

Docs » Symfony Bundles » Platform Bundle » Entities

Entities

MappedSuperclass pattern, entity map, fields, and extending.

Anonymous · 09/09/2026

Entity model

Pattern: MappedSuperclass + ResolveTargetEntityListener

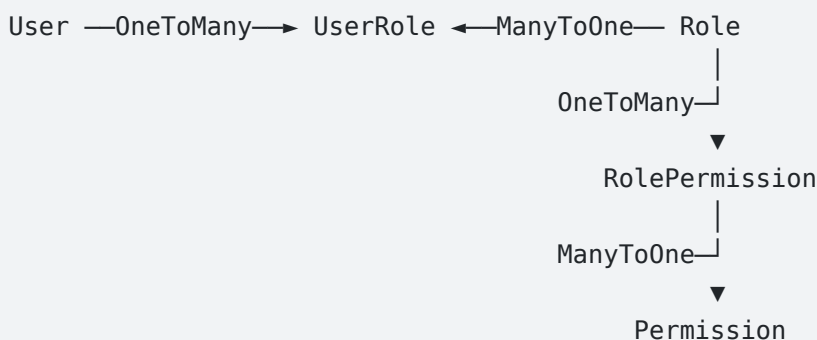
Every entity is split in two:

- **Base*** (bundle) — abstract #[ORM\MappedSuperclass] with all fields, relations, and logic. Lives in `Derafu\PlatformBundle\{Module}\Entity\`.
- **Concrete** (app) — extends the Base, adds #[ORM\Entity] + #[ORM\Table]. Lives in `App\Entity\{Group}\`.

ORM relations in the bundle target **interfaces** (not concrete classes). Doctrine resolves them at runtime via `resolve_target_entities`, wired automatically by the bundle's DI extension.

Identity entities

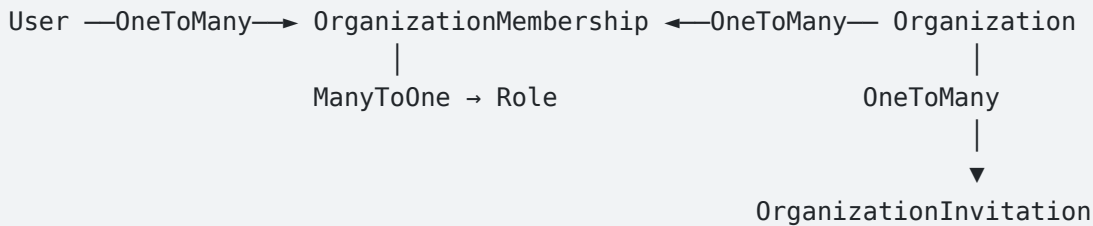
Core identity



- **User** — the authenticated subject. Fields: `id`, `uuid` (v7), `name`, `email` (unique, login identifier), `password` (hashed), `active`, `emailVerifiedAt`, `createdAt`, `config` (JSON). Implements `Symfony's UserInterface` and `PasswordAuthenticatedUserInterface`.
- **Role** — reusable role identified by a unique code (e.g. `org.admin`). No scope column — the scope is defined by the relation that uses the role.
- **Permission** — atomic capability identified by a unique code (e.g. `invoice.create`).
- **UserRole** — global system-level role assignment (super-admin, auditor).

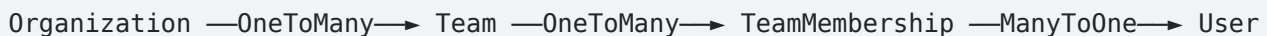
- **RolePermission** — which permissions a role grants.

Organizations



- **Organization** — a tenant. Fields: `id`, `uuid`, `name`, `active`, `createdAt`, `config`. No `owner` column — the owner is the membership with `role.code = 'org.owner'`.
- **OrganizationMembership** — user ↔ organization ↔ role with unique constraint (`user`, `organization`). Field `joinedAt`.
- **OrganizationInvitation** — pending invite. Fields: `email`, `tokenHash` (SHA-256), `expiresAt`, `invitedBy`, `acceptedAt`, `acceptedBy`, `revokedAt`, `createdAt`.

Teams



- **Team** — a named group of org members with a URL-friendly `slug`.
- **TeamMembership** — user ↔ team pivot, validates org membership.

Verification tokens

- **VerificationToken** — one-use token for user lifecycle operations. Fields: `tokenHash` (SHA-256), `user`, `type` (enum), `payload` (JSON), `expiresAt`, `usedAt`, `createdAt`. Types: `email_verify`, `password_reset`, `email_change`, `magic_link`.

API and JWT tokens

- **ApiKey** — long-lived programmatic access token. Format: `dik_` + 40 hex chars. Fields: `name`, `tokenHash`, `displayPrefix`, `scopes` (JSON), `lastUsedAt`, `expiresAt`, `createdAt`.
- **JwtToken** — persisted JWT for audit and revocation. Fields: `jti` (JWT ID), `user`, `issuedAt`, `expiresAt`, `revokedAt`.

Security records

- **LoginRecord** — one row per successful login: `IP`, `User-Agent`, `authenticator`, `loginAt`.
- **ImpersonationRecord** — one row per `switch_user` session: `admin`, `target`, `IP`, `startedAt`, `endedAt`.

Localization

- **Locale** — supported locale code and display name.
- **Timezone** — supported timezone identifier and display name.

Apps entities

```
User / Organization —OneToMany→ UserAppInstallation / OrganizationAppInstallation
└─ config (JSON, encrypted sensitive keys)
```

- **BaseUserAppInstallation** — app installed for a specific user. Fields: `user`, `appCode`, `config` (JSON), `installedAt`, `active`.
- **BaseOrganizationAppInstallation** — app installed for an organization. Fields: `organization`, `appCode`, `config` (JSON), `installedAt`, `active`.
- **BaseResourceAppInstallation** — app installed for a specific app resource. Extend per resource type with a typed FK to the resource.

All live under `Derafu\PlatformBundle\Apps\Entity\`.

Notifications entities

```
User —OneToMany→ UserWebhookEndpoint —OneToMany→ UserWebhookDelivery
User —OneToMany→ UserNotificationPreference
User —OneToMany→ UserDigestPreference
User —OneToMany→ InAppNotification

Organization —OneToMany→ OrganizationWebhookEndpoint
└─OneToMany→ OrganizationWebhookDelivery
```

- **InAppNotification** — in-app notification persisted for the user. Fields: `user`, `eventCode`, `data` (JSON), `readAt`, `createdAt`.
- **UserWebhookEndpoint / OrganizationWebhookEndpoint** — registered webhook URL. Fields: `url`, `secret` (for HMAC signing), `active`, `eventCodes` (JSON array of subscribed events), `createdAt`.
- **UserWebhookDelivery / OrganizationWebhookDelivery** — delivery attempt log. Fields: `endpoint`, `eventCode`, `payload` (JSON), `status`, `statusCode`, `responseBody`, `attemptedAt`.
- **UserNotificationPreference** — per-user, per-event channel preferences.
- **UserDigestPreference** — digest schedule preferences.

All live under `Derafu\PlatformBundle\Notifications\Entity\`.

Extending entities

The concrete entity can add fields without breaking bundle services:

```
#[ORM\Entity]
#[ORM\Table(name: 'auth_users')]
class User extends BaseUser
{
    #[ORM\Column(type: Types::STRING, length: 20, nullable: true)]
    private ?string $phone = null;

    public function getPhone(): ?string
    {
        return $this->phone;
    }

    public function setPhone(?string $phone): static
    {
        $this->phone = $phone;
        return $this;
    }
}
```

Bundle services, voters, and events only see the entity via its interface (`UserInterface`) — extra fields are transparent to the bundle.

Constructors

- `BaseUser` and `BaseOrganization` auto-generate UUID v7 and freeze `createdAt` at construction time.
- `BaseUserRole` and `BaseRolePermission` require both FK sides as constructor arguments.
- `BaseOrganizationMembership` requires (`user`, `organization`, `role`).
- `BaseOrganizationInvitation` requires (`organization`, `email`, `role`, `tokenHash`, `expiresAt`).
- `BaseVerificationToken` requires (`user`, `type`, `expiresAt`, `tokenHash`).

Last updated on 09/09/2026