EVENT DRIVEN ARCHITECTURE:

A Reference Model for Market-Wide Half-Hourly Settlement

Author: Architecture Working Group
Date: January 2021
Version: 1.0

DEFINITION

EDA is an architectural pattern for the production, management and consumption of data events. EDA enables the creation of a responsive/reactive, asynchronous, non-blocking/concurrent and de-coupled systems topology.

MHHS is a distributed network of services and roles that requires constant communication of data for operational purposes.

An Event captures an immutable statement of fact, as represented by the data to be distributed between the organisations of the MHHS market model.

In MHHS, this could be a registration appointment or de-appointment, etc.

EDA is applicable within a connected topology of entities which must transmit events between loosely coupled software components.

In MHHS, these are the IT systems used to execute business processes for Registration, Metering & Data services, Settlement, Distribution and Supplier organisations.

Required Capabilities

Approx. 27 Registration Services must maintain operational data integrity and consistency across approx. 35 Data Services, 85 Metering Services, 17 DNO's and 60 Suppliers.

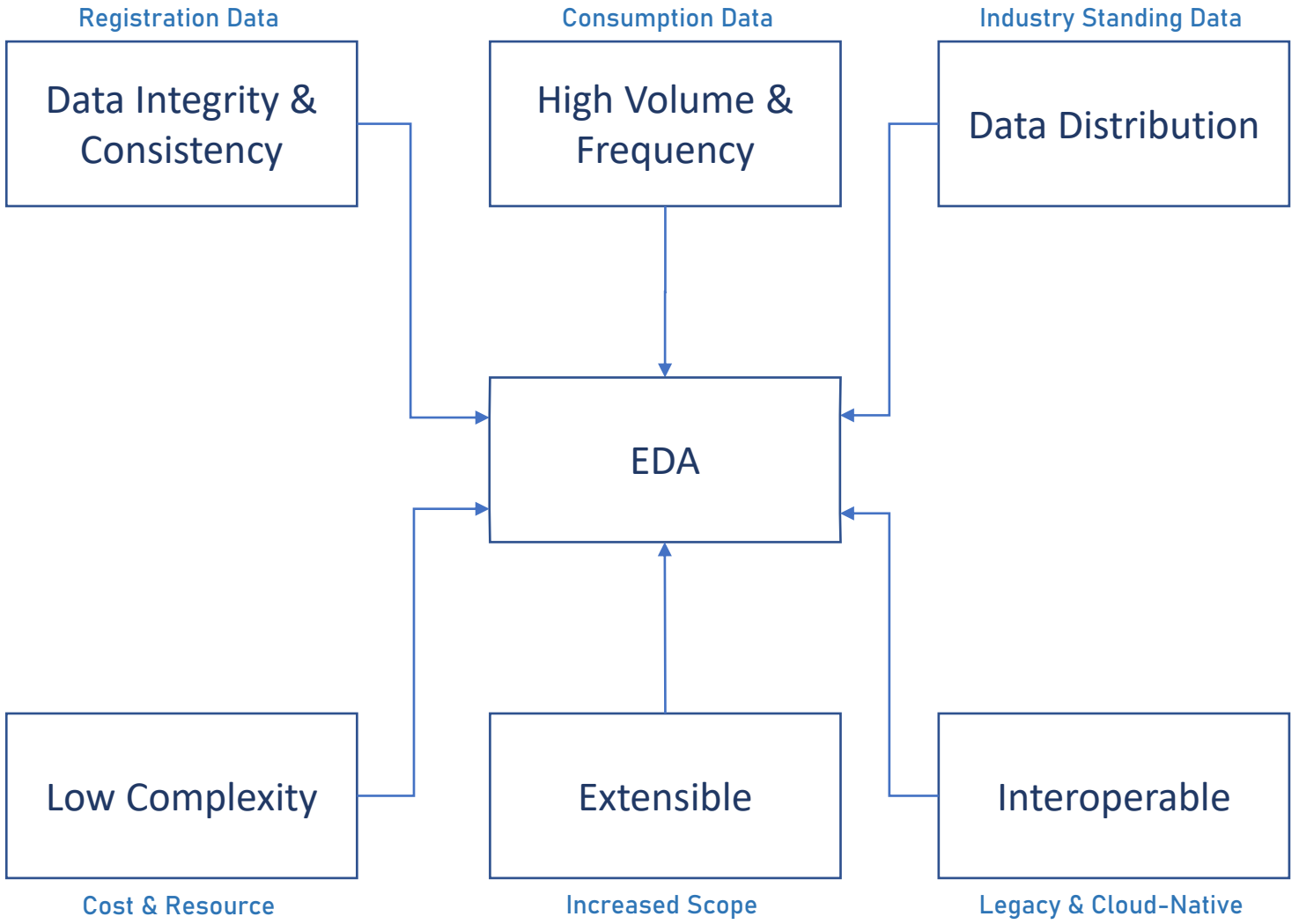
The Data Services must provide an approx. total of 32 Million daily, or approx. 14 Billion annual Consumption events to Central Settlement.

Industry Standing Data must be provided by Central Settlement to any of the approx. 400 active market participants when ISD is modified.

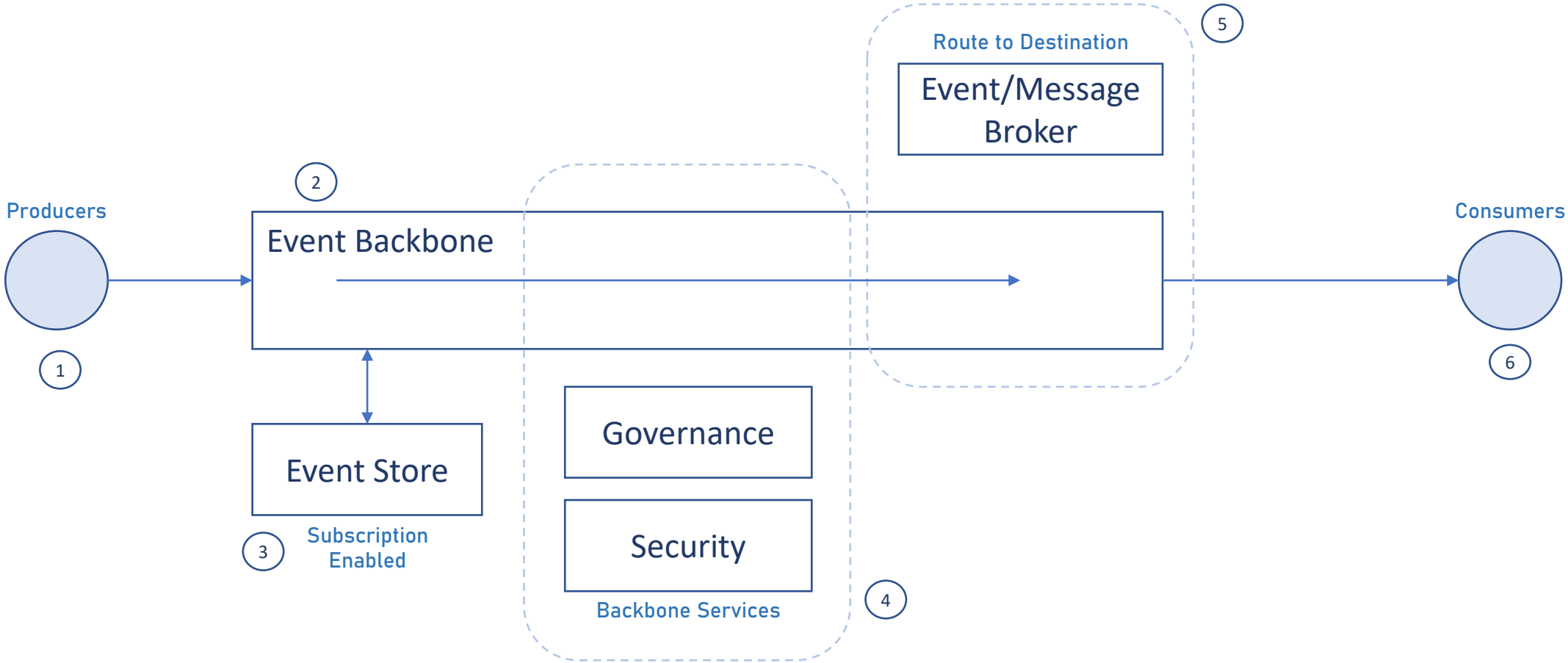
Avoid creation of bespoke point-to-point interfaces which duplicate data distribution costs and effort across the market participants. Avoid creating unnecessary business logic.

Provide a data distribution mechanism that is capable of meeting the future needs and ambition of the industry.

A loosely coupled architecture enables both legacy systems and newer cloud-based solutions to be agnostic of each other. This allows for technology transition to be accomplished over time.



BASELINE



BASELINE

1. Data producers generate immutable events and publish them into event topics within the backbone via a variety of possible technology options.

2. The Event Backbone supports publish/subscribe communications, event log (store), monitoring, policy enforcement and other governance & security process via event stream processing.

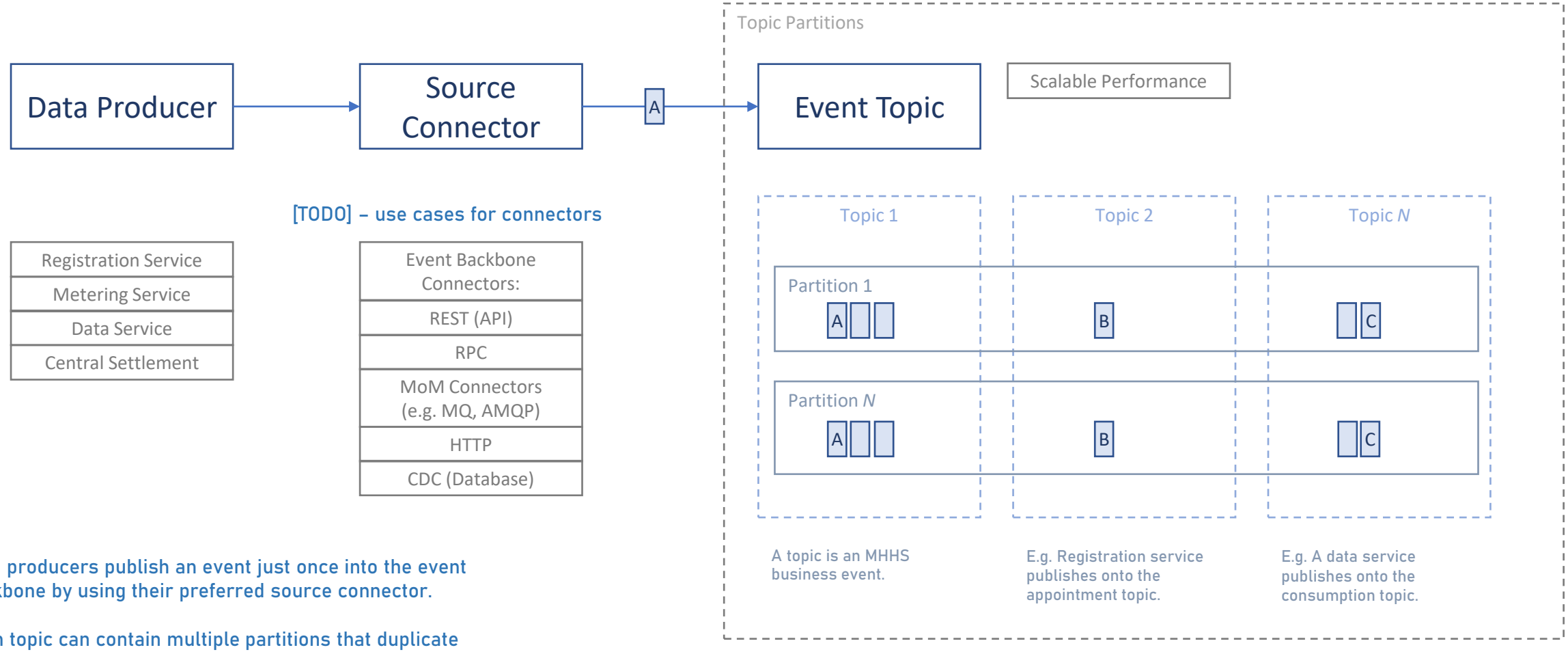
3. The Event Store provides a means to persist events for a period of time. These may be replayed via a subscription, for business recovery or for other valid data distribution purposes.

4. Governance and security are executable processes that ensure event validity, technical completeness and relevant security standards & protocols are correct.

5. Broker middleware decouples the source and destination for data distribution and provides the routing logic to ensure that subscribers can receive the correct data.

6. Data consumers handle events from the subscriptions made available to them by the broker. The events may be retrieved by consumers or they may be notified to consumers. Events may be distributed in real-time or by a persistent queue that guarantee's eventual delivery.

1 Data Producers



[TODO] – use cases for connectors

Registration Service
Metering Service
Data Service
Central Settlement

Event Backbone Connectors:
REST (API)
RPC
MoM Connectors (e.g. MQ, AMQP)
HTTP
CDC (Database)

Data producers publish an event just once into the event backbone by using their preferred source connector.

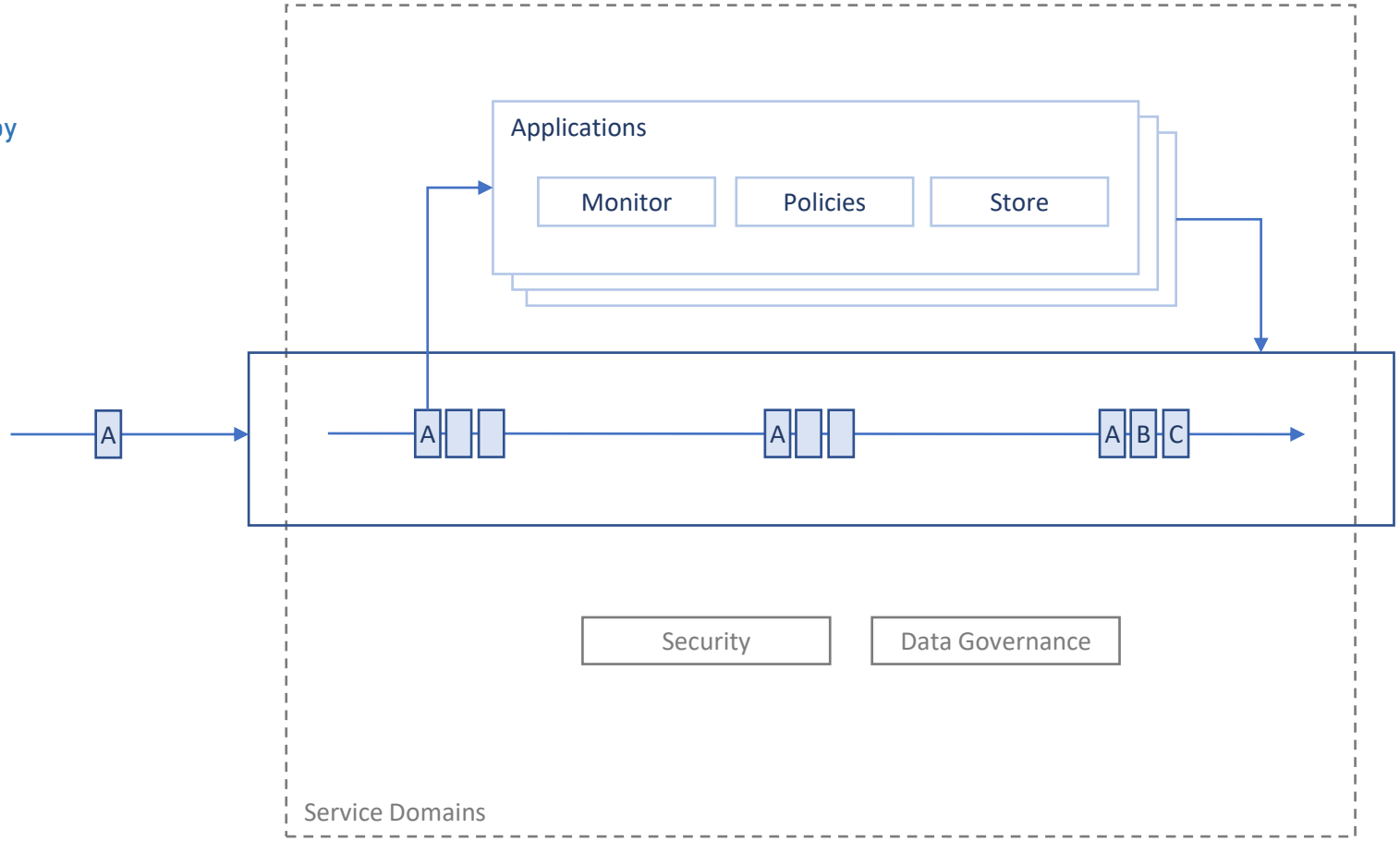
Each topic can contain multiple partitions that duplicate events, in order to provide greater performance and throughput.

2 Event Backbone (with Event Stream Processing)

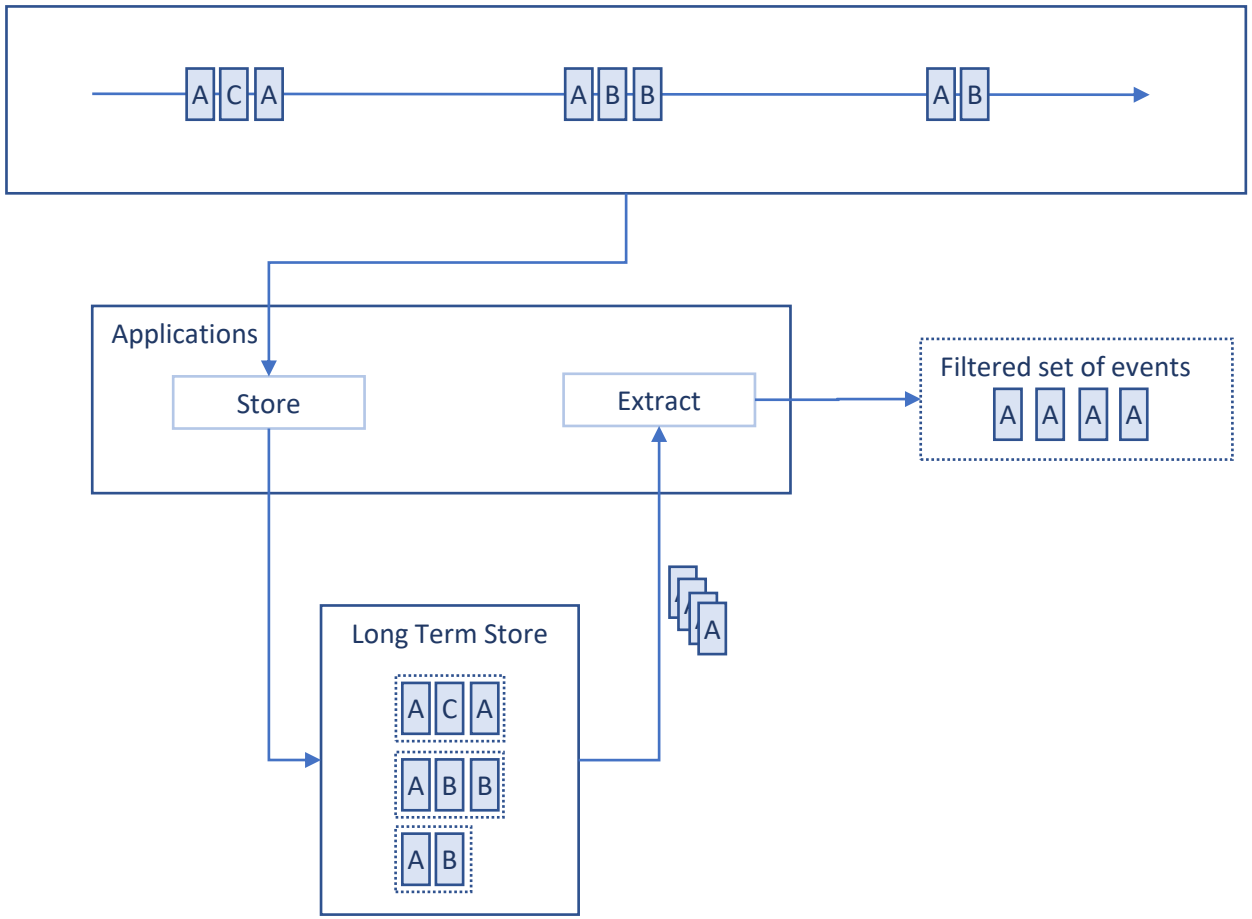
Event inspection can be performed in real-time by scalable demand-driven applications such as cloud-native apps & microservices.
(e.g. to cater for the temporary scale-out processing needs of peak consumption data)

Event sources inject MHHS data into topics. Monitoring and policy enforcement of additional security & data governance rules can be processed. Events can be persisted to the event store.

Service domains can be extended in future to cater for evolving industry requirements.



3 Event Store



Broker live short-term store used to notify consumers of events as they occur.

Processes can be provided that extract from the store for purposes such as:

- Recovery of consumer systems
- Extraction of filtered sets of events for data quality or fault analysis
- Authorised 3rd party data provision
- Any new business services in future

Secure read-only long term storage of events.
Events organised to support processing applications filtering and extracting sets of events.

4 Governance & Security

Events can be technically validated. They can be compared to allowed schema's and action can be taken in case of issues.

GDPR or sensitive data can be encrypted and access to topics storing sensitive data can be restricted and managed appropriately.

Data policies can be enforced by performing actions on events as they are published.

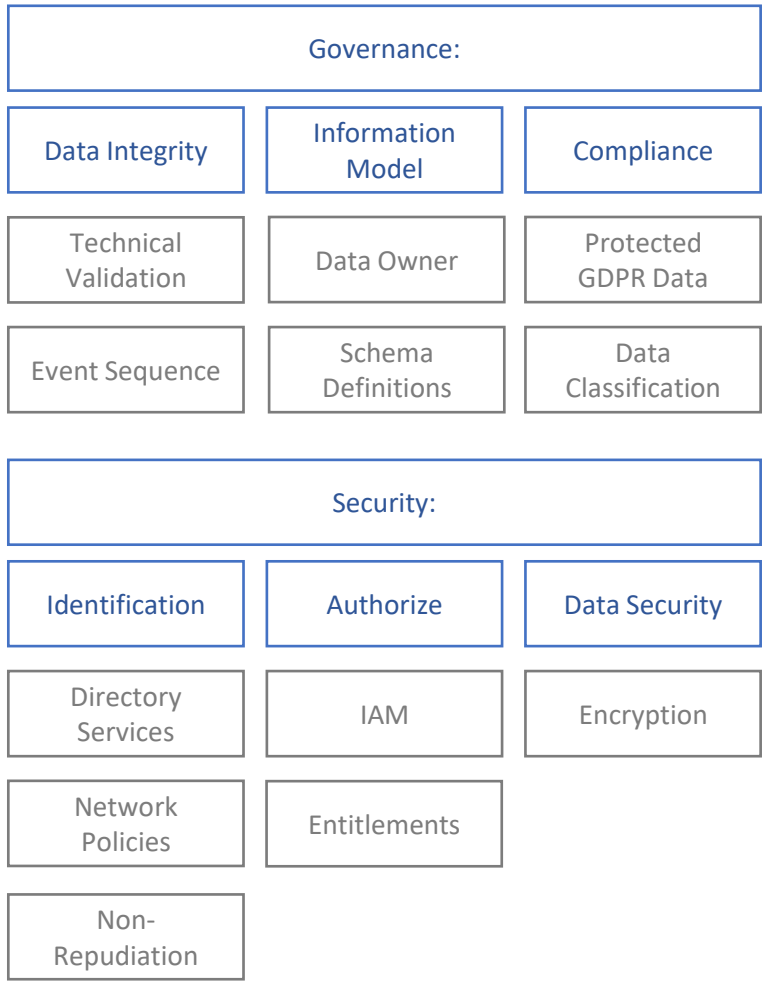
Data producers must be identified, authorized, and their data entitlements for publishing into topics should be validated.

Network security can be used to control inbound connections.

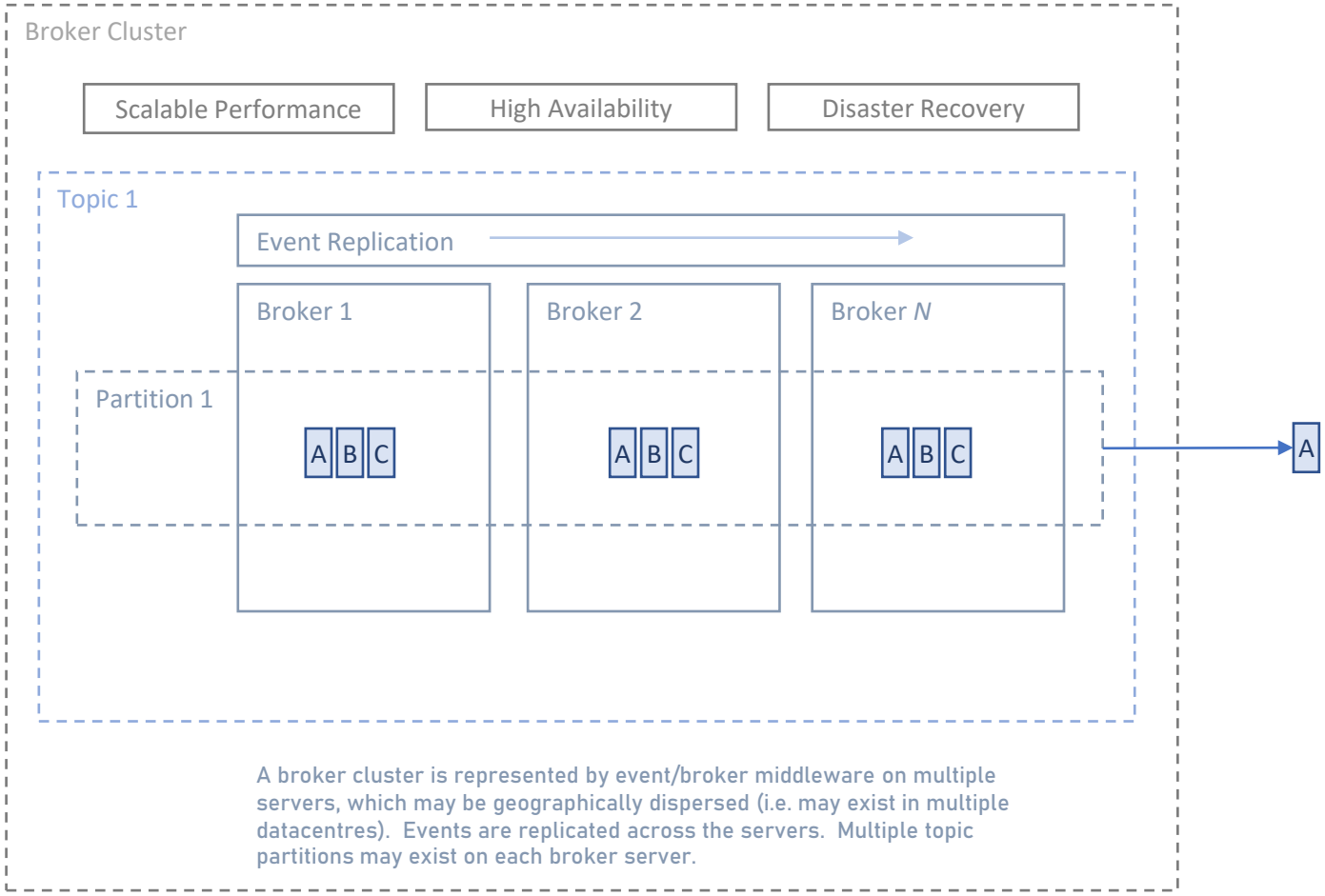
Data must be secure in transit (for example, using TLS) and at rest in the store.

Only valid data consumers may access events from restricted topics.

Entitlements to resources (such as the schema manager) can be managed, for example through ACLs.



5 Event/Message Broker

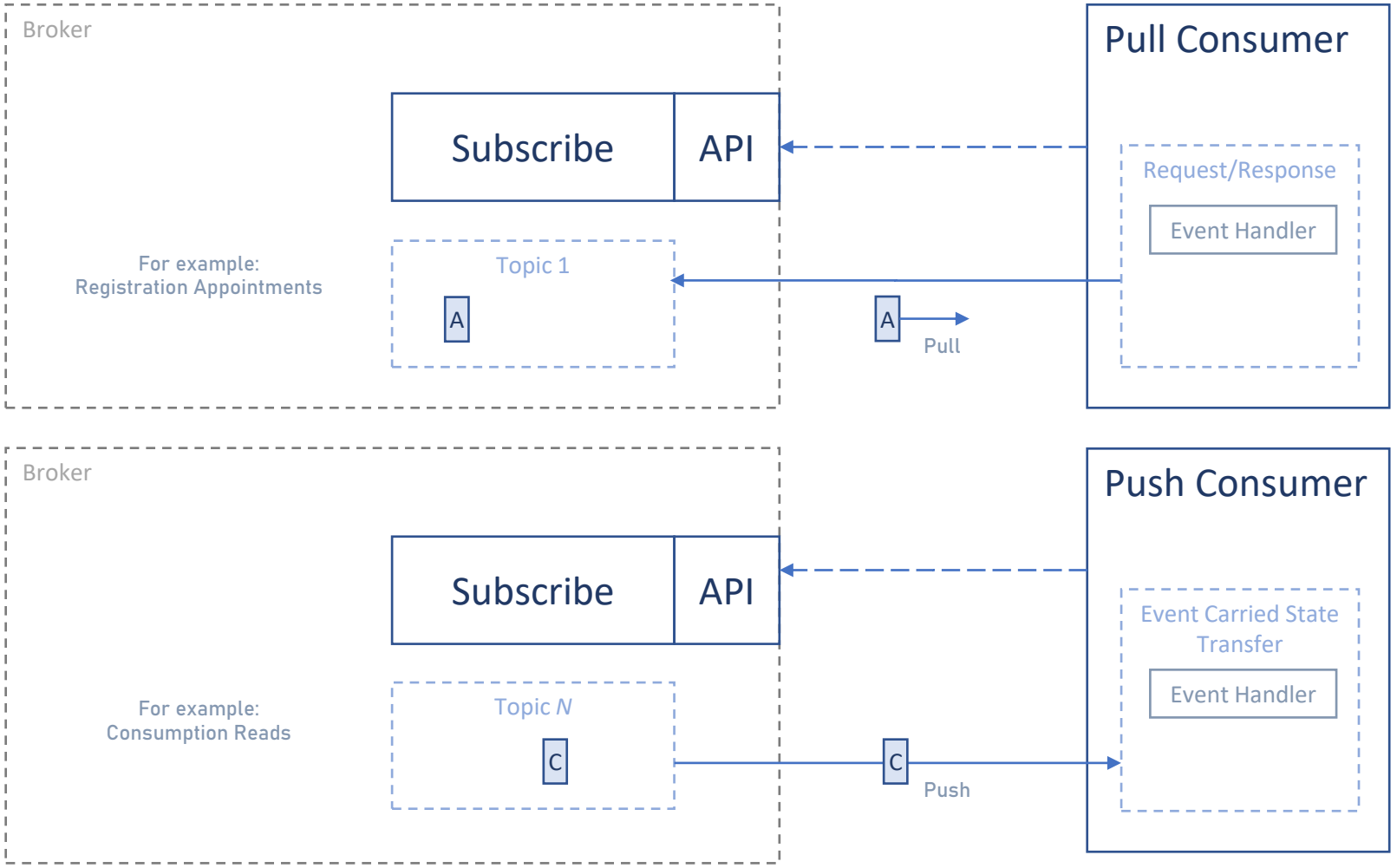


The event backbone utilises clusters of broker servers for high performance, parallelism, fault tolerance and high availability.

The broker cluster configuration provides event replication for each Topic.

The configuration also allows for either an active-active or active-passive operating mode for the cluster.

6 Data Consumers



A consumer must subscribe to events (a topic) by issuing a subscribe to the broker. Each valid subscription results in access to the required sequence of events.

A consumer creates an event handler that can request (pull) events from the topic so as to control the rate at which events are processed. This mode can emulate a batch process.

Registration Service
Metering Service
Data Service
Central Settlement
Supplier
Distributor

A consumer creates an event handler which can receive events that are pushed from the broker.

The consumer does not have to make requests but they must ensure they are available. This mode operates in real-time.

Each topic or partition within a topic contains data events which need to be routed to specific MHHS roles or services. Events are retained in the topic (event log) for a configured duration.