Type: Build	Title: Deploying Remotely	Status: X

Context: Pulling from a database using Prisma CLI on another repo reads from the database, not migrations. This implies that foreign relational keys will occupy the name of its Model, e.g., `post.sections` becomes `post.Section`.

Core Decision: Bypass casing flaw on remote deployment.

Prospects	Decision Parameters	0TD Concerns

1. Control behaviors on build time
2. Produce a script modifying the schema on deploy
	§ **Using Prisma**
	← Create an sql file which alters foreign keys
	  ○— Before remote generation, use `prisma execute --file <file> --schema <schema>`
	← Map fields using `@map`, `@@map`
	§ **Using Java or NodeJs**
	← Create a class or function which is called on build time
	  ○— Use prisma format after the changes
	  ○— Create an npm script that compiles, runs and formats
		♭ The unexpected casing behavior demands revisits to the script if the schema were to change
		♭ If other alternatives are found, the build command pipeline must be reverted and the script deleted

Decision Process	Results and Conflicts

1. Mapped fields and models
2. Created `rectify-schema.sql`
  • Altered "constraint" and foreign ids
  • Java utility class RectifySchema was made
	1. Relational mapping of models requires back-end gynamstics to communicate with real db; out-of-sync
	  ○— Prisma execute is ignored by shadow database and updates real db
	  ○— Relational mapping of relational values is prohibited
	  ‡ Aborted
	2. Java and JavaScript is capable of altering the schema pre-generation
	  ○— Used Java 11 methods; Netlify is at 8 by default
	  ○— Refactored to Java 8
	  ○— Made npm script `rectify` to make schema changes post-pull preceding a format
	  ‡ Successful