



ORM Tips

- You can use your models with any relational database that has DBAPI compatible driver for Python (pretty much everything)
- Let's you avoid using proprietary or non-standard SQL Dialects
- You can still use proprietary function of specific DBs when you need them
- Even though it adds another layer to your application, ORMs do provide opportunities for memoization and other performance tuning techniques (would be a whole other presentation)
- Not working on a new DB? [sqlacodegen](#) can be used to generate model classes from existing DB schemas.



Continuum

- Capturing and storing changes to the data itself in your DB is very useful for audit/compliance, debugging and general peace of mind
- It may not be as important for reporting DBs, where data get refreshed often and in bulk
- It does add some performance overhead
- [SQLAlchemy-continuum](#) is an easy add on to [SQLAlchemy](#) and [Alembic](#) to create a complete solution for a well managed database