

Exportadores

Independientemente de cómo se carga un exportador (ya sea a través de un JAR externo o no), todos los exportadores interactúan de la misma manera con el intermediario, que se define mediante la [interfaz del exportador](#).

Cargando

Una vez configurados, los exportadores se cargan como parte de la fase de inicio del intermediario, antes de realizar cualquier procesamiento.

Durante la fase de carga, se valida la configuración de cada exportador, de modo que el intermediario no se iniciará si:

- Un ID de exportador no es único
- Un exportador apunta a un JAR inexistente / no accesible
- Un exportador apunta a una clase inexistente / no instanciable
- Una instancia de exportador lanza una excepción en su `Exporter#configure` método.

El último punto es proporcionar a los exportadores individuales para realizar una validación ligera de su configuración (por ejemplo, fallar si faltan argumentos).

Una de las advertencias del último punto es que se crea una instancia de un exportador y se descarta inmediatamente; por lo tanto, los exportadores no deben realizar ningún trabajo computacionalmente pesado durante la creación de instancias / configuración.

Nota: Zeebe creará un cargador de clases aislado para cada JAR referenciado por las configuraciones del exportador, es decir, solo una vez por JAR; si se reutiliza el mismo JAR para definir diferentes exportadores, estos compartirán el mismo cargador de clases.

Esto tiene algunas buenas propiedades, principalmente que diferentes exportadores pueden depender de las mismas bibliotecas de terceros sin tener que preocuparse por las versiones o las colisiones de nombres de clase.

Además, los exportadores utilizan el cargador de clases del sistema para las clases del sistema o las clases empaquetadas como parte del JAR de Zeebe.

La configuración específica del exportador se maneja a través del `[exporters.args]` mapa anidado del exportador. Esto proporciona un simple `Map<String, Object>` que se pasa directamente en forma de un objeto de [configuración](#) cuando el intermediario llama al `Exporter#configure(Configuration)` método.

La configuración se produce en dos fases diferentes: durante la fase de inicio del intermediario, y una vez cada vez que se elige un líder para una partición.

Tratamiento

En cualquier punto dado, hay exactamente un nodo líder para una partición dada. Cada vez que un nodo se convierte en el líder de una partición, una de las cosas que hará es ejecutar una instancia de un [procesador de flujo de exportador](#) .

Este procesador de flujo creará exactamente una instancia de cada exportador configurado y reenviará cada registro escrito en el flujo a cada uno de ellos.

Nota: esto implica que habrá exactamente una instancia de cada exportador para cada partición: si tiene 4 particiones y al menos 4 subprocesos para el procesamiento, entonces hay potencialmente 4 instancias de su exportador que exporta simultáneamente.

Tenga en cuenta que Zeebe solo garantiza al menos una vez la semántica, es decir, un exportador verá un registro al menos una vez, y tal vez más. Los casos en que esto puede suceder incluyen:

- Durante el reprocesamiento después de la conmutación por error de la balsa (es decir, elección de nuevo líder)
- En caso de error si la posición aún no se ha actualizado

Para reducir la cantidad de registros duplicados que procesará un exportador, el procesador de flujo realizará un seguimiento de la posición del último registro exportado con éxito para cada exportador; la posición es suficiente ya que una secuencia es una secuencia ordenada de registros cuya posición está aumentando monotónicamente. El exportador mismo establece esta posición una vez que puede garantizar que un registro se ha actualizado correctamente.

Nota: aunque Zeebe intenta reducir la cantidad de registros duplicados que un exportador tiene que manejar, es probable que tenga que hacerlo; por lo tanto, es necesario que las operaciones de exportación sean idempotentes.

Esto se puede implementar en el propio exportador, pero si exporta a un sistema externo, se recomienda que realice una deduplicación allí para reducir la carga en el propio Zeebe. Consulte la documentación específica del exportador para saber cómo se debe lograr esto.

Manejo de errores

Si se produce un error durante la `Exporter#open(Context)` fase, el procesador de flujo fallará y se reiniciará, lo que podría solucionar el error; En el peor de los casos, esto significa que no se está ejecutando ningún exportador hasta que se detengan estos errores.

Si se produce un error durante la `Exporter#close` fase, se registrará, pero aún permitirá que otros exportadores finalicen su trabajo con gracia.

Si se produce un error durante el procesamiento, volveremos a intentar infinitamente el mismo registro hasta que no se produzca ningún error. En el peor de los casos, esto significa que un exportador fallido podría detener a todos los exportadores. Actualmente, se espera que las implementaciones de los exportadores implementen sus propias estrategias de reintento / manejo de errores, aunque esto puede cambiar en el futuro.

Impacto en el rendimiento

Zeebe naturalmente incurre en un impacto en el rendimiento de cada exportador cargado. Un exportador lento reducirá la velocidad de todos los demás exportadores para una partición determinada y, en el peor de los casos, podría bloquear completamente un hilo.

Por lo tanto, se recomienda mantener a los exportadores lo más simple posible y realizar cualquier enriquecimiento o transformación de datos a través del sistema externo.