Datomic Cloud Documentation

Search

What is Datomic?

Local Dev and CI

Cloud Setup

Tutorial

Client API

Videos

- CLI tools

Schema

- Defining Attributes
- Schema Reference
- Changing Schema
- Data Modeling
- Schema Limits

Transactions

- Processing Transactions
- Transaction Data Reference
- Transaction Functions
- ACID
- Client Synchronization

Query and Pull

- Executing Queries
- Query Data Reference
- Pull
- Index-pull
- Raw Index Access

Time in Datomic

- Log API
- Time Filters

Ions

- Ions Tutorial
- Ions Reference
- Monitoring Ions

Analytics Support

- Configuration
- Connecting
- Metaschema
- SQL CLI
- Troubleshooting

Analytics Tools

- Metabase
- R
- Python
- Jupyter
- Superset

Supported Operations

Datomic provides the following set of operations:

- database operations
- ACID transactions
- dynamic schema
- Datalog query and pull
- time operations
- raw index access
- deploy entire applications to Datomic with ions
- SQL analytics support

Database Operations

The Datomic API allows you to create, list, and delete databases. For more information, see the Client API.

ACID Transactions

Databases are accumulate-only, and all new information is added by fully serialized ACID transactions. The `transact` operation takes as its arguments

- a connection to a database
- a collection that can include raw assertions, raw retractions, and transaction functions for conditional operations.

For example, to increase an item's price by 10%, you would pass a transaction function to `transact`, and two new datoms would be added to the database:

- a retraction of the previous price
- an assertion of the new price

For more information, see transactions.

Installing Schema

Schema attributes are ordinary entities in a database, and are added via ACID transactions just like any other data. Schema attributes must be defined in a transaction prior to any transaction that uses them.

For more information, see schema.

Query: Datalog and Pull

The `query` operation provides access to Datomic Datalog, a powerful deductive query system. Datomic Datalog includes unification, joins, predicates, functions, conjunction, disjunction, negation, and reusable rules.

The `pull` operation is a declarative way to make hierarchical (or nested) selections of information about entities. `pull` is integrated in `query` and also available as a standalone API.

For more information, see query.

Time

Datomic is accumulate-only, remembering the history of all information. The `db` operation returns the current value of a database. This value can then be filtered:

- the `as-of` operation filters a database back to a past point in time
- the `since` operation filters a database to include only datoms after a point in time
- the `history` operations returns an unfiltered view of all present and past information.

Datomic transactions are reified, that is, transactions are themselves entities in a database. You can use `query` and `pull` to retrieve information about transactions associated with an entity. Or you can go in the opposite direction, retrieving datoms from transactions directly via the `tx-range` operation.

For more information, see time.

Raw Indexes

Datomic provides direct iteration on indexes. Most applications will not use this, and you should prefer instead the higher-level `query` and `pull` operations.

The `datoms` operation can seek to a point in any index and iterate from there, and the `index-range` operation can provide a value range from the AVET index.

For more information, see indexes.

Client object that provides access to a database. Programs can use a connection to submit transactions and queries.A database is a set of datoms.An atomic fact in the database, associating an entity, attribute, value, and a tx. Opposite of a retraction.An atomic fact in the database, dissociating an entity from a particular value of an attribute. Opposite of an assertion.An atomic fact in a database, composed of entity/attribute/value/transaction/added. Pronounced like "datum", but pluralizes as datoms.An atomic fact in the database, dissociating an entity from a particular value of an attribute. Opposite of an assertion.An atomic fact in the database, associating an entity, attribute, value, and a tx. Opposite of a retraction.An atomic unit of work in a database. All Datomic writes are transactional, fully serialized, and ACID (Atomic, Consistent, Isolated, and Durable).Sorted collection of datoms. Indexes are named by the order in which datom components are used for sort, e.g. An index that sorts first by entity, then attribute, then value, then tx is called EAVT.