Go Data Mover application package. WA Brown aes@3pp.com

1. bridge.go: Pub/Sub to Kafka Bridge

- **Purpose**: Acts as a service that bridges Google Pub/Sub (message subscription system) with Apache Kafka (message queuing system).
- **Key Components**:
  - **Bridge struct**:
    - Contains a subscriber (Pub/Sub), a producer (Kafka), logger, and metrics.
  - **Metrics struct**:
    - Tracks metrics like messages processed, failed Pub/Sub/Kafka messages, and the last processed time.
  - **NewBridge**:
    - Initializes the bridge service with a subscriber, producer, and logger.
  - **Start Method**:
    - Processes messages from Pub/Sub and sends them to Kafka.
    - Records processing time, updates metrics, and logs results.
  - **Stop Method**:
    - Gracefully stops the subscriber and producer.
  - **GetMetrics Method**:
    - Exposes service metrics.
- **Interaction**: Processes messages from Pub/Sub, produces them to Kafka, and logs/report metrics.

2. subscriber.go: Google Pub/Sub Subscriber

- **Purpose**: Handles message retrieval from a Google Pub/Sub subscription.
- **Key Components**:
  - **Subscriber struct**:
    - Contains the Pub/Sub client, subscription, and logger.
  - **NewSubscriber**:
    - Creates a new Pub/Sub subscription client for a given project and subscription ID.
    - Validates if the subscription exists.
  - **Start Method**:
    - Listens for messages, invokes a provided MessageHandler, and processes Ack/Nack based on success.
  - **Close Method**:

- ◆ Gracefully closes the Pub/Sub client.
- **Interaction**: Provides messages to the bridge service, which then sends them to Kafka.

3. producer.go: Kafka Producer
- **Purpose**: Handles message production to an Apache Kafka topic.
- **Key Components**:
  - ○ **Producer struct**:
    - ◆ Contains the Kafka producer instance, topic name, and logger.
  - ○ **NewProducer**:
    - ◆ Creates and configures a Kafka producer using provided settings (e.g., bootstrap servers, security).
    - ◆ Starts a goroutine to log delivery reports.
  - ○ **Produce Method**:
    - ◆ Publishes messages to the configured Kafka topic.
  - ○ **Close Method**:
    - ◆ Flushes and closes the Kafka producer.
- **Interaction**: Consumes messages from the bridge and sends them to the Kafka topic.

4. config.go: Configuration Management
- **Purpose**: Loads configuration from a YAML file and environment variables.
- **Key Components**:
  - ○ **Config struct**:
    - ◆ Contains nested configurations for Pub/Sub, Kafka, and application settings.
  - ○ **LoadConfig**:
    - ◆ Reads the YAML configuration file and overrides settings with environment variables (if present).
  - ○ **validateConfig**:
    - ◆ Ensures required fields are set (e.g., Kafka bootstrap servers, Pub/Sub subscription ID).
- **Interaction**: Provides configuration values to initialize the subscriber, producer, and bridge.

5. main.go: Application Entry Point
- **Purpose**: Initializes and orchestrates the components, handles lifecycle management.
- **Key Steps**:
  - ○ **Flag Parsing**: Reads the configuration file path from command-line

flags.
- ○ **Logger Initialization**: Sets up structured logging using zap.
- ○ **Configuration Loading**: Loads and validates configuration using config.go.
- ○ **Pub/Sub Subscriber**: Creates a Pub/Sub subscriber using the configuration.
- ○ **Kafka Producer**: Initializes a Kafka producer with appropriate settings.
- ○ **Bridge Service**: Sets up the bridge between the subscriber and producer.
- ○ **Graceful Shutdown**:
  - ◆ Listens for OS signals (e.g., SIGINT, SIGTERM) to stop services cleanly.
  - ◆ Ensures pending operations complete within a timeout.
- ○ **Metrics Reporting**:
  - ◆ Periodically logs metrics from the bridge (messages processed, failures, etc.).
- **Interaction**: Orchestrates all components, monitors for shutdown signals, and logs runtime metrics.

Overall Flow:
1. **Configuration**:
   - ○ Load from a YAML file (default: config/config.yaml) or environment variables.
2. **Subscriber**:
   - ○ Listens to messages from Google Pub/Sub.
3. **Bridge**:
   - ○ Processes and relays Pub/Sub messages to Kafka.
4. **Producer**:
   - ○ Sends messages to a Kafka topic.
5. **Metrics**:
   - ○ Periodically reports processing metrics and logs them.
6. **Graceful Shutdown**:
   - ○ Cleans up resources (e.g., close Kafka producer and Pub/Sub client).