**Adventure Works Cycles Bigdata Solution/Design**

Based on the case study, Adventure Works Cycles is looking to modernize and expand their data infrastructure to support their global growth, improve customer experience, and enable advanced analytics capabilities.

Here's a proposed solution design and implementation using the specified tools.

|  |  |
| --- | --- |
| **Data Infrastructure** | |
| **Storage and Processing** | **1.** **HDFS** (Hadoop Distributed File System): HDFS provides scalable and fault-tolerant storage for large volumes of structured and unstructured data.  **2.** **Spark**: High-performance processing engine for big data analytics and machine learning. Implement Spark jobs to extract, load, and transform data from various sources (e.g., on-premises databases, flat files) into the data warehouse.  **3.** **Hive**: Data warehouse solution for querying large datasets using SQL-like syntax. It replicates the RDBMS (Relational Database Management Systems). Thus it stores Structured Data in table format. It is an ETL tool extract the data from HDFS.  **4.** **HBase**: Column-family database for storing large tables with frequent read and write operations. HBase is a NO-SQL database, it satisfies the property of handling both unstructured data and semi-structured data. |
| **Messaging** | **Kafka**: Real-time messaging platform for streaming data ingestion and distribution.  \* Use Kafka, a distributed streaming platform, to ingest real-time data from the connected bicycles and social media hashtags.  \* Implement Kafka producers to collect data from the bicycle computers and social media APIs, and send it to Kafka topics.  \* Use Kafka consumers to process the real-time data and store it in the appropriate data stores (e.g., HBase for bicycle telemetry data, HDFS for social media data). |
| **Querying and Visualization** | **1.** **Impala**: Interactive SQL query engine for fast data exploration and reporting.  **2. MySQL:** Relational database management system for structured data storage and querying. |
| **Resource Management** | **YARN**(Yet Another Resource Negotiator): Resource manager for scheduling and allocating cluster resources  Configure YARN to allocate resources dynamically based on the workload requirements, ensuring optimal utilization of the cluster. |
| **Orchestration** | **Zookeeper**: Coordination service for managing distributed systems |
| **Data Integration** | **Sqoop**: Data transfer tool for importing data from relational databases into Hadoop. |
| **Machine Learning and Analytics** | Use **Spark MLlib**, a distributed machine learning library, to build predictive analytics models for bicycle maintenance and fraud detection. |
| **Security** | 1.Encrypt data at-rest and in-transit using HDFS transparent encryption and SSL/TLS  2. Apply FID access control, such as cyberark  3. Role segregation |

|  |  |
| --- | --- |
| **Solution Design** | |
| **Website Product Images** | 1. Create a HDFS directory for storing product images.  2. Use HBase to store image metadata (e.g., name, size, description). |
| **Sales / Ordering System** | 1.Migrate the on-premises AdventureWorks2012 database to a distributed SQL database like MySQL or Impala. This will provide global availability and scalability for the sales and ordering system.  2. Implement high availability using replication or active/passive failover.  3. Use Kafka for real-time data streaming of orders. |
| **Data Analysis** | 1. Migrate the AdventureWorksDW database to a Hadoop-based data warehouse.  2. Use Hive for data querying and reporting.  3. Explore predictive analytics using Spark MLlib. |
| **Customer Service / Presales** | 1. Create a chatbot using natural language processing (NLP) models.  2. Store conversation history in HBase for quick access.  3. Use Kafka for real-time data streaming of chatbot interactions.  4. Implement a fraud detection system using machine learning algorithms and real-time data analysis (Spark MLlib) |
| **Social Media Analysis** | 1. Use Kafka to stream social media data.  2. Store hashtag tracking data in HBase.  3. Use Spark Streaming for real-time analysis of social media data |
| **Connected Bicycle** | 1. Store bicycle telemetry data in HDFS.  2. Use Spark for data processing and analysis.  3. Use Kafka for real-time data streaming of bicycle location and status.  4. Create a REST API using Spark for third-party application integration. |
| **Bicycle Maintenance Services** | 1. Store sensor data and usage information in HDFS.  2. Use machine learning algorithms to predict maintenance needs.  3. Notify bike owners via email or mobile app using Kafka. |
| **Business Continuity** | 1. Implement data replication for HDFS and HBase/Hive/MYSQL etc.  2. Establish a disaster recovery plan for all critical services.  3. Regularly test and update backup and recovery procedures |

**Rationale for Solutions:**

**HDFS and Spark:** Scalable and high-performance for storing and processing large amounts of data.  
**HBase**: Fast and durable for storing frequently accessed data.  
**Kafka**: Real-time and reliable for data streaming and distribution.  
**Hive and Impala**: Efficient for data querying and reporting.  
**MySQL**: Familiar and cost-effective for structured data storage.  
**Sqoop**: Flexible and efficient for data integration.  
**Zookeeper**: Essential for coordinating and managing distributed systems.