

THE PROBLEM

Silent, compounding drift

A hotfix changes a nullable column to NOT NULL. Alembic has no history.

Staging has 3 extra columns from an abandoned branch; no migration covers it.

You deploy with Alembic's `create_all()`. New tables appear - altered columns are silently ignored.

Constraints went missing after a manual patch. Your models say otherwise.

HOW IT WORKS

Your SQLAlchemy models

Your PostgreSQL database

`compare()` # creates a `ConformPlan`

+

`apply_changes()` # executes DDL

=

Conformant Database

BASIC USAGE

A PostgreSQL Example

```
from dbconform import DbConform
from my_app.schemas import Product, Cart

conform = DbConform(
    credentials={
        "url": "postgresql+psycopg://
        user:pass@host/db"
    },
    target_schema="public"
)
result = conform.apply_changes([Product, Cart])
print(result)
```

Capability	Alchemy <code>create_all()</code>	Alembic	Atlas	dbconform
Create new tables	✓	✓	✓	✓
Alter existing tables	✗	✓	✓	✓
Can fix schema drift	✗	✗	✓	✓
Works without migration history	✓	✗	✗	✓
Pure Python	✓	✓	✗	✓
SQLite rebuild capabilities	✗	✗	✗	✓
Safe defaults	✓	⚠	⚠	✓
In-process, programmatic	✓	✓	✗	✓

SAFE BY DEFAULT

Changes are additive unless you opt-in.
Destructive operations are blocked unless explicitly unlocked.

<code>allow_drop_extra_tables</code>	opt-in ✗
<code>allow_drop_extra_columns</code>	opt-in ✗
<code>allow_shrink_column</code>	opt-in ✗
<code>allow_drop_extra_constraints</code>	on ✓
<code>allow_sqlite_table_rebuild</code>	on ✓

WHEN TO REACH FOR THIS

- Inherited a DB with no usable migration history
- Databases in dev and prod structurally diverged
- Direct DB patches that never made it into code
- Schema conformance enforced at application startup
- No appetite for a migration pipeline at all

PostgreSQL

SQLite

Sync/Async Python

Bring an engine, or let dbconform create one

What I need from you

- Feedback
- Battle-testing in real-world schemas
- PostgreSQL and SQLite edge cases

Brian Pond

brian@datahenge.com

pypi.org/dbconform

github.com/brian-pond/dbconform



Scan to open repo