

This document primarily records and verifies the schema evolution capabilities of Flink SQL and its related ecosystem. The following tests will focus on three representative systems: **Paimon**, **Iceberg**, and **Fluss**.

Data Table Preparation

Paimon

```
CREATE CATALOG paimon_catalog WITH (  
  'type' = 'paimon',  
  'warehouse' = 'file:/tmp/paimon'  
);  
USE CATALOG paimon_catalog;  
CREATE TABLE paimon_table (  
  id BIGINT PRIMARY KEY NOT ENFORCED, -- Primary key for unique record identification  
  name STRING,                        -- Username  
  age INT,                            -- User age  
  email STRING                        -- User email  
);
```

Iceberg

```
CREATE CATALOG iceberg_catalog WITH (  
  'type'='iceberg',  
  'catalog-type'='hadoop',  
  'warehouse'='file:/tmp/iceberg',  
  'property-version'='1'  
);  
USE CATALOG iceberg_catalog;  
CREATE TABLE iceberg_table (  
  id BIGINT PRIMARY KEY NOT ENFORCED, -- Primary key for unique record identification  
  name STRING,                        -- Username  
  age INT,                            -- User age  
  email STRING                        -- User email  
);
```

Fluss

```
CREATE CATALOG my_fluss WITH (  
  'type' = 'fluss',  
  'bootstrap.servers' = 'coordinator-server:9123'  
);  
USE CATALOG my_fluss;  
CREATE TABLE fluss_table (  

```

```

id BIGINT PRIMARY KEY NOT ENFORCED, -- Primary key for unique record identification
name STRING,                        -- Username
age INT,                             -- User age
email STRING                         -- User email
);

```

Compatibility Validation

This section verifies whether schema modifications allow the system to read historical data. Compatibility is generally guaranteed by the underlying storage system.

Test Steps:

1. Start a real-time job before schema changes to continuously read current data.
2. Insert test data.
3. Modify the schema (e.g., add columns, drop columns, rename columns).
4. Insert additional test data.
5. Start a real-time job after schema changes to read data continuously.
6. Compare differences in the data read before and after schema changes.

Validation Process:

Step 1: Start a real-time job (before schema changes).

```
SELECT * FROM paimon_table;
```

Step 2: Insert test data.

```

-- Paimon
INSERT INTO paimon_table VALUES
(1, 'Alice', 30, 'alice@example.com'),
(2, 'Bob', 25, 'bob@example.com');
-- Iceberg
INSERT INTO iceberg_table VALUES
(1, 'Alice', 30, 'alice@example.com'),
(2, 'Bob', 25, 'bob@example.com');
-- Fluss
INSERT INTO fluss_table VALUES
(1, 'Alice', 30, 'alice@example.com'),
(2, 'Bob', 25, 'bob@example.com');

```

Step 3: Modify the schema (e.g., add, drop, reorder, or rename columns).

```
-- Add column: phone_number
```

```

ALTER TABLE paimon_table ADD phone_number STRING;
ALTER TABLE iceberg_table ADD phone_number STRING;
-- Drop column: email
ALTER TABLE paimon_table DROP email;
ALTER TABLE iceberg_table DROP email;
-- Reorder column: move 'name' after 'age'
ALTER TABLE paimon_table MODIFY name STRING AFTER `age`;
ALTER TABLE iceberg_table MODIFY name STRING AFTER `age`;
-- Rename column: name → user_name
ALTER TABLE paimon_table RENAME name TO user_name;
ALTER TABLE iceberg_table RENAME name TO user_name;

```

Updated schema for paimon_table

```

+-----+-----+-----+-----+-----+-----+
| name | type | null | key | extras | watermark |
+-----+-----+-----+-----+-----+-----+
| id | BIGINT | TRUE | | | |
| age | INT | TRUE | | | |
| user_name | STRING | TRUE | | | |
| phone_number | STRING | TRUE | | | |
+-----+-----+-----+-----+-----+-----+

```

4 rows in set

Fluss currently does not support schema evolution.

Step 4: Insert additional test data.

-- Schema-adapted data insertion

```

INSERT INTO paimon_table VALUES
(3, 28, 'Charlie', '123-456-7890');
INSERT INTO iceberg_table VALUES
(3, 28, 'Charlie', '123-456-7890');

```

Validation Results

Real-time job results before schema changes:

<u>id</u>	<u>name</u>	<u>age</u>	<u>email</u>
1	Alice	30	alice@example.com
2	Bob	25	bob@example.com
3	Charlie	28	<NULL>

Real-time job results after schema changes:

id	age	user_name	phone_number
1	30	Alice	<NULL>
2	25	Bob	<NULL>
3	28	Charlie	123-456-7890

Results Summary:

The above validation confirms two key scenarios:

1. A new schema can read old data.
2. An old schema can read new data.

Both scenarios correctly read the data, with compatibility ensured by the storage systems and connectors.

Special Scenarios

1. Dropping a Non-NULL Column

Both Paimon and Iceberg support this scenario:

```
CREATE TABLE paimon_table (
  id BIGINT,          -- Primary key
  name STRING,        -- Username
  age INT,            -- Age
  email STRING NOT NULL -- Non-NULL email
);
```

```
INSERT INTO paimon_table VALUES
(1, 'Alice', 30, 'alice@example.com'),
(2, 'Bob', 25, 'bob@example.com');
```

```
ALTER TABLE paimon_table DROP email;
```

Default behavior in the Flink Paimon connector assigns NULL to dropped fields. This causes issues when writing to sinks if a NOT NULL constraint is set:

Caused by: org.apache.flink.table.api.TableException: Column 'email' is NOT NULL, however, a null value is being written into it.

2. Modifying Primary Key Types

Modifying primary key types is unsupported by most storage systems but not restricted by Flink:

```
ALTER TABLE paimon_table MODIFY id STRING PRIMARY KEY NOT ENFORCED;
```

Error:

Change is not supported: ModifyUniqueConstraint

3. Adding Primary/Partition Keys

Unsupported by the underlying systems.

Example:

```
ALTER TABLE paimon_table MODIFY name STRING PRIMARY KEY NOT ENFORCED;
```

4. Data Type Modifications

- Paimon has fewer restrictions (e.g., converting STRING to INT is allowed).
- Iceberg enforces stricter type compatibility (e.g., allows INT → BIGINT, FLOAT → DOUBLE, but disallows nullable changes).

Summary

For data lake systems like Paimon and Iceberg:

- Simple operations such as adding, dropping, or renaming columns do not affect downstream job correctness.
- Key changes (primary/partition key modifications) are unsupported.
- **Fluss** does not yet support schema evolution.