reply
ADAPTOR
SOAP/JAXWS Endpoint Adaptor
SOAP Response Payload
SOAP Response Payload
Cassandra
Wire Tap
ActiveMQ Update Queue
ROUTE 2
SOAP Response Payload
ActiveMQ Update Queue
Cassandra Update Processor
Cassandra
ADAPTOR
SOAP/JAXWS Endpoint Adaptor
ROUTE 3
PUT HTTP Request
JAXRS Endpoint
ActiveMQ Update Queue
<camelContext xmlns="http://camel.apache.org/schema/blueprint">
<route>
<from uri="cxfrs:bean:rsServer"/>
<to uri="policiaSecurityDataCheckProcessor"/>
<to uri="policiaProcessor"/>
<!-- Wiretap to AMQ Here -->
<wireTap uri="seda:policia_update"/>
<to uri="jsonProcessor"/>
</route>
<route>
<from uri="seda:policia_update"/>
<to uri="marshaller"/>
<to uri="jms:policia_update"/>
</route>
<route>
<from uri="jms:policia_update"/>
<to uri="policiaUpdateProcessor"/>
</route>
</camelContext>


Exchange
Exchange Headers
Message
Message Headers
Message Body


ROUTE OSGi BUNDLE
ROUTE 1
HTTP Request
JAXRS Endpoint
Message FIlter DataCheck Processor
Restful Processor
- Implemented a service pool to talk to Adaptors
- If adaptor was down, it referenced Cassandra
Request Reply
ADAPTOR
SOAP/JAXWS Endpoint Adaptor
SOAP Response Payload
SOAP Response Payload
Cassandra
Wire Tap
ActiveMQ Update Queue
ROUTE 2
SOAP Response Payload
ActiveMQ Update Queue
Cassandra Update Processor
Cassandra
ADAPTOR
SOAP/JAXWS Endpoint Adaptor
ROUTE 3
PUT HTTP Request
JAXRS Endpoint
ActiveMQ Update Queue
<camelContext xmlns="http://camel.apache.org/schema/blueprint">
<route>
<from uri="cxfrs:bean:rsServer"/>
<to uri="policiaSecurityDataCheckProcessor"/>
<to uri="policiaProcessor"/>
<!-- Wiretap to AMQ Here -->
<wireTap uri="seda:policia_update"/>
<to uri="jsonProcessor"/>
</route>
<route>
<from uri="seda:policia_update"/>
<to uri="marshaller"/>
<to uri="jms:policia_update"/>
</route>
<route>
<from uri="jms:policia_update"/>
<to uri="policiaUpdateProcessor"/>
</route>
</camelContext>

```
<camelContext xmlns="http://camel.apache.org/schema/blueprint">

      <route>
          <from uri="cxfrs:bean:rsServer"/>
          <to uri="policiaSecurityDataCheckProcessor"/>
          <to uri="policiaProcessor"/>
          <!-- Wiretap to AMQ Here -->
          <wireTap uri="seda:policia_update"/>
          <to uri="jsonProcessor"/>
      </route>

      <route>
          <from uri="seda:policia_update"/>
          <to uri="marshaller"/>
          <to uri="jms:policia_update"/>
      </route>

      <route>
          <from uri="jms:policia_update"/>
          <to uri="policiaUpdateProcessor"/>
      </route>
  </camelContext>
```

OSGi BUNDLE

- Implemented a service pool to talk to Adaptors
- If adaptor was down, it referenced Cassandra

Restful Processor

ter

k

SOAP Response Payload

SOAP Response Payload

Request

Reply

```
String cedula = inMessage.getBody(String.class);
PoliciaServicePort client = null;
try {

    client = (PoliciaServicePort) getServicePool().borrowObject();
    Antecedentes registroMercantil = (Antecedentes) getCaller().callRemoteService("getAntecedentes", client, cedula);

    exchange.getIn().setHeader("cedula", cedula);
    if (registroMercantil == null) {
        Antecedentes antecedentesFromCassandra = persistentSvc.findAntecedentes(cedula);
        if (antecedentesFromCassandra != null) {
            exchange.getIn().setHeader("cedula", cedula);
            exchange.getIn().setBody(antecedentesFromCassandra);
        }
    } else {
        exchange.getIn().setBody(registroMercantil);
    }
} catch (Exception e) {
    LOG.warn("Web exception caught " + e);
    //Service is down so get the DTO record directly from Cassandra
    Antecedentes antecedentesFromCassandra = persistentSvc.findAntecedentes(cedula);
    if (antecedentesFromCassandra != null) {
        exchange.getIn().setHeader("cedula", cedula);
        exchange.getIn().setBody(antecedentesFromCassandra);

        //We don't want to send this to the update processor since it comes from cassandra...
        // so stop it in its tracks
        exchange.setProperty(Exchange.ROUTE_STOP, Boolean.TRUE);
    } else {
        //Error handling to go here if not found in Cassandra because antecedentesFromCassandra will be null
        //and hence 404 error should be returned
        webFault(404, "La cedula que ingreso no existe.", exchange, true);
    }
} finally {
    if (client != null) {
        getServicePool().returnObject(client);
    }
}
```

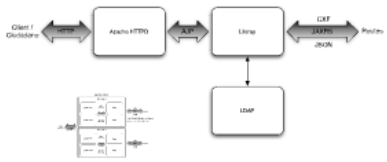
DINARDAP

SINARDAP - SISTEMA NACIONAL DE DATOS PÚBLICOS

WEB TIER

LIFERAY

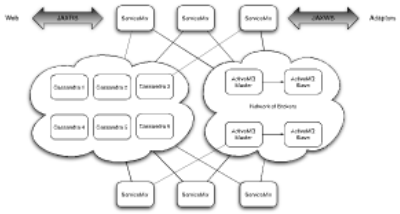
Front end - Client access for registration of cedulas and data access. This is the direct point for which the end user will access all of their information.



ROUTES TIER

ESB

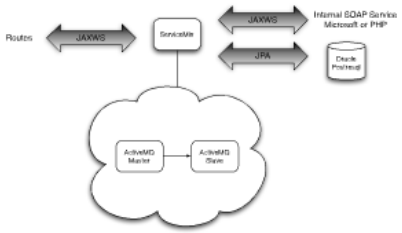
The main conduit for handling requests and mediation between the web tier and the external system adaptors. This tier handles guaranteed messaging and data caching.

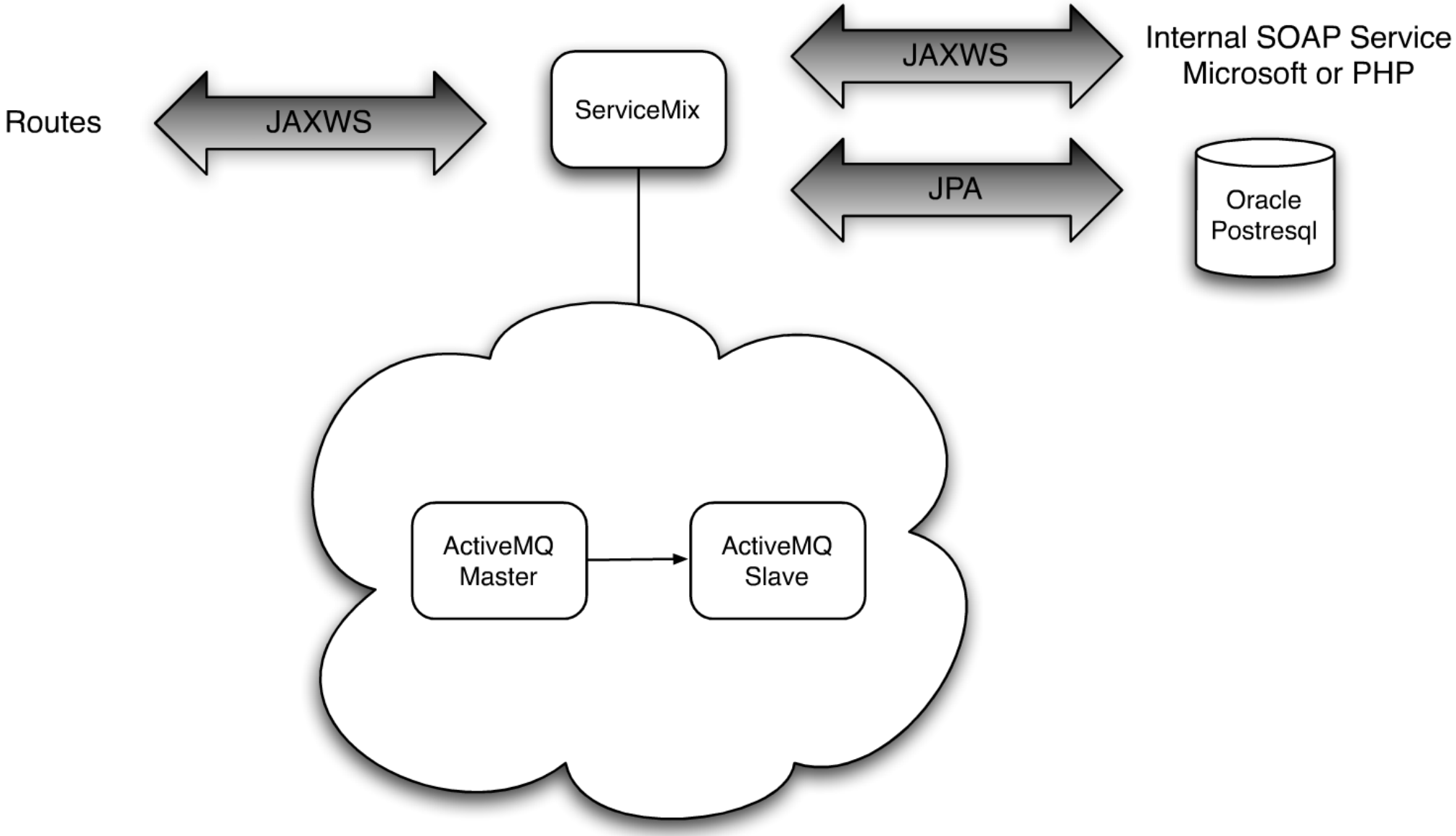


ADAPTOR TIER

ESB ENDPOINTS

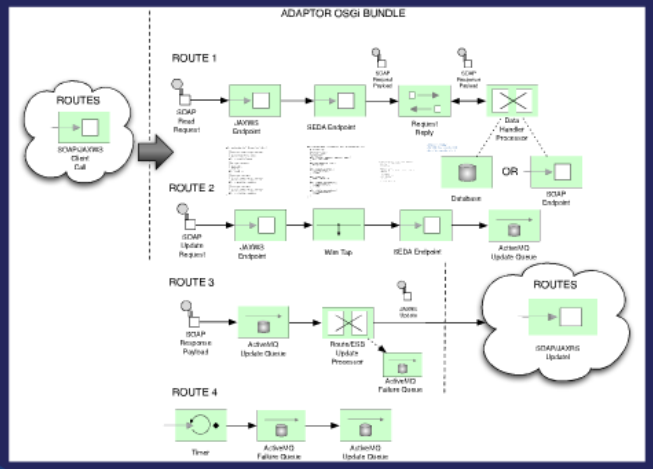
Thee are the adaptors that live in locations near the government services. These are endpoints that translate JAXRS requests to the government services; be it direct database access or WS Proxy

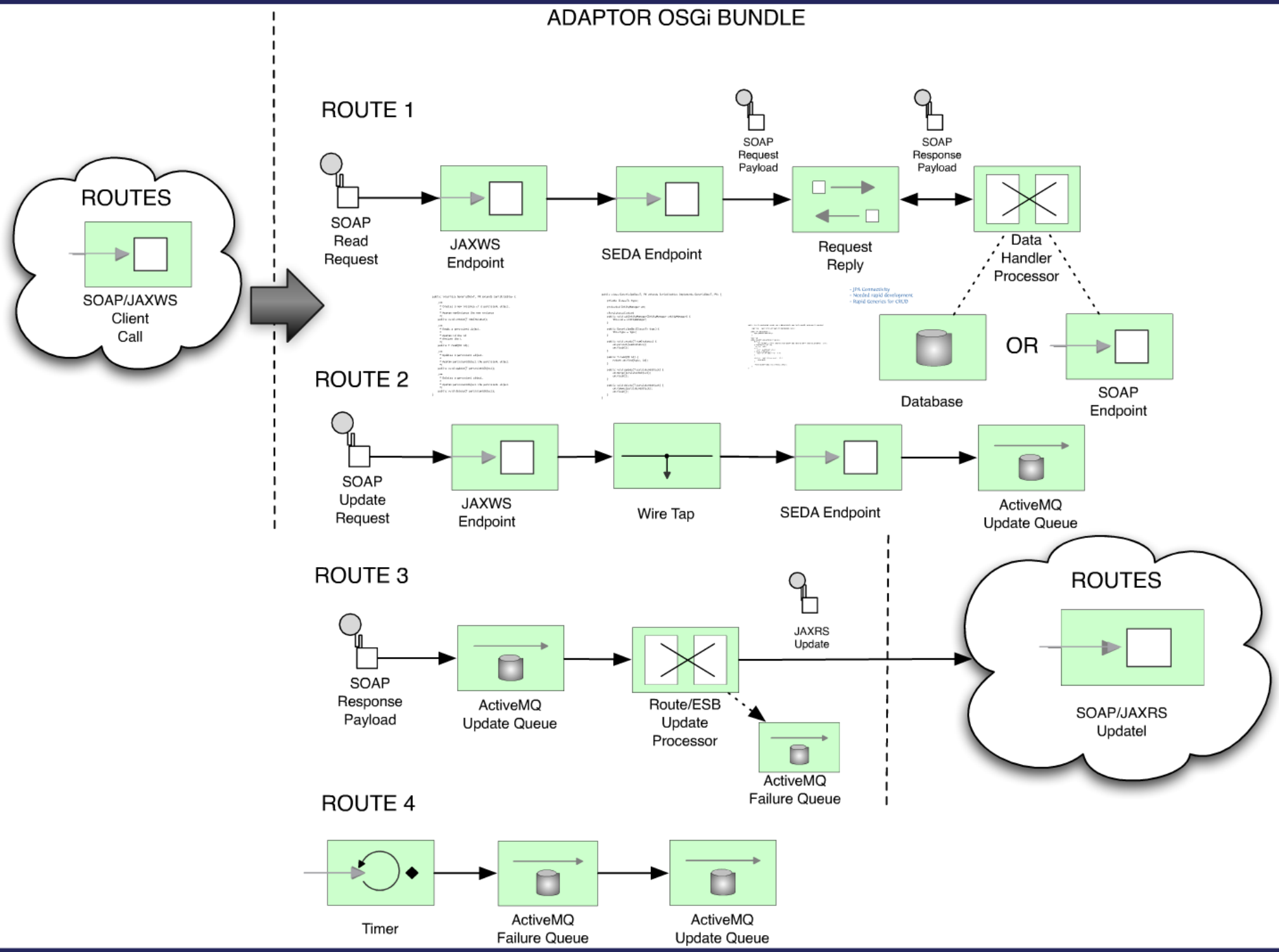



Routes
JAXWS
ServiceMix
JAXWS
Internal SOAP Service
Microsoft or PHP
JPA
Oracle
Postresql
ActiveMQ
Master
ActiveMQ
Slave

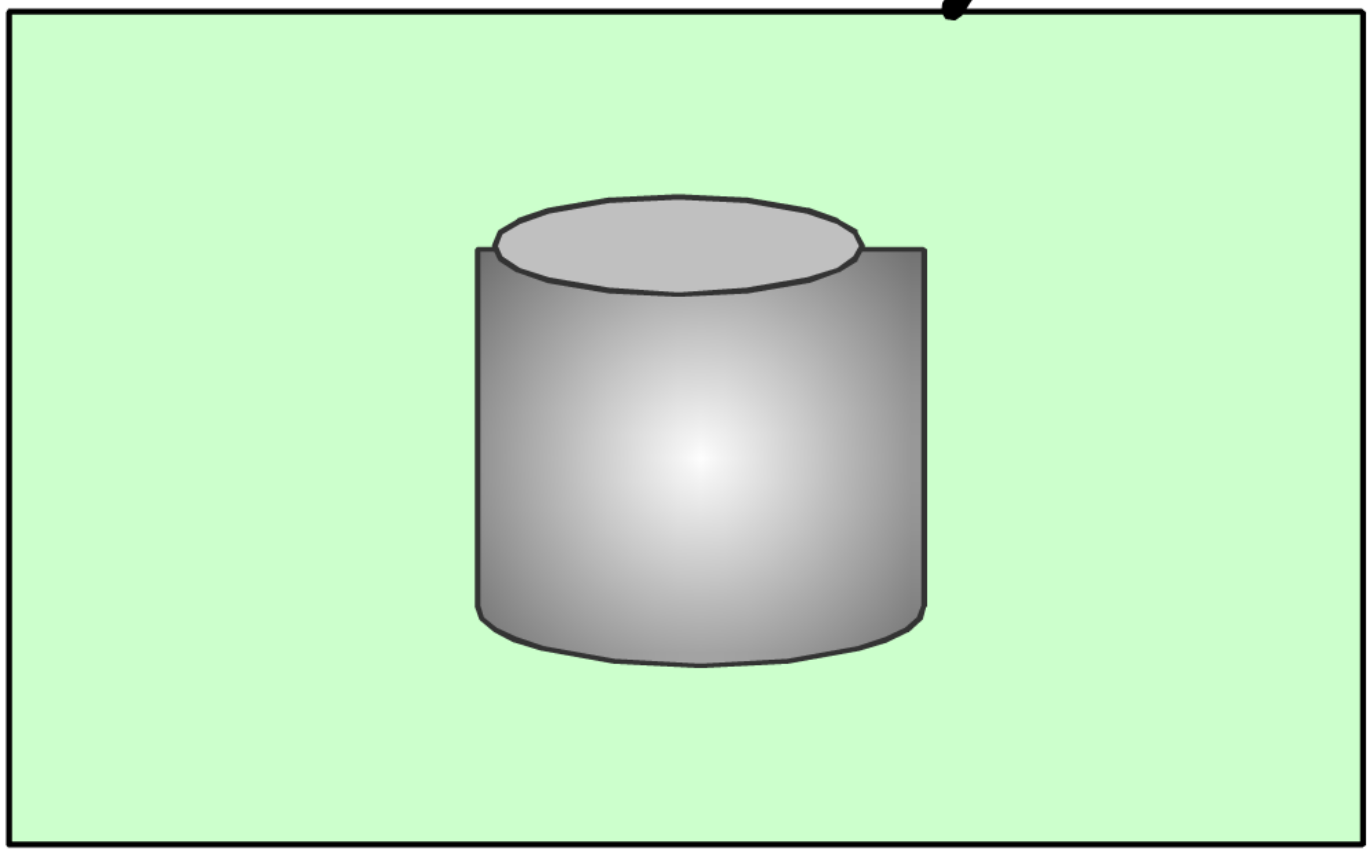
Apache Camel (Adaptors)

- Each "company" has its own service
- Lives at its respective data center close to the "company"
- Mediates between SOAP and the back-end company datasink

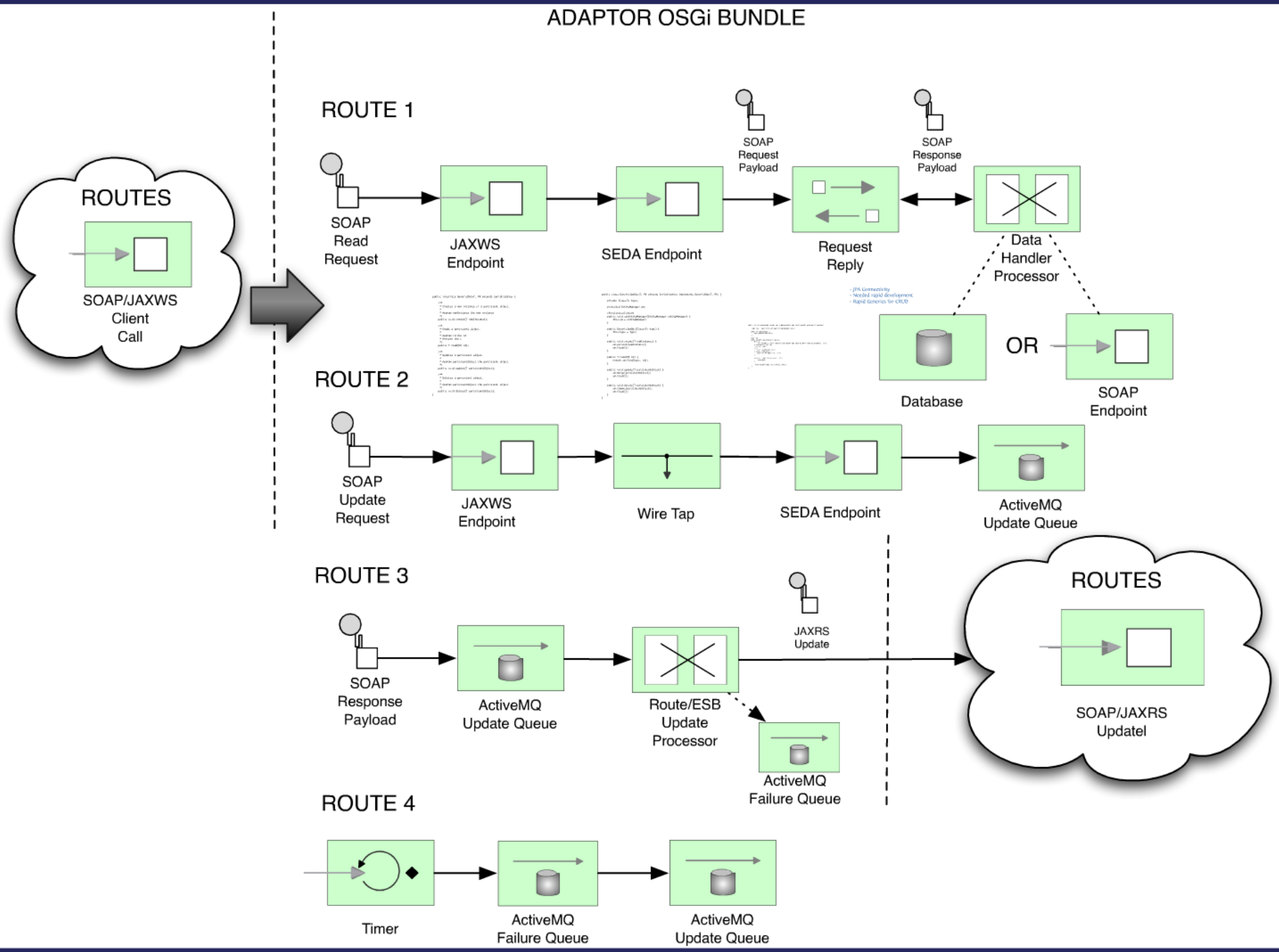



ADAPTOR OSGi BUNDLE
ROUTES
SOAP/JAXWS
Client
Call
ROUTE 1
SOAP
Read
Request
JAXWS
Endpoint
SEDA Endpoint
SOAP
Request
Payload
Request
Reply
SOAP
Response
Payload
Data
Handler
Processor
Database
OR
SOAP
Endpoint
ROUTE 2
SOAP
Update
Request
JAXWS
Endpoint
Wire Tap
SEDA Endpoint
ActiveMQ
Update Queue
ROUTE 3
SOAP
Response
Payload
ActiveMQ
Update Queue
Route/ESB
Update
Processor
JAXRS
Update
ActiveMQ
Failure Queue
ROUTES
SOAP/JAXRS
Updatel
ROUTE 4
Timer
ActiveMQ
Failure Queue
ActiveMQ
Update Queue

- JPA Connectivity
- Needed rapid development
- Rapid Generics for CRUD



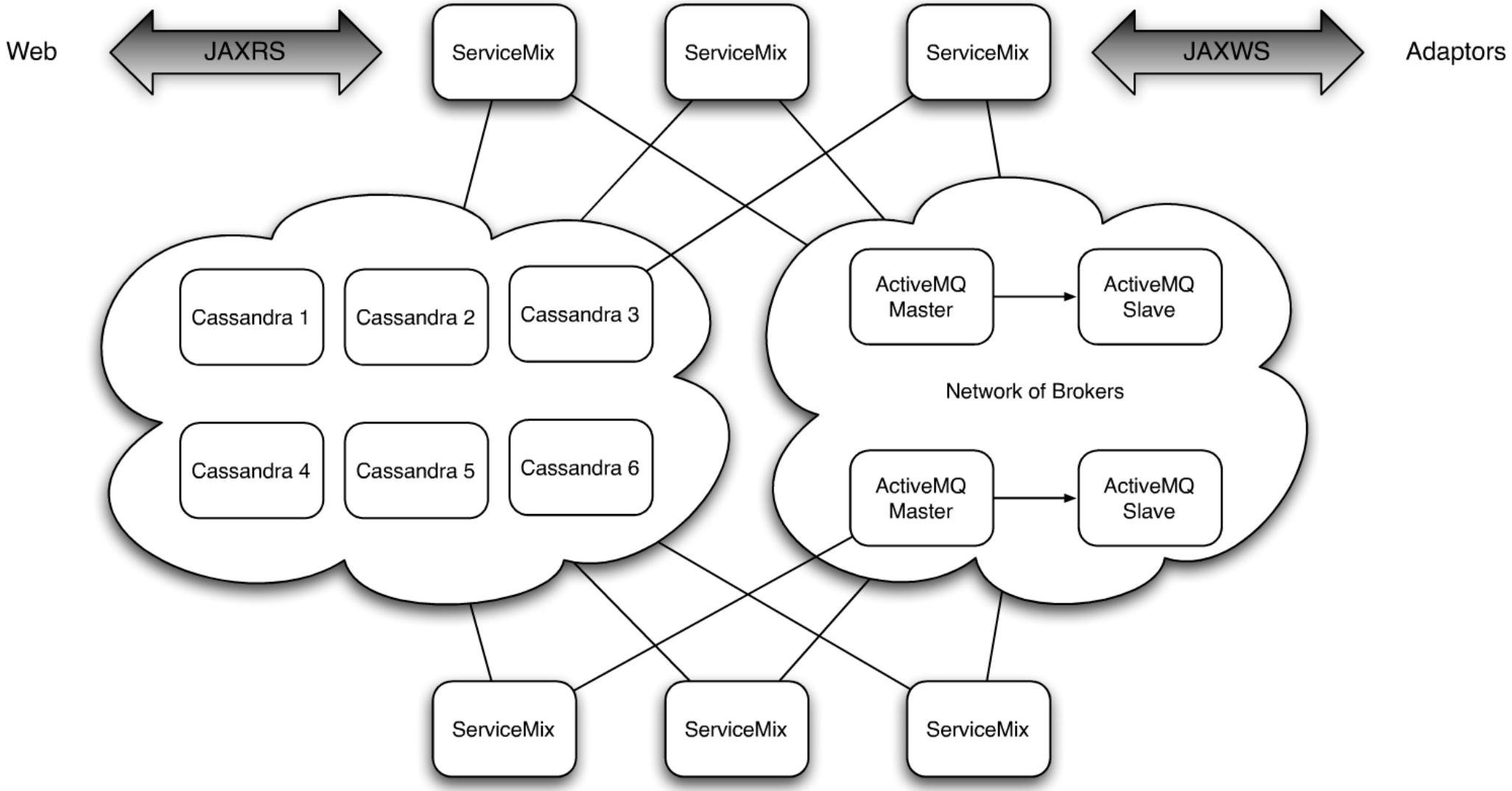
Database


ADAPTOR OSGi BUNDLE
ROUTES
SOAP/JAXWS
Client
Call
ROUTE 1
SOAP
Read
Request
JAXWS
Endpoint
SEDA Endpoint
SOAP
Request
Payload
Request
Reply
SOAP
Response
Payload
Data
Handler
Processor
Database
OR
SOAP
Endpoint
ROUTE 2
SOAP
Update
Request
JAXWS
Endpoint
Wire Tap
SEDA Endpoint
ActiveMQ
Update Queue
ROUTE 3
SOAP
Response
Payload
ActiveMQ
Update Queue
Route/ESB
Update
Processor
JAXRS
Update
ActiveMQ
Failure Queue
ROUTES
SOAP/JAXRS
Updatel
ROUTE 4
Timer
ActiveMQ
Failure Queue
ActiveMQ
Update Queue

```
public interface GenericDAO<T, PK extends Serializable> {

    /**
     * Creates a new instance of a persistent object.
     *
     * @param newInstance the new instance
     */
    public void create(T newInstance);

    /**
     * Reads a persistent object.
     *
     * @param id the id
     * @return the t
     */
    public T read(PK id);

    /**
     * Updates a persistent object.
     *
     * @param persistentObject the persistent object
     */
    public void update(T persistentObject);

    /**
     * Deletes a persistent object.
     *
     * @param persistentObject the persistent object
     */
    public void delete(T persistentObject);
}
```

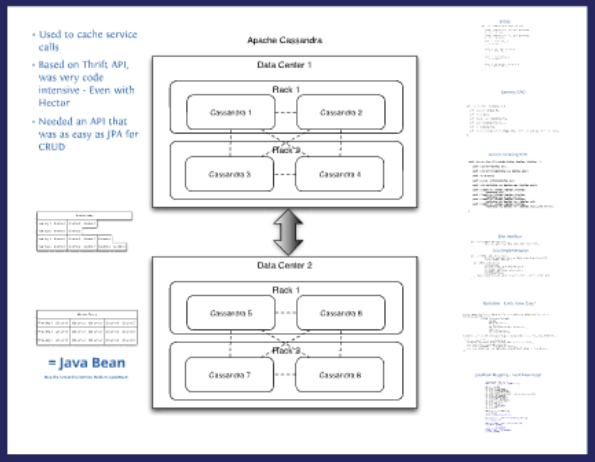
```
public class GenericJpaDao<T, PK extends Serializable> implements GenericDAO<T, PK> {

    private Class<T> type;

    protected EntityManager em;

    @PersistenceContext
    public void setEntityManager(EntityManager entityManager) {
        this.em = entityManager;
    }

    public GenericJpaDao(Class<T> type) {
        this.type = type;
    }

    public void create(T newInstance) {
        em.persist(newInstance);
        em.flush();
    }

    public T read(PK id) {
        return em.find(type, id);
    }

    public void update(T persistentObject) {
        em.merge(persistentObject);
        em.flush();
    }

    public void delete(T persistentObject) {
        em.remove(persistentObject);
        em.flush();
    }
}
```

```
public class PersonasDaoImpl extends GenericJpaDao<VmAntPersona, AntPersonasPK> implements PersonasDao {

    Logger log = LoggerFactory.getLogger(PersonasDaoImpl.class);

    public PersonasDaoImpl() {
        super(VmAntPersona.class);
    }

    @Override
    public VmAntPersona getById(String id) {
        Query q =
             em.createQuery("select contactos from VmAntPersona contactos where contactos.documento = :id");
        q.setParameter("id", id);
        List list = null;
        try {
            list = q.getResultList();
        } catch (Exception e) {
            log.error("Personas error " + e);
        }

        if (list == null || list.size() == 0) {
            return null;
        }

        return (VmAntPersona) list.iterator().next();
    }
}
```

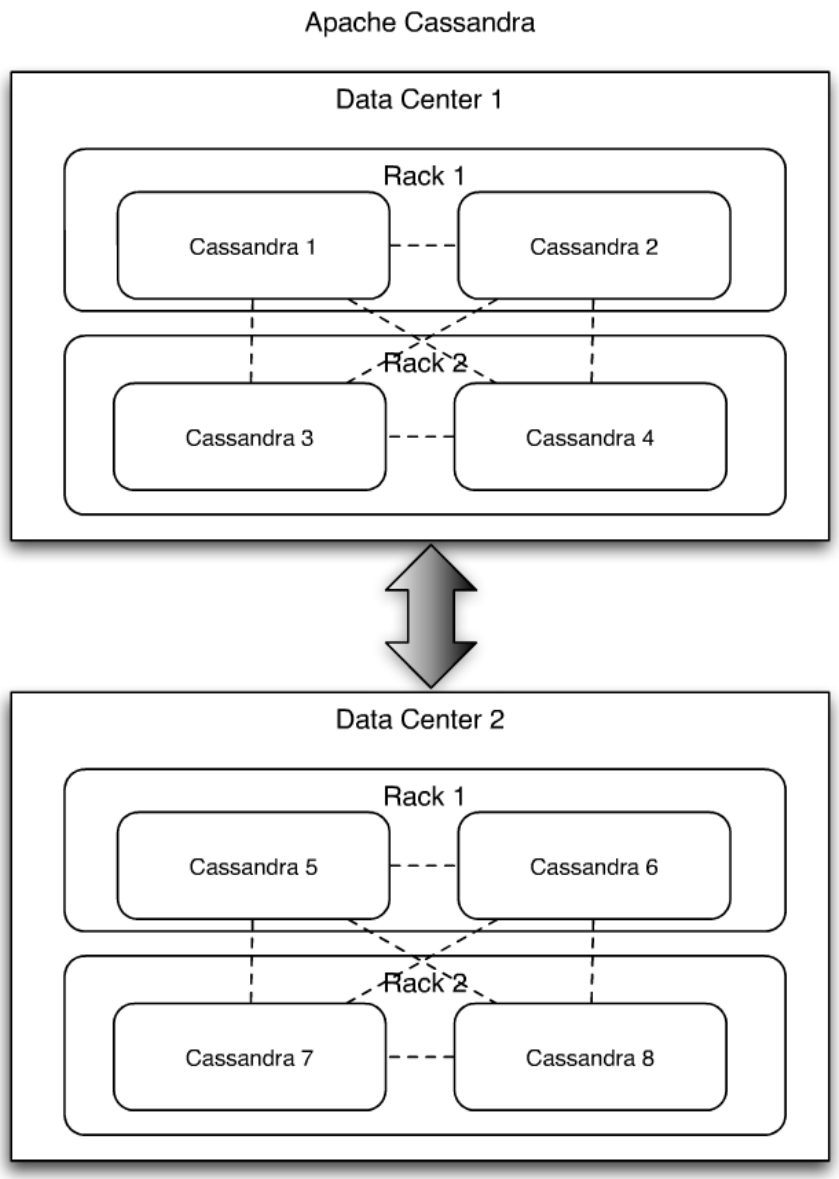

Web
JAXRS
ServiceMix
ServiceMix
ServiceMix
JAXWS
Adaptors
Cassandra 1
Cassandra 2
Cassandra 3
Cassandra 4
Cassandra 5
Cassandra 6
ActiveMQ Master
ActiveMQ Slave
Network of Brokers
ActiveMQ Master
ActiveMQ Slave
ServiceMix
ServiceMix
ServiceMix

Apache Cassandra

- NoSQL Database that is Java-based
- Allowed system to have fault tolerant data access
- Extremely high performant and has cross-datacenter capabilitites and its decentralized



- Used to cache service calls
- Based on Thrift API, was very code intensive - Even with Hector
- Needed an API that was as easy as JPA for CRUD



Column Family					
Row Key 1	Column 1	Column 2	Column 3		
Row Key 2	Column 1	Column 2			
Row Key 1	Column 1	Column 2	Column 3	Column 4	
Row Key n	Column 1	Column 2	Column 3	Column 4	Column n

Column Family					
Row Key 1	Column 1	Column 2	Column 3	Column 4	Column 5
Row Key 2	Column 1	Column 2	Column 3	Column 4	Column 5
Row Key n	Column 1	Column 2	Column 3	Column 4	Column 5

= Java Bean

Map the row and column to fields in a JavaBean

Entity

Generic DAO

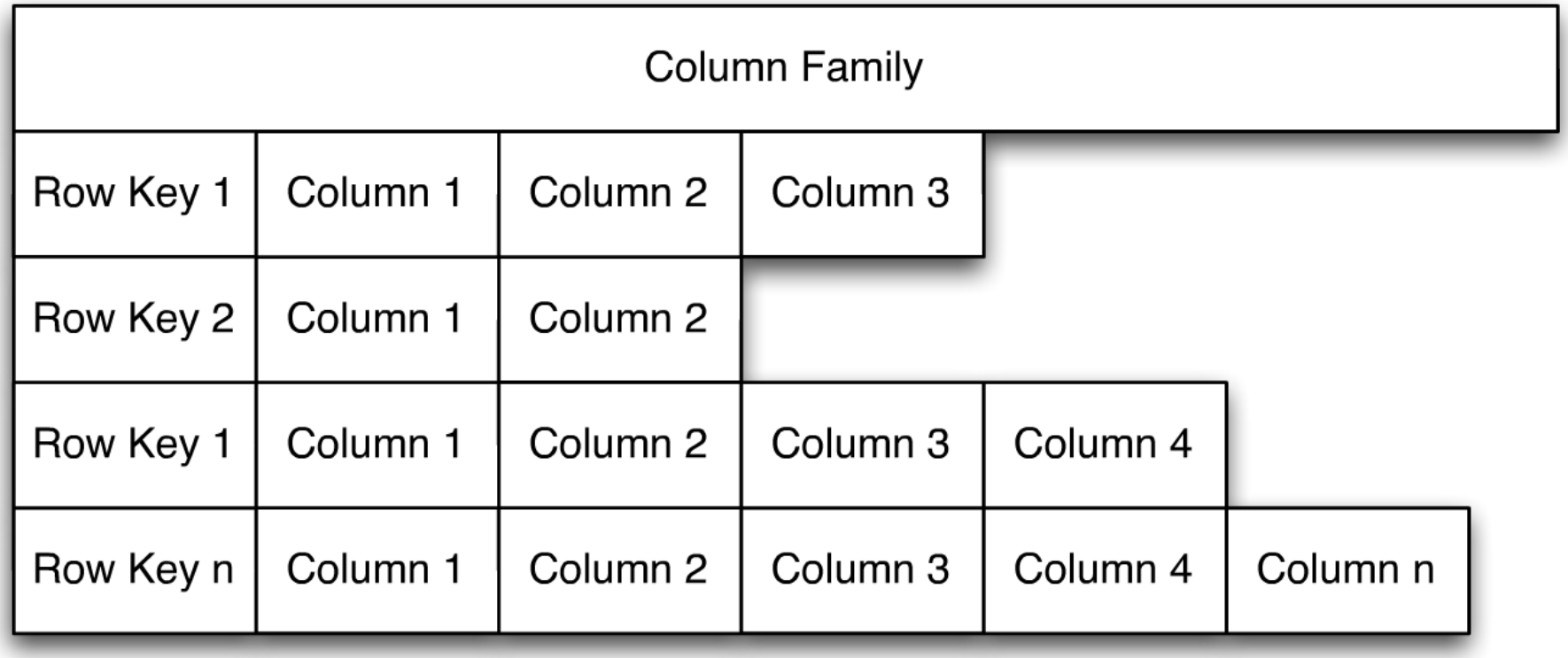
Generic Iterating DAO

DAO Interface

DAO Implementation

Iteration - Look How Easy!

JavaBean Mapping - Look How Easy!


Column Family
Row Key 1
Column 1
Column 2
Column 3
Row Key 2
Column 1
Column 2
Row Key 1
Column 1
Column 2
Column 3
Column 4
Row Key n
Column 1
Column 2
Column 3
Column 4
Column n

Column Family					
Row Key 1	Column 1	Column 2	Column 3	Column 4	Column 5
Row Key 2	Column 1	Column 2	Column 3	Column 4	Column 5
Row Key n	Column 1	Column 2	Column 3	Column 4	Column 5

= Java Bean

Map the row and column to fields in a JavaBean

Entity

```
public class CTEFechaTipoCedulaCompositeColumn {

    @CompositeComponent(order = 1, type = Type.UTF8Type)
    private String tipo;

    @CompositeComponent(order = 2, type = Type.UTF8Type)
    private String cedula;

    public String getTipo() {
        return tipo;
    }

    public void setTipo(String tipo) {
        this.tipo = tipo;
    }

    public String getCedula() {
        return cedula;
    }

    public void setCedula(String cedula) {
        this.cedula = cedula;
    }
}
```

Generic DAO

```
public interface GenericDao<KeyType, T> {

    public void delete(KeyType key);

    public Set getKeys();

    public boolean containsKey(KeyType key);

    public void save(KeyType key, T pojo);

    public T find(KeyType key);

    public Set<T> findItems(final List<KeyType> keys, final String rangeFrom, final String rangeTo);

}
```

Generic Iterating DAO

Generic Iterating DAO

```
public interface GenericIteratingDao<KeyType, NameType, ValueType>  {

    public void delete(KeyType key);

    public void deleteColumn(KeyType key, NameType name);

    public Set getKeys();

    public boolean containsKey(KeyType key);

    public void save(KeyType key, NameType name, ValueType value);

    public ColumnIterator<KeyType, NameType, ValueType>
                find(KeyType key);
    public ColumnIterator<KeyType, NameType, ValueType>
                find(KeyType key, boolean reverse);
    public ColumnIterator<KeyType, NameType, ValueType>
                find(KeyType key, NameType start, NameType end);
    public ColumnIterator<KeyType, NameType, ValueType>
                find(KeyType key, NameType start, NameType end, boolean reverse);

}
```

DAO Interface

```
public interface CTEFechaTipoCedulaDao extends
                  GenericIteratingDao<String, CTEFechaTipoCedulaCompositeColumn, UUID> {
}
```

DAO Implementation

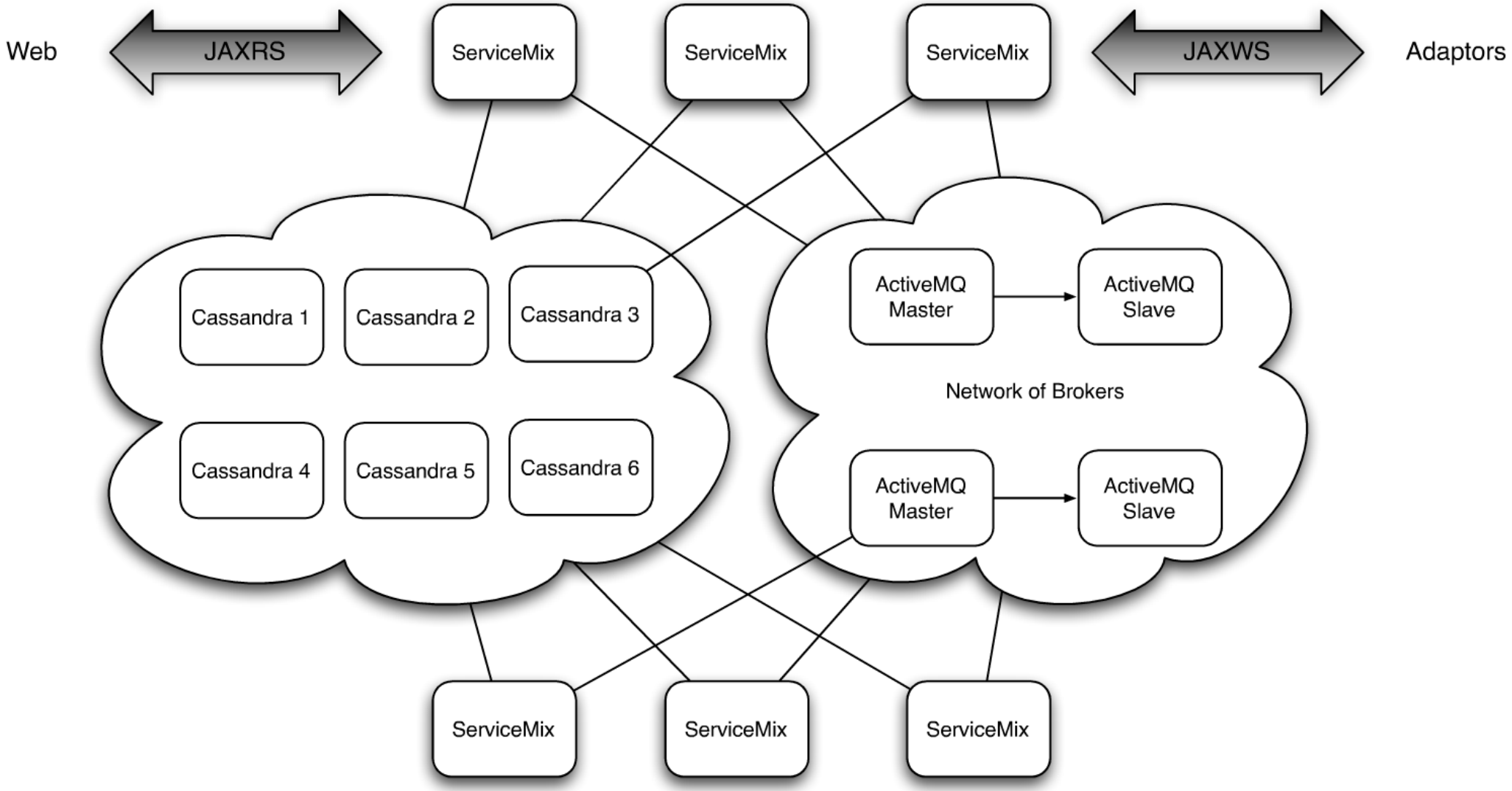
```
public class CTEFechaTipoCedulaDaoImpl extends
                  AbstractIteratingColumnFamilyDao<String, CTEFechaTipoCedulaCompositeColumn, UUID>
                  implements CTEFechaTipoCedulaDao {

    public CTEFechaTipoCedulaDaoImpl(
                  String clusterName,
                  CassandraKeyspaceConfigurator keyspaceConfigurator,
                  Class<String> stringClass,
                  Class<CTEFechaTipoCedulaCompositeColumn> nameClass,
                  Class<UUID> valueClass,
                  String columnFamilyName) {
        super(clusterName, keyspaceConfigurator, stringClass, nameClass, valueClass, columnFamilyName);
    }
}
```

Iteration - Look How Easy!

```
CassandraKeyspaceConfigurator keyspaceConfigurator = new CassandraKeyspaceConfigurator(….);
CTEFechaTipoCedulaDao dao =
                    new CTEFechaTipoCedulaDaoImpl(
                            CLUSTER,
                            keyspaceConfigurator,
                            String.class,
                            CTEFechaTipoCedulaCompositeColumn.class,
                            UUID.class,
                            CTE_FECHA_TIPO_CEDULA);

ColumnIterator<String, CTEFechaTipoCedulaCompositeColumn, UUID> iterator = dao.find("2012-07-20");

if (iterator != null && iterator.hasNext()){
    Column<CTEFechaTipoCedulaCompositeColumn, UUID> col = iterator.next();
    assertEquals("B", col.getName().getTipo());
    assertEquals("130893080", col.getName().getCedula());
    assertEquals(UUID.fromString("810f8d30-d832-11e1-9af9-e0f84722b90c"), col.getValue());
}
```

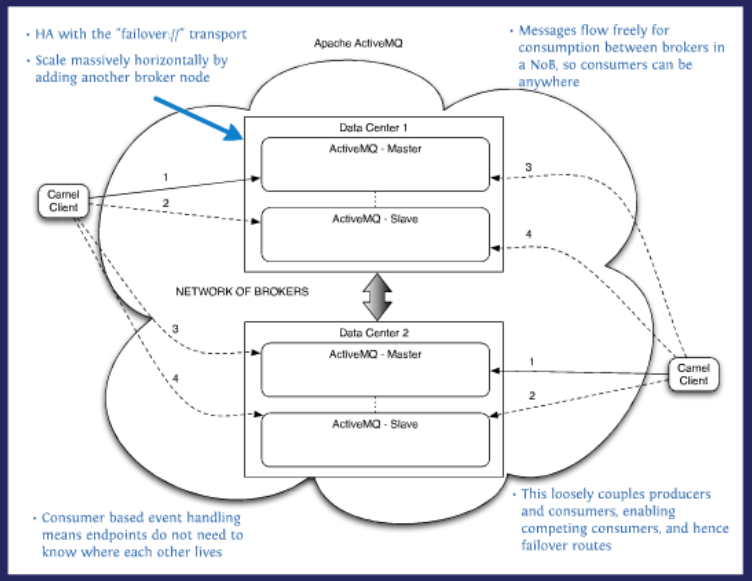
JavaBean Mapping - Look How Easy!

```
...
        //From Blueprint OSGi Service
        CNEDao cneDao = getOsgiService(CNEDao.class);

        CNE cne = new CNE();
        cne.setProvincia("MANABI");
        cne.setCanton("PUERTO LOPEZ");
        cne.setParroquia("PUERTO LOPEZ");
        cne.setRecinto("Este campo se habilitará antes de una elección");
        cne.setCedula("123456789");
        cne.setNombre("VALDEZ JUAN");
        cne.setFecha("2011-07-24");
        cne.setCertificado("123-4567");
        cne.setSufrago(false);
        cne.setMulta(21.4);
        cne.setPartido("NINGUNO");
        cne.setTipo("");

        CNEColumn col = new CNEColumn();
        col.setCne(cne);
        cneDao.save(cne.getCedula(), col);

        boolean contains = cneDao.containsKey(cne.getCedula());
        assertTrue(contains);

        CNEColumn newCNE = cneDao.find(cne.getCedula());
        assertNotNull(newCNE);

        cneDao.delete("130893080");
        newCNE = cneDao.find("130893080");
        assertNull(newCNE);
...
```


Web
JAXRS
ServiceMix
ServiceMix
ServiceMix
JAXWS
Adaptors
Cassandra 1
Cassandra 2
Cassandra 3
Cassandra 4
Cassandra 5
Cassandra 6
ActiveMQ Master
ActiveMQ Slave
Network of Brokers
ActiveMQ Master
ActiveMQ Slave
ServiceMix
ServiceMix
ServiceMix

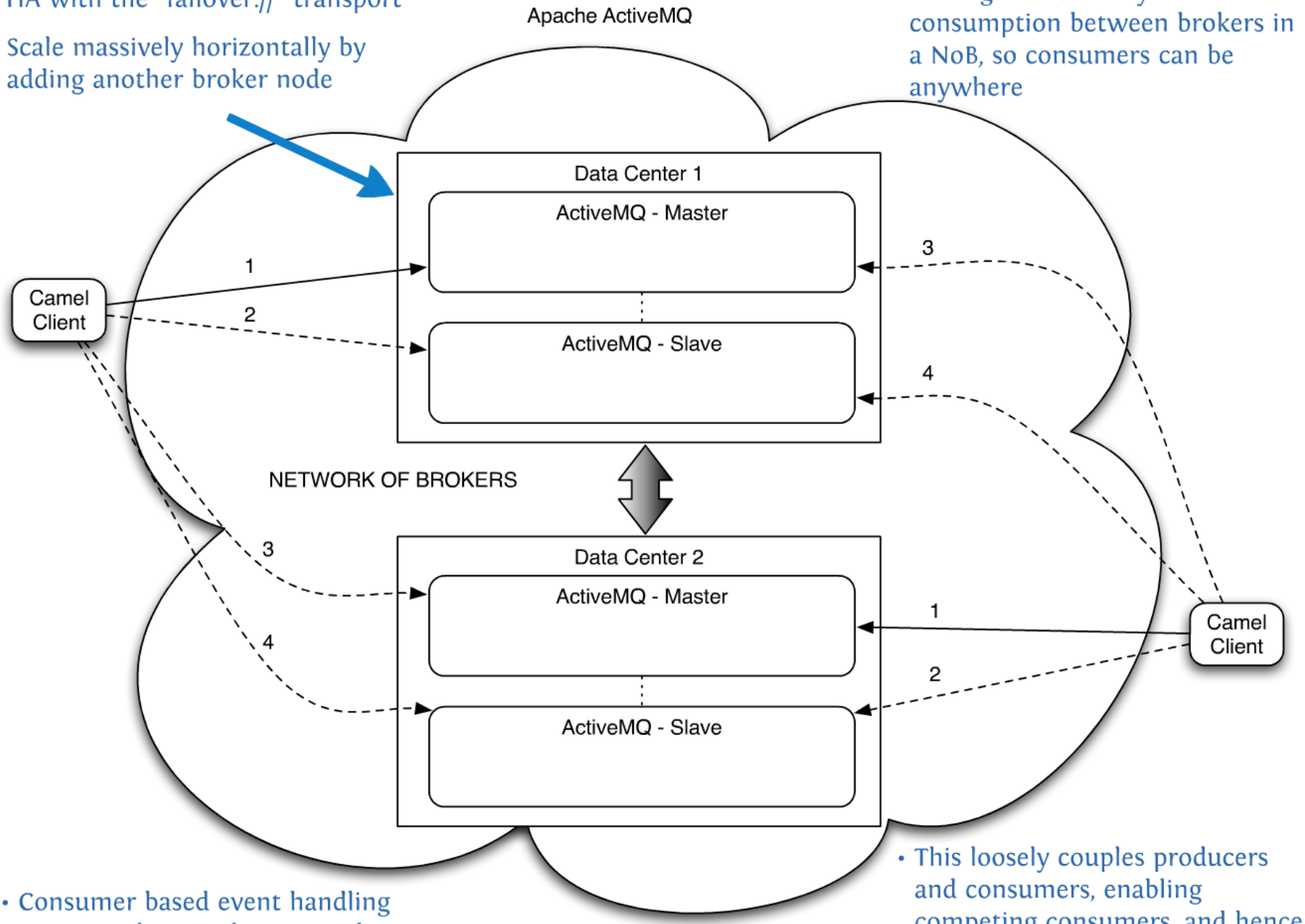
ActiveMQ

- Java Messaging Service (JMS)
- Persistent Messaging
- Removes Point To Point / Consumer Polling Strategy



- HA with the "failover://" transport
- Scale massively horizontally by adding another broker node

- Messages flow freely for consumption between brokers in a NoB, so consumers can be anywhere



- Consumer based event handling means endpoints do not need to know where each other lives

- This loosely couples producers and consumers, enabling competing consumers, and hence failover routes

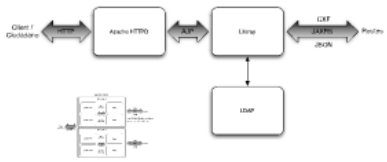
DINARDAP

SINARDAP - SISTEMA NACIONAL DE DATOS PÚBLICOS

WEB TIER

LIFERAY

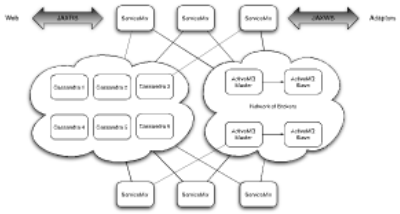
Front end - Client access for registration of cedulas and data access. This is the direct point for which the end user will access all of their information.



ROUTES TIER

ESB

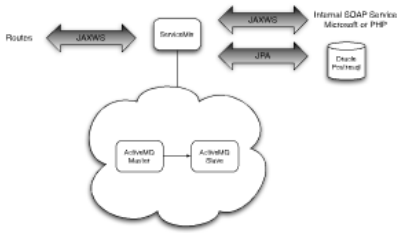
The main conduit for handling requests and mediation between the web tier and the external system adaptors. This tier handles guaranteed messaging and data caching.



ADAPTOR TIER

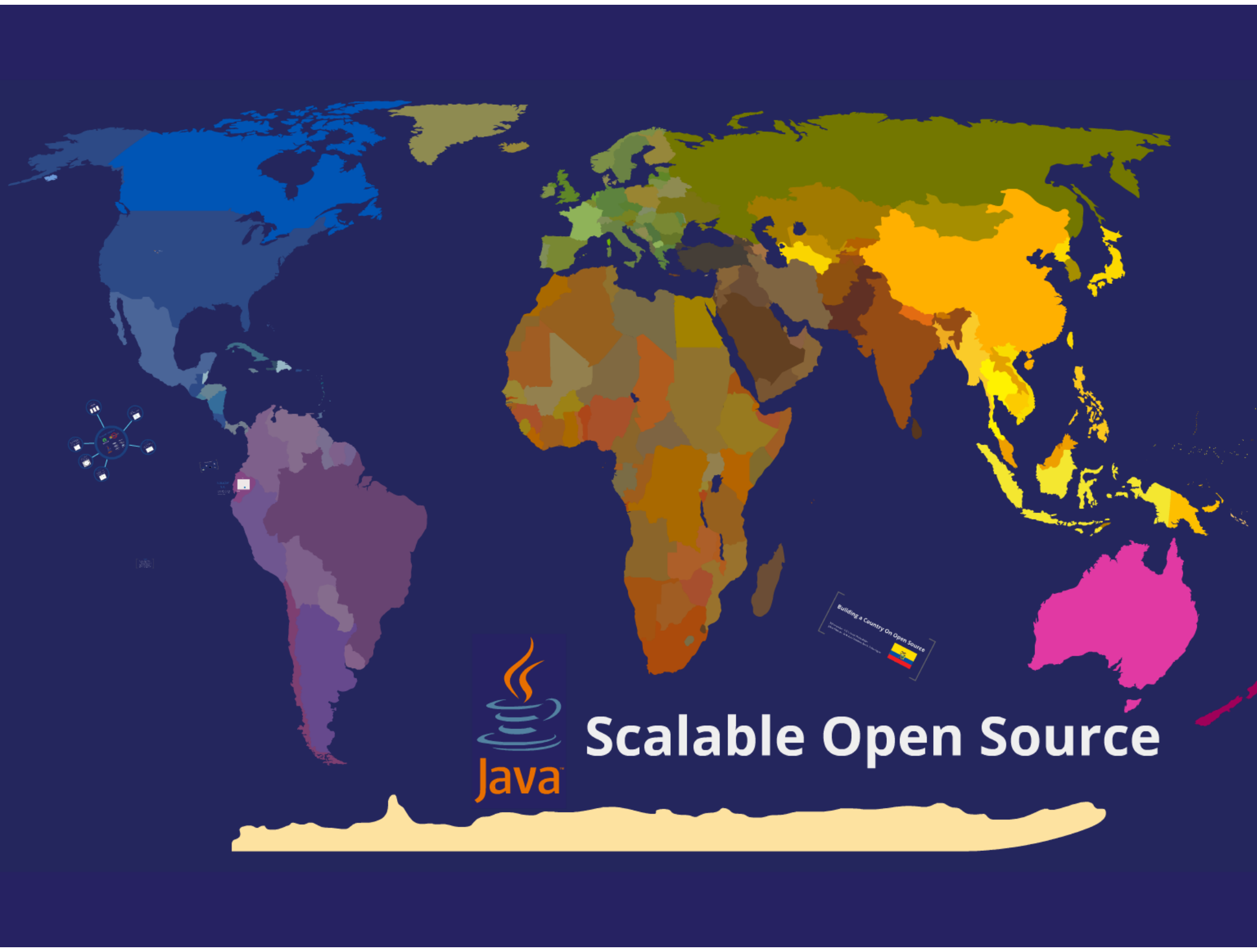
ESB ENDPOINTS

Thee are the adaptors that live in locations near the government services. These are endpoints that translate JAXRS requests to the government services; be it direct database access or WS Proxy



Result?

- Able to handle over 40 Million transactions per day
- Full failover and can process transactions when almost any server is down - no single points of failure
- Ecuadorian government found the system to be an immense success and a tribute to Java and leveraging open source
- The system has become a poster-child for other Latin American countries, such as Chile, Peru, and Colombia to examine their own internal systems and create their own initiative
- The system's success has enabled other Ecuadorian government to further build out applications based on Java and open source


Building a Country On Open Source
Java
Scalable Open Source