

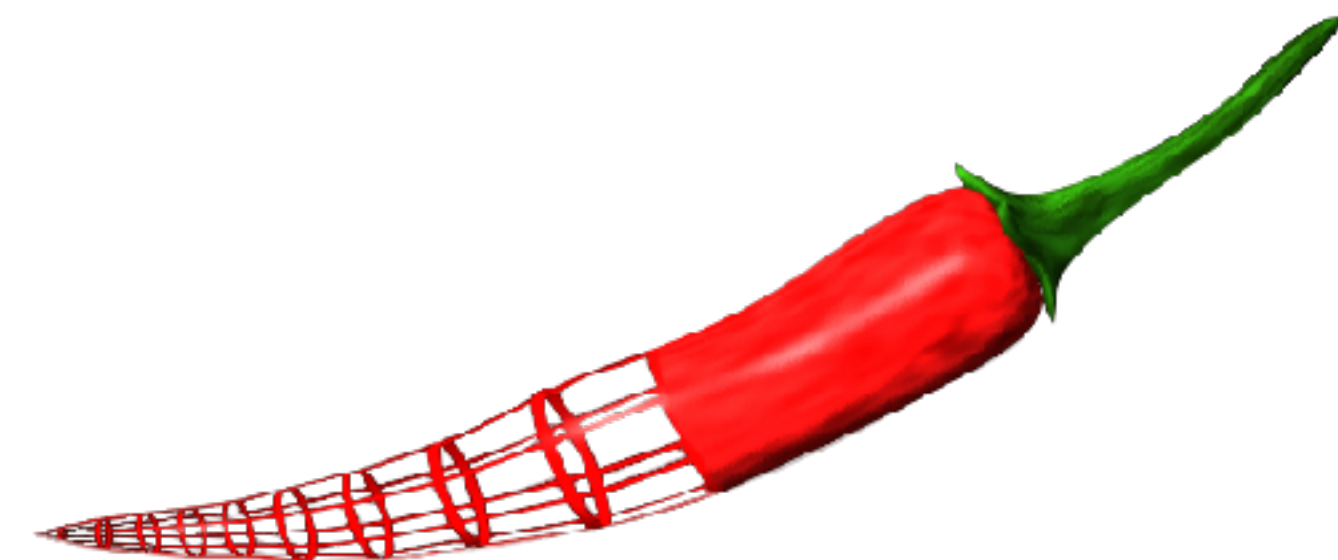


# Apache Cayenne

a Java ORM Alternative

by Andrus Adamchik

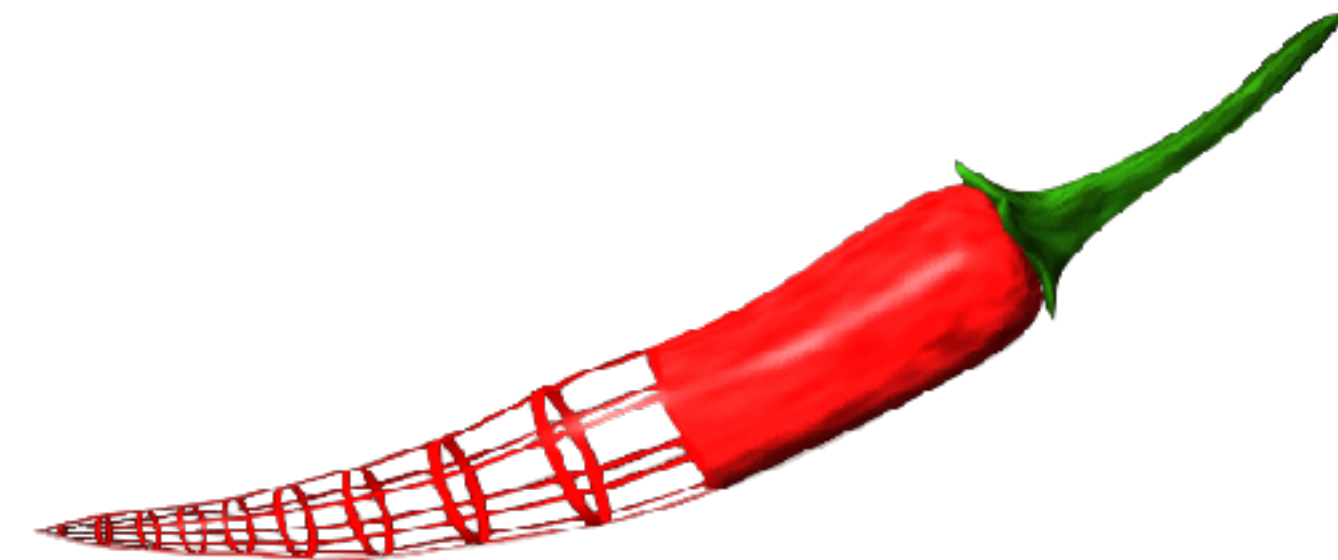
 @andrus\_a





# What is Apache Cayenne

- A mature user-friendly Java ORM with modeling tools.
- An alternative object persistence philosophy.
- Community-driven. Dozens of contributors, 8 PMC members.

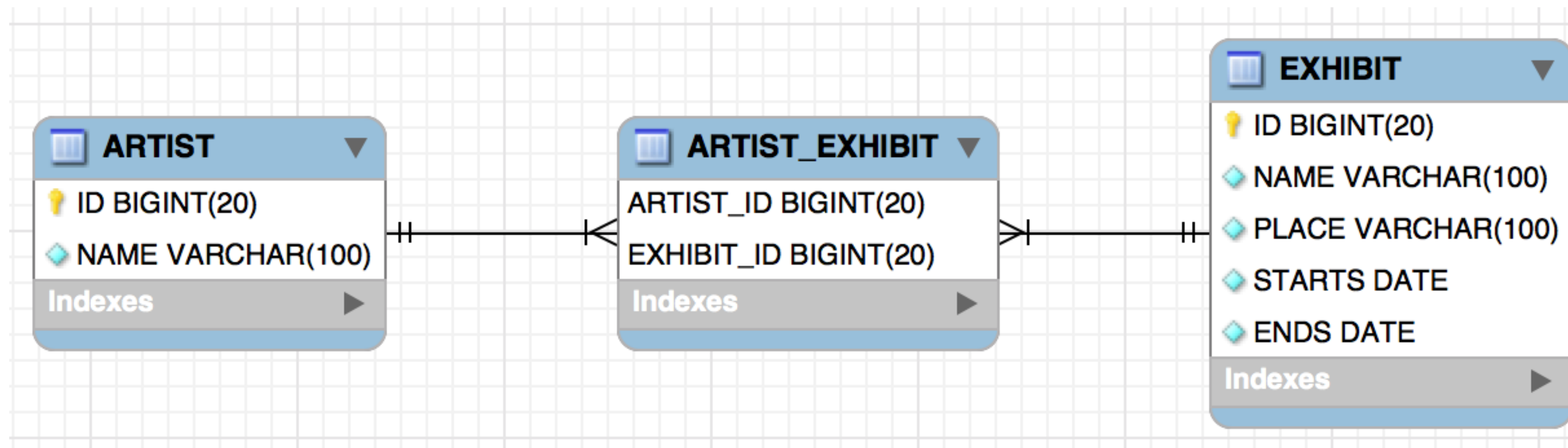


# Apache: Community over Code

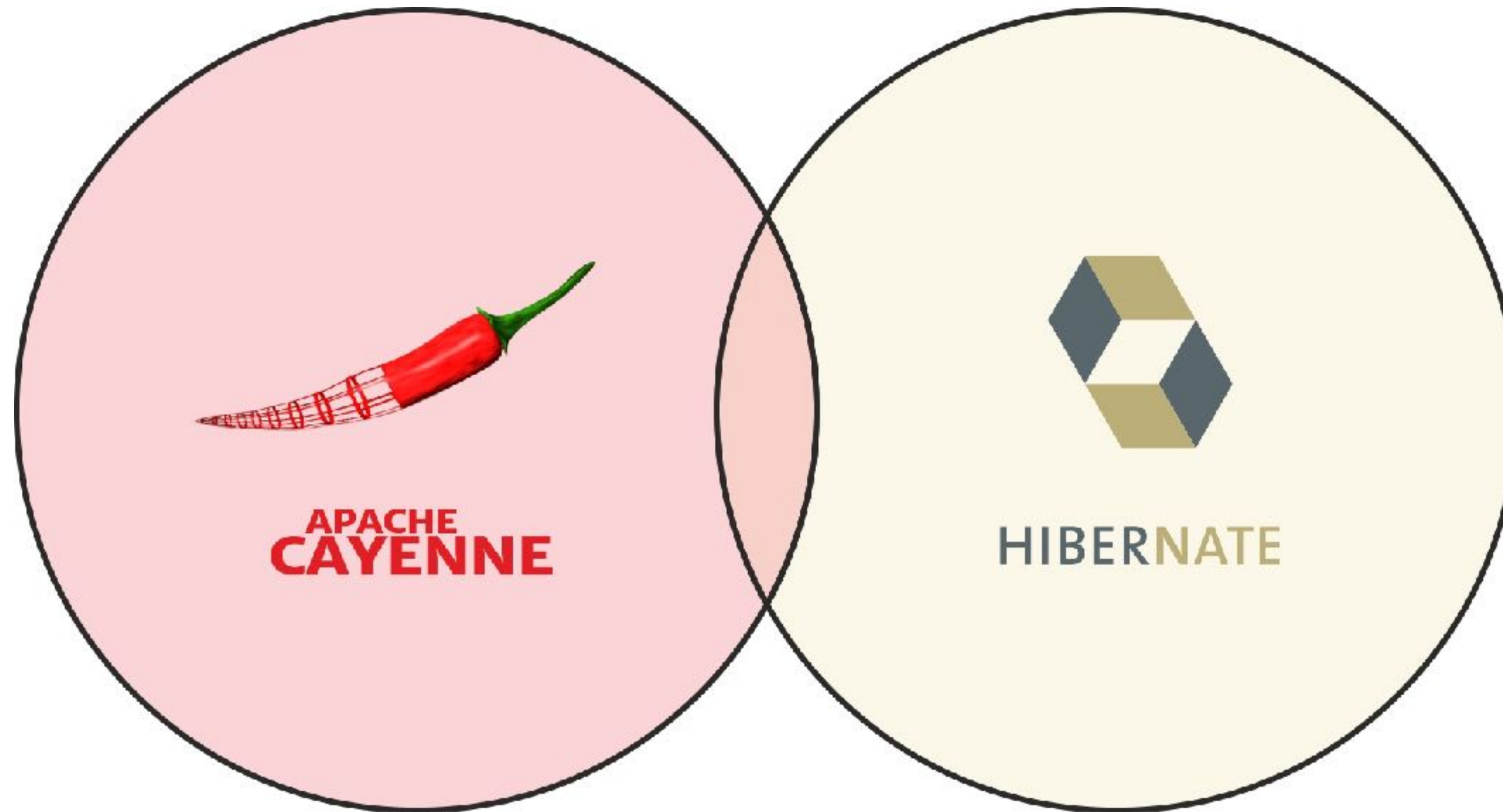
- Apache Foundation focuses on project communities.
- Meritocracy - anyone can earn a vote by contributing.
- Clear license policies and traceable code origins.
- Successful model: hosts or incubates more than 350 projects.



# Demo - Art Exhibits



# Cayenne vs. JPA / Hibernate



# Generated SQL

- Hibernate:

```
select artist0_.id as id1_0_, artist0_.name as  
name2_0_ from ARTIST artist0_
```

- Cayenne:

```
SELECT t0.name, t0.id FROM ARTIST t0
```

# Transactions

- JPA/Hibernate were born in the EJB world. Both are transaction-centric.
- Cayenne transactions are implicit.
- Cayenne is not connected to DB between JDBC operations.
  - ➔ Better connection pool reuse and no user TX code.



# ObjectContext

- Vaguely analogous to `SessionFactory / EntityManager`.
- Better analogy might be Git - `ObjectContext` is an object holder disconnected from DB.
- Not holding any open resources (connections, etc.), doesn't need to be closed.
- Allows to expand object graph lazily and transparently, as relationships are traversed in the code.

# ObjectContext Avoids This:

```
org.hibernate.LazyInitializationException:  
failed to lazily initialize a collection of role: h.Artist.paintings,  
could not initialize proxy – no Session  
    at o.h.c.i.AbstractPersistentCollection  
        .throwLazyInitializationException  
    (AbstractPersistentCollection.java:582)
```

# Persistent Object

- Mapping is separated from objects. No pollution with ORM annotations.
- Framework-provided superclass - **CayenneDataObject**.
- No class enhancement or dynamic proxies.
- Generic property access API.
- Generic mapping - can be created in runtime.

# How Do You Protect DB Data?

- Passwords.
- Credit card numbers.
- Personal user data.
- Sensitive business data.

# 1-Way Hash (for passwords)

```
byte[] passwordHash(String password, byte[] salt) throws Exception {  
  
    int iterations = 1000;  
    int keyLength = 64 * 8;  
  
    PBEKeySpec spec =  
        new PBEKeySpec(password.toCharArray(), salt, iterations, keyLength);  
  
    return SecretKeyFactory.getInstance("PBKDF2WithHmacSHA512")  
        .generateSecret(spec)  
        .getEncoded();  
}
```

# 2-Way Encryption (for everything else)

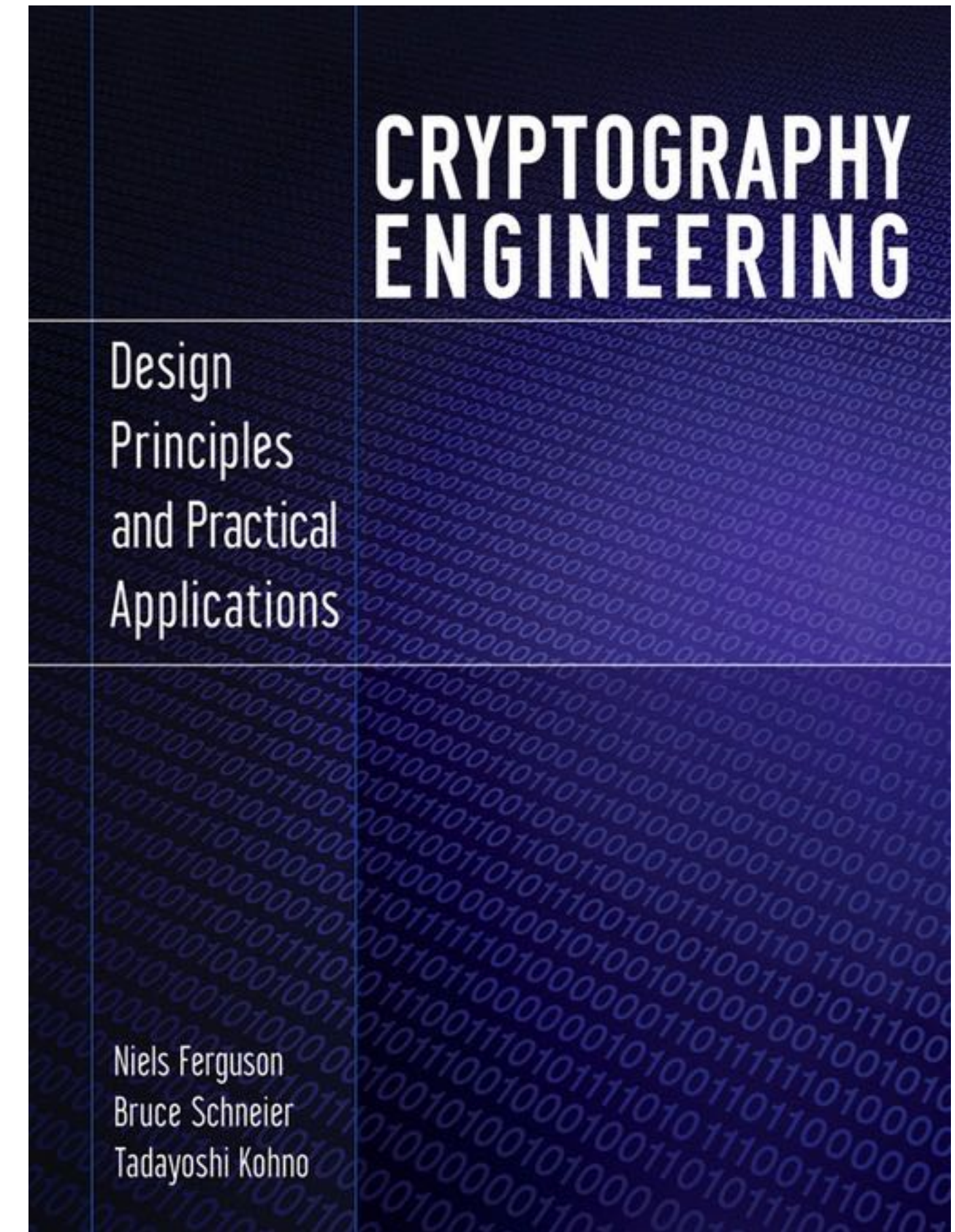
- How do we implement reliable cryptography?
- How do we revoke a key?
- Can we have multiple keys?

# Demo - Encryption



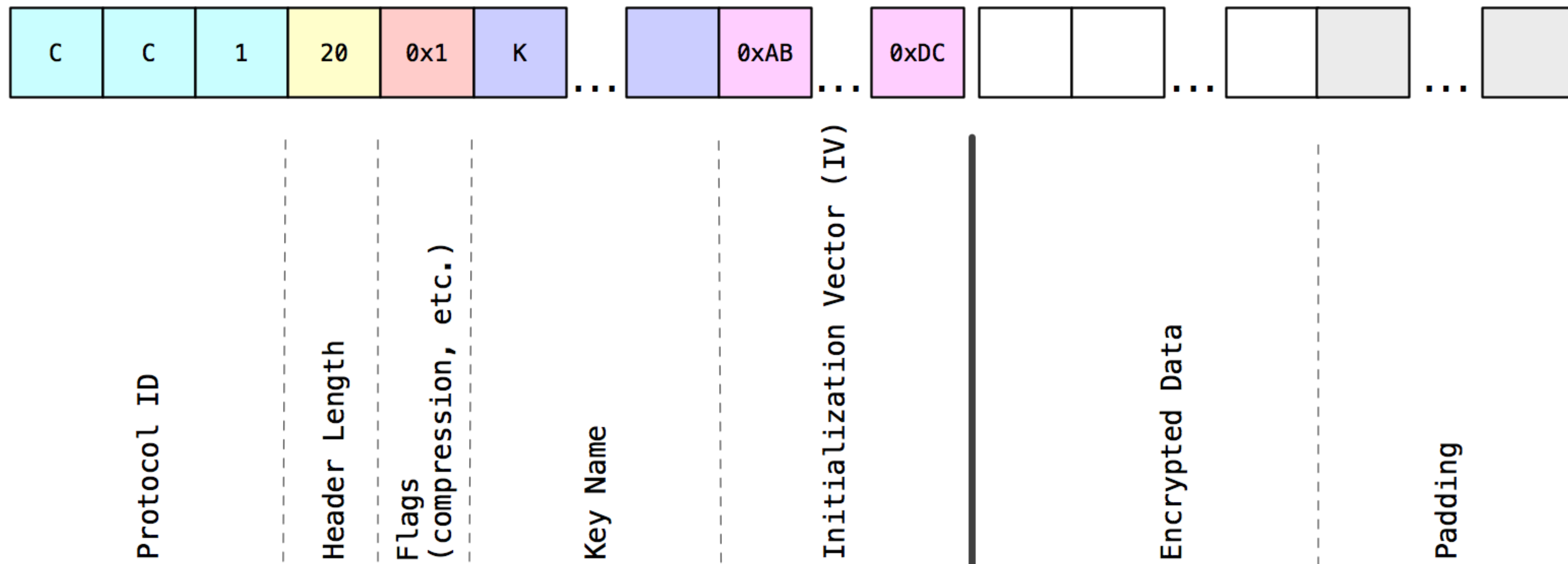
# Encryption Details

- AES block cipher with 128 or 256 bit key
- Block Size: 16 bytes
- Cipher Mode: CBC (Cipher Block Chaining)
- Padding: PKCS#5





# How Does It Work?



Other Good Things

# Dependency Injection (DI)

```
Module myExtensions = (binder) ->
    binder.bind(QueryCache.class).to(EhCacheQueryCache.class);
```

```
ServerRuntime runtime = ServerRuntimeBuilder.builder()
    .addModule(myExtensions)
    .addConfig("cayenne-project.xml")
    .build();
```

# Caching

```
List<Artist> artists = ObjectSelect
    .query(Artist.class)
    .localCache("artists") // associates query with cache policy
    .select(context);
```

# Handling Large Result Sets

```
// iterate over a data stream of unknown size  
ObjectSelect.query(Artist.class).iterate(context,  
    artist -> System.out.println(artist.getName()));
```

# Batch Iterator

```
// same, but iterate in batches
try (ResultBatchIterator<Artist> it = ObjectSelect.query(Artist.class).
    batchIterator(context, 100)) {

    it.forEach(batch -> {

        ObjectContext editContext = runtime.newContext();


        // do something with each object in the batch

        editContext.commitChanges();
    });
}
```

# Why Cayenne?

- Mature, performant, user-friendly.
- The only ORM alternative to JPA/Hibernate.
- Implicit transactions and smooth object graph navigation.
- Community-driven. You can have a voice!

# Questions?

-  @ApacheCayenne
- <http://cayenne.apache.org/>
- <https://github.com/apache/cayenne>

