

Prisma Migration Notes - Phase 7 Quality Framework

Migration Strategy: Database-First Reconciliation

Date: August 29, 2025

Strategy: DB-First (Non-destructive)

Status:  Complete

Why Database-First?

1. **Production Safety:** Avoids data loss by treating the live database as the source of truth
2. **Non-destructive:** No `prisma migrate reset` or destructive operations
3. **Incremental:** Allows gradual addition of Phase 7 features alongside existing data
4. **Lower Risk:** Maintains compatibility with existing production deployments

Migration History

Baseline Migration: `0_init_from_db_20250829_121502`

Purpose: Establish clean migration baseline from current production state

Includes:

- All existing models (User, Site, Category, Week, Topic, Connection, JobRun)
- NextAuth models (Account, Session, VerificationToken)
- **Phase 7 models** (already in production DB):
 - `SiteStrategy` - Per-site quality customization
 - `GlobalRulebook` - Global quality standards
 - `RulebookVersion` - Version history for rulebooks

Strategy: Created baseline migration SQL and marked as applied using:

```
npx prisma migrate resolve --applied "0_init_from_db_20250829_121502"
```

Schema Updates Applied

Enhanced Models:

```

model Site {
  strategy SiteStrategy? // Added Phase 7 relation
}

model Topic {
  flags Json? // Added for quality flags
}

model JobRun {
  metadata Json? // Added for observability data
}

model Connection {
  kind String @unique // Added unique constraint
}

```

New Phase 7 Models:

```

model SiteStrategy {
  id String @id @default(cuid())
  siteId String @unique
  strategy Json // Site-specific quality rules
  createdAt DateTime @default(now())
  updatedAt DateTime @updatedAt
  site Site @relation(fields: [siteId], references: [id], onDelete: Cascade)
  @@map("site_strategies")
}

model GlobalRulebook {
  id String @id @default(cuid())
  version Int // Version number
  rules Json // Quality rules JSON
  sources Json // Source URLs for rules
  updatedBy String? // Who updated the rulebook
  createdAt DateTime @default(now())
  updatedAt DateTime @updatedAt
  @@map("global_rulebooks")
}

model RulebookVersion {
  id String @id @default(cuid())
  version Int // Version number
  rules Json // Historical rules snapshot
  sources Json // Historical sources
  notes String? // Update notes
  createdAt DateTime @default(now())
  @@map("rulebook_versions")
}

```

Current State

Migration Status:  Clean

```

npx prisma migrate status
# Output: 1 migration found in prisma/migrations
# Database schema is up to date!

```

Database: Production Neon PostgreSQL

Tables: 13 models total (10 existing + 3 Phase 7)







Constraints: All foreign keys, unique indexes, and relations working

Future Migrations

Process:

1. Edit `prisma/schema.prisma` with changes
2. Run `npx prisma migrate dev --name descriptive_name`
3. Commit the new migration file
4. Deploy with `npx prisma migrate deploy`

Guidelines:

-  **DO:** Add new fields as optional (`field_name Type?`)
-  **DO:** Add new tables and relations
-  **DO:** Create additive indexes
-  **AVOID:** Dropping columns with data
-  **AVOID:** Changing column types that lose data
-  **AVOID:** Renaming tables or columns in production

Rollback Strategy

Current Baseline: Can safely revert to pre-Phase 7 state

Method:

1. Remove Phase 7 models from schema
2. Create migration to drop Phase 7 tables
3. Deploy migration

Data Preservation: Phase 7 rollback preserves all existing content data

Troubleshooting

Common Issues:

1. **Migration Drift:** Run `npx prisma db pull` then create new migration
2. **Schema Sync:** Use `npx prisma generate` after schema changes
3. **Connection Issues:** Verify `DATABASE_URL` environment variable

Verification Commands:

```
# Check migration status
npx prisma migrate status

# Verify schema sync
npx prisma validate

# Test database connection
npx prisma db pull --preview-feature
```

References

- **Prisma Docs:** <https://www.prisma.io/docs/concepts/components/prisma-migrate>
- **Production Migrations:** <https://www.prisma.io/docs/guides/migrate/production-troubleshooting>
- **Phase 7 Spec:** `docs/PHASE7_QUALITY_FRAMEWORK.md`