Amazon QLDB

July 2019

Before DLT, Blockchain, and QLDB…. Audit Tables!

	AuditLogDataId	TableName	ColumnName	RecordIdentifie...	RecordIdentifie...	ActionBy_UserId	ActionDateTime
	351	CountryRegion	CountryRegionC...	CountryRegionC...	1	59	2012-06-10 21:...
	352	CountryRegion	CountryRegionN...	CountryRegionC...	1	59	2012-06-10 21:...
	353	CountryRegion	ModifiedBy	CountryRegionC...	1	59	2012-06-10 21:...
	354	CountryRegion	ModifiedDate	CountryRegionC...	1	59	2012-06-10 21:...
	368						
	369						
	370						
	374						
	375						
	384						
	385						
▶	386						
	387						
	388						
	389						
	390						
	391						
	392						
*	NULL						

ActionType	Audit_Description	OldValue	OldValue_Decode	NewValue	NewValue_Dec...	AuditSubCateg...
1	CountryRegionC...	NULL	NULL	1	NULL	1
1	CountryRegionN...	NULL	NULL	INDIA	NULL	1
1	ModifiedBy Colu...	NULL	NULL	1	NULL	1
1	ModifiedDate Co...	NULL	NULL	Jun 10 2012 9:...	NULL	1
2	SalesPersonID C...	1	NULL	1	NULL	1
2	SalesPersonNam...	Jash	NULL	Jeba	NULL	1
2	TerritoryID Colu...	1	Anna University	2	NULL	1
2	ModifiedBy Colu...	1	NULL	2	NULL	1
2	ModifiedDate Co...	Jun 10 2012 9:...	NULL	Jun 10 2012 9:...	NULL	1
3	TerritoryID Colu...	1	NULL	NULL	NULL	1
3	TerritoryName C...	Anna University	NULL	NULL	NULL	1
3	CountryRegionC...	1	INDIA	NULL	NULL	1
3	ModifiedBy Colu...	1	NULL	NULL	NULL	1
3	ModifiedDate Co...	Jun 10 2012 9:...	NULL	NULL	NULL	1
3	CountryRegionC...	1	NULL	NULL	NULL	1
3	CountryRegionN...	INDIA	NULL	NULL	NULL	1
3	ModifiedBy Colu...	1	NULL	NULL	NULL	1
3	ModifiedDate Co...	Jun 10 2012 9:...	NULL	NULL	NULL	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

`Table Name`
`Column Name RecordIdentifierName` (Primary Column name)
`RecordIdentifierValue` (Primary Column Data)
`Actionby_UserID`
`ActionDateTime` (Time stamp)
`ActionType` (Insert / Update /Delete)
`Audit_Description`
`Old Value` (Data will be loaded for Update and Delete action)
`Old Value Decode` (Data from foreign key reference table)
`New Value` (Data will be loaded for Insert and Update action)
`New Value Decode` (Data from foreign key reference table)
`Category`

What is Amazon QLDB?

- Owned by a central trusted authority
- Tracks each and every data change
- Append-only journal history
- Cryptographically verifiable transaction log
- Fully managed - high performance and auto scales

Accessing QLDB

- AWS Command Line Interface
- API
- AWS Console

AWS Command Line Interface (CLI)

aws qldb

- create-ledger
- delete-ledger
- describe-journal-s3-export
- describe-ledger
- export-journal-to-s3
- get-block
- get-digest
- get-revision
- list-journal-s3-exports
- list-journal-s3-exports-for-ledger
- list-ledgers
- list-tags-for-resource
- tag-resource
- untag-resource

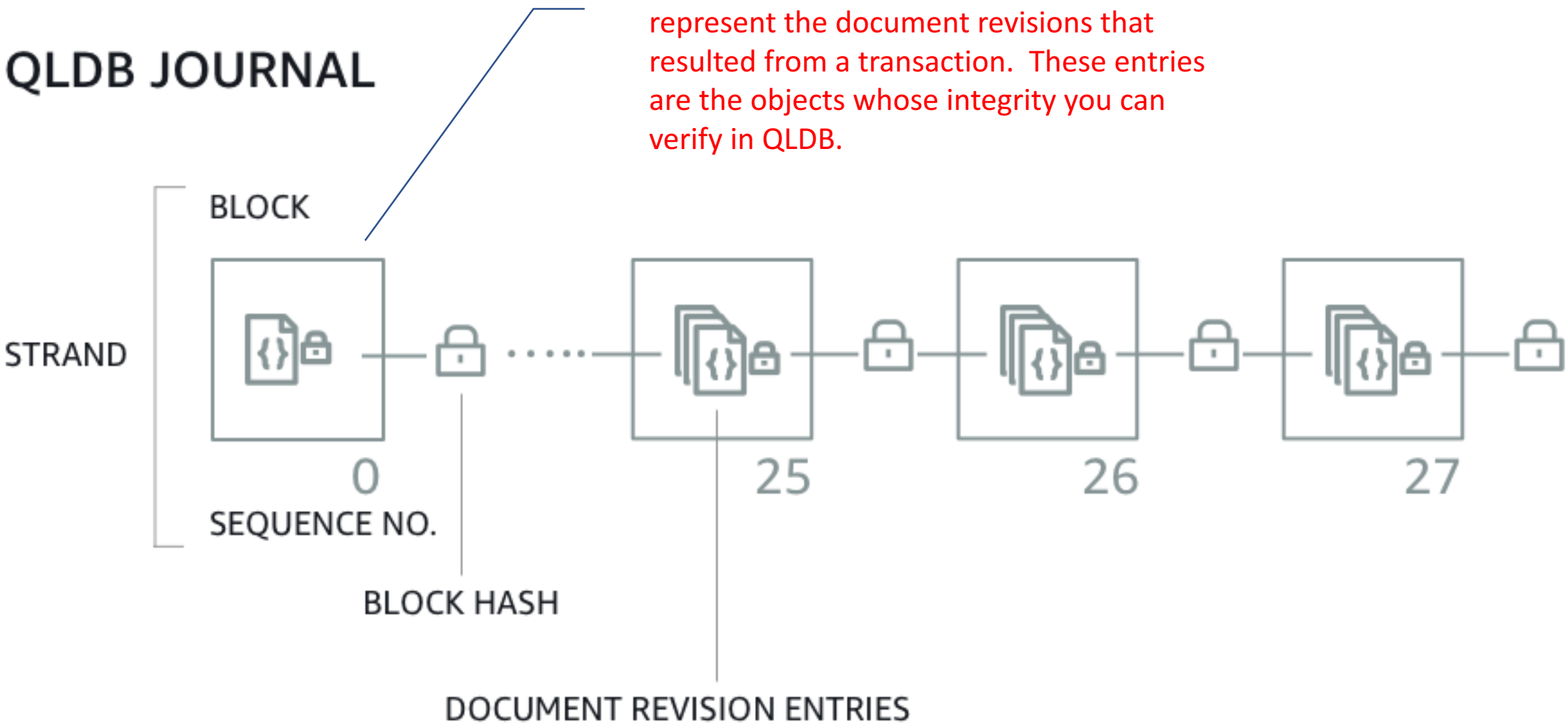
QLDB API

- SDK currently available for Java
- Consists of:
  - QLDB Java Client for calling the QLDB control plane
  - QLDB Driver – similar to JDBC driver. You can create a session with connectivity to a specific ledger in QLDB. This session enables you to execute QLDB SQL statements and retrieve the results of those statements. You can also take control over transactions to group multiple executions within a transaction.
  - Jackson Data Format for Amazon Ion - Superset of JSON used for storing both structured and unstructured data. Open source, richly typed, self-describing, hierarchal data serialization format

Data Verification in QLDB

- With Amazon QLDB, you can trust that the history of changes to your application data is accurate. QLDB uses an immutable transactional log (i.e. journal) for data storage. The journal tracks every change to your data and maintains a complete and verifiable history of changes over time.
- Sourcing data from the journal, QLDB uses a cryptographic hash function (SHA-256) with a Merkle tree–based model to generate a secure output file of your ledger's full hash chain. This output file is known as a ***digest*** and acts as a fingerprint of your data's entire change history as of a point in time. It enables you to look back and validate the integrity of your data revisions relative to that fingerprint.

Each ***block*** contains entry objects, which represent the document revisions that resulted from a transaction. These entries are the objects whose integrity you can verify in QLDB.



Data Integrity in QLDB

- Data integrity in QLDB means that your *ledger's journal is in fact immutable*. In other words, your data (specifically, each document revision) is in a state where the following are true:
  - It exists at the same location in your journal where it was first written.
  - It hasn't been altered in any way since it was written.
- What you are specifically proving is that the document revision was not altered between the time that you saved this digest and when you run the verification.