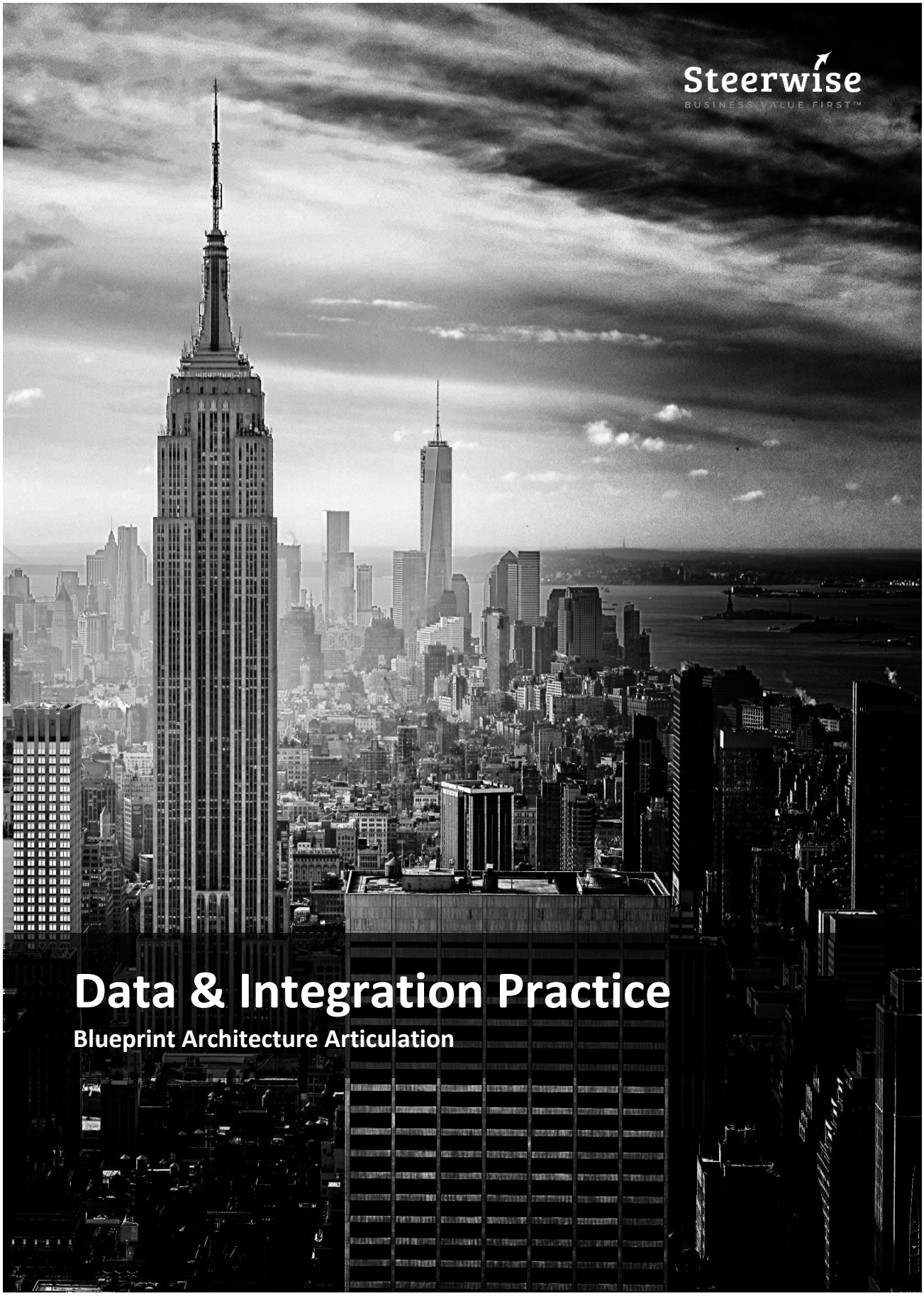

Steerwise
BUSINESS VALUE FIRST™
Data & Integration Practice
Blueprint Architecture Articulation

Enterprise Data Platform

Blueprint Architecture

Table of Contents

1.0 Enterprise Data Platform Blueprint Architecture

1.1 EDP Blueprint Architecture Overview

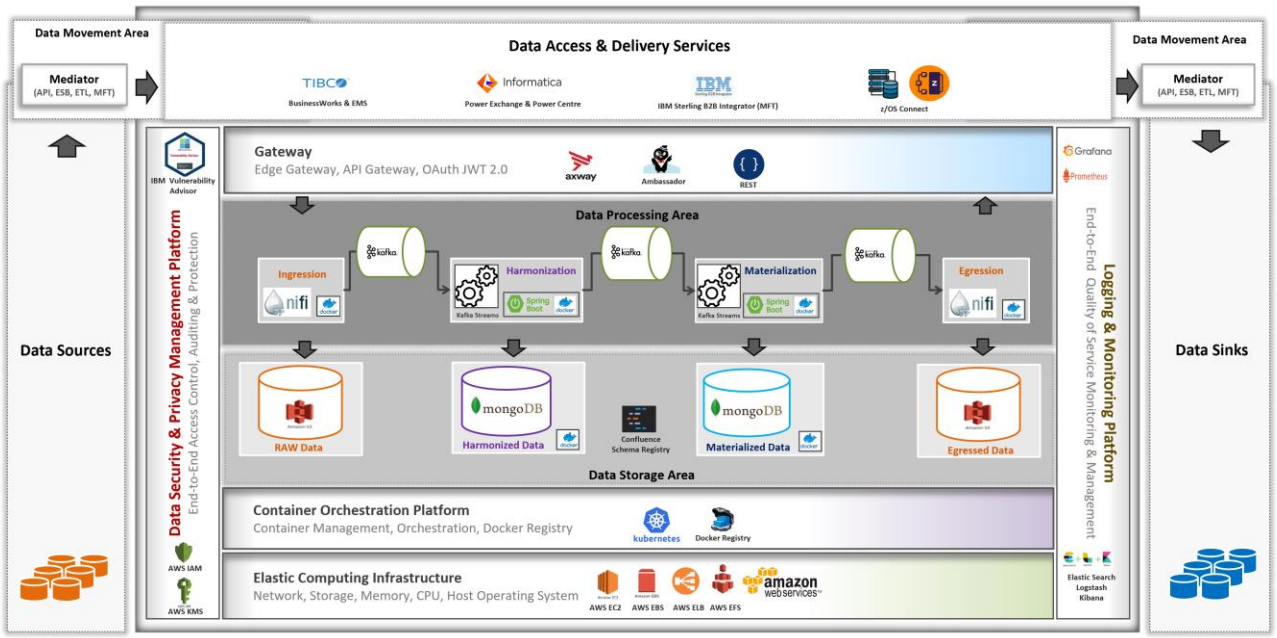
To successfully realize the EDP Reference Architecture, a blueprint of cloud-native based digital technology architecture has been defined. Blueprint Architecture identifies the key vendor technology platforms & tools to meet the immediate needs of the business.

Key technology components that are prioritized & scoped by TFS for the current phase are:

- **Container Infrastructure & Orchestration**
- **Data Storage & Persistence Services**
- **Data Processing Pipeline Services**
- **Logging & Monitoring Services**
- **Data Security Services**
- **Gateway Services**
- **Data Access & Delivery Services**

1.2 EDP Blueprint Architecture

The **EDP Reference Architecture** is realized through the following **Blueprint Architecture** to implement the initial set of requirements by identifying EDP component specific Vendor Technology Platforms & Tools as recommended and chosen by TFS.



The above **Blueprint Architecture** leverages existing Vendor Technology Platforms & Tools at TFS where appropriate and introduces advanced Vendor Technology Platforms & Tools where necessary to fulfill the prioritized requirements.

The **Blueprint Architecture** will continue to evolve with additional set of Vendor Technology Platforms & Tools based on the ongoing & future requirements.

1.3 Container Infrastructure & Orchestration

1.3.1 AWS Elastic Computing Infrastructure

- Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud.
- Preconfigured templates for instances, known as Amazon Machine Images (AMIs), that package the bits needed for server (including the operating system and additional software) customizable to enterprise specific standards.
- Various configurations of CPU, memory, storage, and networking capacity for the instances, known as instance types.
- Persistent storage volumes for data using Amazon Elastic Block Store (Amazon EBS), known as Amazon EBS volumes.
- Virtual networks can be created that are logically isolated from the rest of the AWS cloud, and that can optionally connect to on-premise enterprise network, known as virtual private clouds (VPCs)

1.3.2 Kubernetes Container Orchestration

- Kubernetes provides a portable, extensible open-source platform for **managing containerized workloads and services**, that facilitates both declarative configuration and automation.
- It orchestrates computing, networking, and storage infrastructure on behalf of user workloads. This provides much of the simplicity of Platform as a Service (PaaS) with the flexibility of Infrastructure as a Service (IaaS) and enables portability across infrastructure providers.
- Managing multiple containers as one entity is a capability that logically groups a set of containers into one entity to co-locate main application with its helpers. This preserves the one-application-per-container model or can define a complex multi-tier application as a set of smaller entities working together.
- Container placement algorithm selects a specific host for a specific container or a set of containers using different rules, including current load of the hosts, colocation constraints and availability constraints.

- **Container replication** ensures that a specified number of equivalent containers ("replicas") are running at the time.
- Container **auto-scaling** automatically changes the number of running containers, based on CPU utilization or other application-provided metrics.
- **Volume management** is used to control persistent storage for containers. According to best practices, containers should be stateless and files in a container should be ephemeral, so that when a container crashes and gets restarted the changes to the files will be lost.
- **Networking** plays a significant role for container orchestration to isolate independent containers, connect coupled containers and provide access to containers from external clients.
- **Service discovery** allows containers to discover other containers and establish connections to them.
- **Load balancing** works in conjunction with container replication, scaling and service discovery. Load balancing is a dedicated service that knows which replicas are running and provides an endpoint that is exposed to clients. Connections to the exposed endpoint are distributed over running replicas using different methods such as DNS round-robin.

1.3.3 Docker Container Registry

- The Docker Registry is a stateless, highly scalable application that provides the ability to **store and distribute Docker images**.
- Docker Registry allows us to tightly control where the docker images are being stored, fully own the image distribution pipeline and integrate image storage and distribution tightly into the development & deployment workflow.
- A container repository is critical to agility. It eliminates the need for re-building image on every machine and ensures consistency across all instances.

1.4 Big Data Storage & Persistence

1.4.1 AWS S3 Object Store (Raw Data & Egressed Data)

- Amazon S3 provides the ability to store and retrieve any amount of data, at any time, from anywhere on the web with full **versioned history**.
- Amazon S3 provides access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure.
- Amazon S3 provides easy-to-use management features to organize data into buckets and configure finely-tuned access controls to meet specific business, organizational, and compliance requirements.
- Amazon S3 offers a range of storage classes designed for different use cases. These include S3 Standard for general-purpose storage of frequently accessed data; S3 Intelligent-Tiering for data with unknown or changing access patterns; S3 Standard-Infrequent Access (S3 Standard-IA) and S3 One Zone-Infrequent Access (S3 One Zone-IA) for long-lived, but less frequently accessed data; and Amazon S3 Glacier (S3 Glacier) and Amazon S3 Glacier Deep Archive (S3 Glacier Deep Archive) for long-term archive and digital preservation.

1.4.2 Mongo DB Object Store (Harmonized & Materialized Data)

- **Mongo DB is a Document-oriented** – a NoSQL type Object Store. This makes MongoDB very flexible and adaptable to real business world situation and requirements. Needless to say that schema governance controls, data access, complex aggregations, and rich indexing functionality are not compromised in any way. Without downtime one can modify the schema dynamically.
- **Replication** - MongoDB can provide high availability with replica sets. A replica set consists of two or more mongo DB instances. Each replica set member may act in the role of the primary or secondary replica at any time. The primary replica is the main server which interacts with the client and performs all the read/write operations. The Secondary replicas maintain a copy of the data of the primary using built-in replication. When a primary replica fails, the replica set automatically switches over to the secondary and then it becomes the primary server.
- **Load balancing** - MongoDB uses the concept of sharding to scale horizontally by splitting data across multiple MongoDB instances. MongoDB can run over multiple servers, balancing the load and/or duplicating data to keep the system up and running in case of hardware failure.

1.4.3 Confluent Schema Registry (Meta Data)

- **Schema as a first-class citizen**. The structure of the data, or its schema, may either be specified and uploaded through a REST interface or by the message producer itself. Each piece of data that is passed through the system is in turn stamped with the name and version of the schema it represents.
- With data schemas, every single piece of data traveling through the system is completely discoverable, enabling us to build systems that can easily adapt as the data changes.
- Schema Registry provides a serving layer for the metadata. It provides a RESTful interface for storing and retrieving data object schemas.
- Schema Registry stores a **versioned history of all schemas**, provides multiple **compatibility settings** and allows evolution of schemas according to the configured compatibility settings.

1.5 Data Processing Pipeline Services

1.5.1 Apache Nifi Data Flow Processing

- NiFi is an enterprise integration and **dataflow automation tool** that provides the ability to send, receive, route, transform, and sort data, as needed, in an automated and configurable way.
- **Visual Command and Control**: NiFi enables visual establishment & management of dataflows in real-time.
- **Data Provenance**: NiFi automatically records, indexes, and makes available provenance data as objects flow through the system even across fan-in, fan-out, transformations, and more. This information becomes extremely critical in supporting re-play, compliance, troubleshooting, optimization, and other scenarios.
- **Flexible Scaling Model**: NiFi is designed to scale-out through the use of clustering many nodes together as described above. If a single node is provisioned and configured to handle hundreds of MB per second, then a modest cluster could be configured to handle GB per second.
- **Data Buffering w/ Back Pressure and Pressure Release:** NiFi supports buffering of all queued data as well as the ability to provide back pressure as those queues reach specified limits or to age off data as it reaches a specified age (its value has perished).
- **Flow Specific QoS** (latency v throughput, loss tolerance, etc.): NiFi enables the fine-grained flow specific configuration to manage QoS.

1.5.2 Kafka Message Queue

- Kafka Messaging platform has three key capabilities:
  - **Publish** and **Subscribe** to streams of records (**Topics)**, similar to a message queue or enterprise messaging system.
  - Store streams of records in a fault-tolerant durable way.
  - Process streams of records as they occur.
- Kafka is generally used for two broad classes of applications:
  - Building **real-time streaming data pipelines** that reliably get data between systems, between applications & between processing stages within an application.

- Building real-time streaming applications that transform or react to the streams of data

1.5.3 Kafka Stream Processing

- Kafka Streams enables building streaming applications, specifically applications that transform input Kafka topics into output Kafka topics (or calls to external services, or updates to databases, or whatever) in a distributed and fault-tolerant way.
- Kafka Streams facilitates complex transformations, aggregations off of streams or join streams together, handling out-of-order data, reprocessing input as code changes, performing stateful computations, etc.

1.6 Logging & Monitoring Services

1.6.1 ELK Stack

- The ELK stack consists of **E**lasticsearch, **L**ogstash, and **K**ibana.
- **Elasticsearch** is a log data store, search and analytics engine.
  - Real-time data and real-time analytics, with the ability to perform super-fast data extractions from virtually all structured or unstructured data sources.
  - Horizontally Scalable, high-availability, multi-tenant
  - Full text search based on Lucene search engine with multi-language support & geolocation support.
  - Document oriented data store in JSON format
- **Logstash** is a server-side data processing pipeline that ingests data from multiple sources simultaneously, transforms it, and then sends it to a "stash" like Elasticsearch.
  - Logstash is a tool for log data intake, processing, and output. This includes virtually any type of logs: system logs, webserver logs, error logs, and app logs.
  - Logstash will serve as the workhorse for storage, querying and analysis of system logs and extract the *relevant, high-value* data from the logs.
- **Kibana** lets users visualize data with charts and graphs in Elasticsearch.
  - Kibana is dashboard for log-data with point-and-click pie charts, bar graphs, trendlines, maps and scatter plots.
  - It provides the ability to visualize trends and patterns for data that would otherwise be extremely tedious to read and interpret & send alert notifications.

1.6.2 Prometheus

- Prometheus is a white box monitoring and alerting system that is designed for large, scalable environments that fits both machine-centric monitoring as well as monitoring of highly dynamic microservice-oriented architectures.
- A **multi-dimensional data model**, with **time series** where data can be sliced and diced along multiple dimensions like host, service, endpoint and method.

- Operational simplicity, to setup & configure monitoring anywhere as needed.
- Scalable and decentralized, for independent and reliable monitoring.

1.6.3 Grafana

- Grafana provides the ability to query, visualize, alert on and understand the performance metrics no matter where they are stored. Create, explore, and share dashboards with your team and foster a data driven culture.
- Grafana is designed for analyzing and visualizing metrics such as system CPU, memory, disk and I/O utilization.
- Elegant graphics for data visualization with Fast and flexible graphs and support for notifications via Slack, PagerDuty, and more.

1.7 Data Security Services

1.7.1 AWS IAM

- AWS Identity and Access Management (IAM) enables to manage access to AWS services and resources securely. IAM can be used to create and manage AWS users and groups and use permissions to allow and deny their access to AWS resources.
- Fine-grained access control to AWS resources
- Multi-factor authentication for highly privileged users
- Integrate with corporate directory

1.7.2 AWS KMS

- AWS Key Management Service (KMS) makes it easy to create and manage keys and control the use of encryption across a wide range of AWS services and applications. AWS KMS is a secure and resilient service that uses FIPS 140-2 validated hardware security modules to protect keys.
- Encrypt data across applications through centralized key management
- Enforce Compliance with Built-in auditing

1.7.3 Vulnerability Advisor

- Continuously monitors, scans & runs security checks on containers running in the distributed cluster.
- Provides an evaluation report with recommendations to fix security violations that is based on standardized security practices.

1.8 Gateway Services

1.8.1 Ambassador

- Ambassador is a Kubernetes-native, microservices focused API gateway built on the Envoy Proxy.
- Ambassador can also be used to handle the functions of a Kubernetes ingress controller and load balancer
- Ambassador supports a wide variety of features needed in an edge proxy, e.g., rate limiting, distributed tracing, dynamic routing, metrics, and more.
- Ambassador also includes an authentication API where external authentication service can be plugged in.

1.8.2 Axway

- Axway API Gateway manages, delivers, and secures enterprise APIs, applications, and consumers.
- API bi-directional transformation with support for wide range of protocols, data formats, and standards (for example, REST-to-SOAP, XML-to-JSON, and HTTP-to-JMS).
- API lifecycle management from creation to end-of-life
- Identity management with configurable authentication & authorization and support for integration with existing third-party Identity Management (IM) infrastructures.
- Centralized API policy management
- The API Gateway Analytics console provides monitoring, auditing and reporting on usage across all entry points and creates comprehensive reports to meet operational and compliance requirements.
- API Gateway protects services from unanticipated traffic spikes by smoothing out traffic.

1.9 Data Access & Delivery Services

1.9.1 TIBCO BusinessWorks & EMS

- TIBCO **BusinessWorks** provides a comprehensive integration platform to address business problems of varying complexity using the following integration styles:
  - Batch-oriented - provides non real-time integration for endpoints such as databases or files and uses records for data abstraction.
  - Process-oriented - provides real-time integration for endpoints such as application APIs and adapters and uses APIs, objects, and messages for data abstraction.
  - Service-oriented - provides real-time integration for endpoints such as web services and APIs and uses services and messages for data abstraction.
  - Resource-oriented - provides real-time integration for endpoints such as mobile or web applications and APIs and uses resources for data abstraction.
- A TIBCO Enterprise Management Service (EMS) server provides messaging services for applications that communicate by monitoring queues on a high-performance messaging backbone.
- EMS supports real-time decision-making by scaling horizontally across the organization to unlock data in diverse databases and applications and support fast, event-driven execution of business operations.

1.9.2 Informatica Power Center & Power Exchange

- **Informatica PowerCenter** aids extracting data from its source, transforming it as per business requirements and loading it into a target data warehouse.
- Informatica PowerCenter is an enterprise data integration suite with high availability being fully scalable and high performance, providing the foundation for all the data integration across the enterprise.
- **Informatica® PowerExchange®** is a family of products that enables IT organization to retrieve all sources of enterprise data by accessing mission-critical operational data where it's stored and delivering it where and when it's needed, maximizing the business value of data.
- PowerExchange scales readily from batch access to real-time data integration on multiple platforms like Mainframe systems, Midrange system & Message-oriented middleware.

- The PowerExchange Change Data Capture Option captures changes in a number of environments as they occur, so up-to-the-minute data can be delivered to the business.

1.9.3 Managed File Transfer

- IBM MFT is a managed file transfer solution which automates and secures file transfers using a centralized enterprise-level approach in a managed and auditable way, regardless of file size or the operating systems used.

- MFT can process high volumes of file transfers for enterprises by load balancing processes across multiple systems through clustering technology with support for active-active automatic failover for disaster recovery.

1.9.4 z/OS Connect

- IBM® z/OS® Connect enables enterprises to unleash existing market-differentiating assets (data & services) hosted on IBM Z® platform with RESTful APIs in a simple and intuitive way.

- IBM® z/OS® Connect Enterprise Edition provides a framework that enables z/OS based programs and data hosted on CICS®, IMS™, WebSphere® MQ, DB2®, and Batch systems to participate fully in the new API economy for mobile and cloud applications.