Lifecycle of XDR record shipment, with metrics

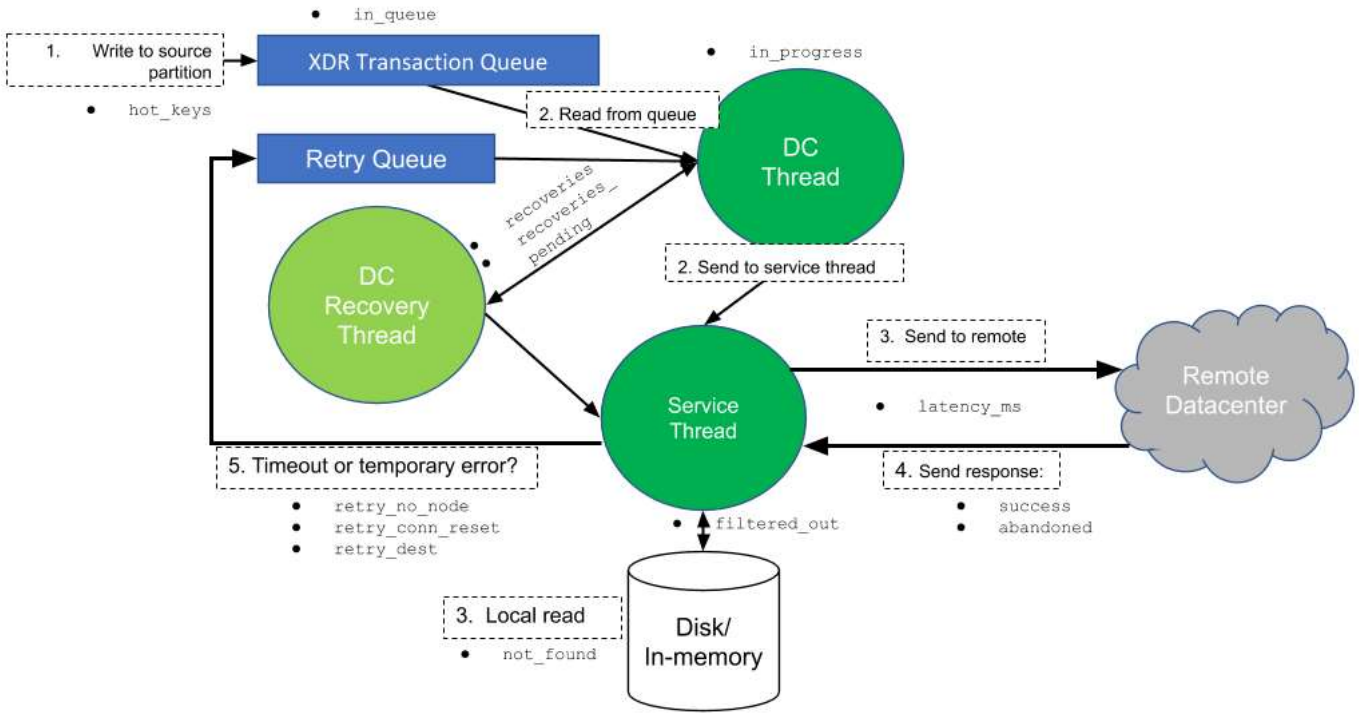
Last updated: 2020-06-18

A record progresses through a lifecycle of various states of XDR shipment. A record's states in its lifecycle are logged with various XDR metrics that indicate progress of the shipment.

You can start writing records as soon as you enable a namespace. The system starts logging in the in-memory transaction queue even if the datacenter is not connected at that point. After the connection is established, the system starts shipping.

The diagram below shows the lifecycle of a record from a single partition in a single namespace in a single datacenter. In actuality, these processes occur for all partitions and namespaces.

- Dashed boxes are processes discussed immediately below the diagram.
- Metric/state names are bulleted.



XDR record shipment lifecycle, with metrics

1. After a successful write in the source partition, the record is submitted to the per-partition in-memory transaction queue of each datacenter.
2. The DC thread picks the record from the transaction queue and forwards it to the service thread.
3. The service thread reads the record locally, prepares the record for shipment, and ships it to the remote destination.
4. The remote destination attempts to write the record and returns the completion state of the transaction to the source datacenter. The completion state can be success, temporary failure (like key busy/device overload), or permanent error (like record too big).
5. Based on the response, the service thread might put the record in the retry queue. If a retry is necessary, the shipment process of the record repeats from the retry queue.

Metrics of XDR processes

For precise definitions of these metrics, click the linked metric names.

Record state	Metric name	Description
Waiting for processing and processing started	`in_queue` (/docs/reference/metrics/index.html#in_queue) `in_progress` (/docs/reference/metrics/index.html#in_progress)	The record is in the XDR in-memory queue and is awaiting shipment.-XDR is actively shipping the record.
Retrying	`retry_conn_reset` (/docs/reference/metrics/index.html#retry_conn_reset) `retry_dest` (/docs/reference/metrics/index.html#retry_dest) `retry_no_node` (/docs/reference/metrics/index.html#retry_no_node)	This is a transient state. The shipment is being retried due to some error. Retries continue until one of the completed states is achieved.
Recovering	`recoveries` (/docs/reference/metrics/index.html#recoveries) `recoveries_pending` (/docs/reference/metrics/index.html#recoveries_pending)	This is a transient state that indicates internal-to-XDR "housekeeping". If XDR cannot find record keys in its in-memory transaction queue, it has to read the primary index to recover those keys for processing.

Record state	Metric name	Description
Completed	`success` (/docs/reference/metrics/index.html#success) `abandoned` (/docs/reference/metrics/index.html#abandoned) `not_found` (/docs/reference/metrics/index.html#not_found) `filtered_out` (/docs/reference/metrics/index.html#filtered_out)	Shipment is complete. The state of shipment is either success or one that indicates a non-success result.

(/) Contact Us (/forms/contact-us) Terms of Use (/forms/terms-conditions/) Privacy Policy (/forms/privacy-policy) © 2020 Aerospike, Inc. (/forms/terms-of-use/) (https://github.com/aerospike) (http://www.meetup.com/Scale-Warriors-of-Silicon-Valley) (https://www.yout