



WERKGROEP GAP		249
16 maart 2010	Not.	6

Aanwezig: Bart, Johan, Peter, Broes, Tommy
Verontschuldigd: --

Als verslag is het voorbereidingsdocument bijgewerkt. Toevoegingen en aanpassingen in het blauw.

1. Beperkingen van het gebruikte ORM-systeem

Een eerste probleem

Bekijk dit fictieve stukje code, dat uit de businesslaag zou kunnen komen:

```
var ids = new int[] { gpID1, gpID2 };
var gelPersonen = gpDao.Ophalen(ids, gp=>gp.Categorie);
var p1 = gelPersonen.First();
if (p1 != null && p1.Categorie.First() != null)
{
    // verwijder eerste koppeling persoon-categorie
    p1.Categorie.First().TeVerwijderen = true;
    gpDao.Bewaren(p, gp=>gp.Categorie);
}
```

Stel dat je een categorie 'Blabla' hebt, met daaraan gekoppeld 2 personen: Jos en Eugène. Noch Jos noch Eugène zijn gekoppeld aan een andere categorie.

Het aanroepen van bovenstaand stukje code zal dan zowel Jos als Eugène loskoppelen van de categorie 'Blabla', en dat is mogelijk niet wat je verwacht.

Hoe komt dat nu?

De aanroep van 'gpDao.Ophalen' haalt de 2 gelieerde personen (laat ons zeggen p1 en p2) op. Deze 2 personen zijn gekoppeld aan 1 en hetzelfde categorie-object (c).

Om de link te verwijderen tussen p1 en c, moeten we in onze manier van werken c.TeVerwijderen true maken, en daarna gpDao.Bewaren aanroepen. AttachObjectGraph zal omwille van een gezette c.TeVerwijderen, de link tussen p1 en c laten vallen. Maar omdat ook p2 aan c gekoppeld is, verdwijnt om diezelfde reden ook de link p2-c.

Een tweede probleem

```
var gPers = gpDao.Ophalen(gpID, gp=>gp.Categorie);
if (gPers.Categorie.First() != null)
{
    // probeer persoon los te koppelen van categorie
    gPers.Categorie.First().TeVerwijderen = true;

    // bewaren...
    // GOED:
    gpDao.Bewaren(gPers, gp=>gp.Categorie);

    // FOUT (want verwijdert categorie):
    // catDao.Bewaren(cat, cat=>GelieerdePersoon)
```

```
}
```

De twee 'Bewaren'-statements vertrekken van dezelfde data. In het eerste geval bewaar je de persoon, met de gekoppelde categorieën, in het tweede geval de categorie, met de gekoppelde personen.

Toch is het resultaat verschillend. In het eerste geval wordt de categorie van de persoon losgekoppeld, in het tweede geval verdwijnt de categorie.

Achterliggende oorzaak

Zowel om een entiteit te verwijderen, als om een link naar een entiteit te verwijderen, gebruiken we de property `IbasisEntiteit.TeVerwijderen`. Alnaargelang de situatie wordt `TeVerwijderen` anders geïnterpreteerd:

- Als het 'root-object' van een te bewaren entity-graph 'TeVerwijderen' is, dan verdwijnt het root-object.
- Als een object op een dieper niveau in een entity-graph 'TeVerwijderen' is...
 - en de entiteit is gedecoreerd met een `AssociationEndBehaviorAttribute` dat verwijst naar de navigation property, dan verdwijnt het `TeVerwijderen` object
 - in het andere geval verdwijnt gewoon de link

Fouten vermijden

Om fouten te vermijden, moet er heel voorzichtig omgesprongen worden met 'TeVerwijderen'. Ik stel het volgende voor:

1. 'TeVerwijderen' moet zo weinig mogelijk gebruikt worden. We hebben wel geluk: we zullen in onze applicatie in het algemeen meer moeten toevoegen en wijzigen dan verwijderen.
2. De 'TeVerwijderen' property mag enkel in de businesslaag gebruikt worden.
3. Bijgevolg moet iedere business method die 'TeVerwijderen' gebruikt persisteren.
4. Na iets gepersisteerd te hebben, moet verder gewerkt worden met wat het DAO teruggeeft, en niet met het origineel.

Punt 4 vraagt wat meer verduidelijking. Bekijk dit voorbeeld:

```
public Categorie LeegMaken(Categorie c)
{
    foreach (var gp in c.GelieerdePersoon)
    {
        gp.TeVerwijderen = true;
    }
    // GOED:
    // persisteren, en bijgewerkte categorie opleveren
    return catDao.Bewaren(c, cat=>cat.GelieerdePersoon);

    // FOUT zou zijn:
    // catDao.Bewaren(c, cat=>cat.GelieerdePersoon);
    // return c;
    // want c.GelieerdePersoon blijft niet-leeg met members
    // die TeVerwijderen zijn.
}
```

`catDao.Bewaren` zal `c` attachen aan de object graph, dan de wijzigingen committen, en de opnieuw gedetachte bijgewerkte `c` opleveren. Deze bijgewerkte versie moet de terugkeerwaarde van de functie `LeegMaken` zijn.

Maar dan nog is voorzichtigheid geboden, want er kan nog vanalles mislopen. Bijvoorbeeld:

```
var categorie = catMgr.OphalenMetGroepEnPersonen(catID);
Debug.Assert(categorie.Groep != null);
Debug.Assert(categorie.GelieerdePersoon.Count() > 0); // nemen we aan

categorie = catMgr.LeegMaken(categorie);
Debug.Assert(catMgr.GelieerdePersoon.Count == 0);
// Bij de aanroep van LeegMaken is het belangrijk dat we daarna verder
// werken met de terugkeerwaarde van LeegMaken, en niet met de
// oorspronkelijke categorie. Want aan de oorspronkelijke categorie
// blijven de GelieerdePersonen gekoppeld, met TeVerwijderen = true.1

// Dik nadeel: op dit moment is categorie.Groep opnieuw null.
```

Het zou beter zijn moest

- AttachObjectGraph van de originele gedetachte entity-graph
- de TeVerwijderern koppelingen verwijderen
- de TeVerwijderen entiteiten loskoppelen, en hun ID opnieuw 0 maken
- Na de SaveChanges alle toegekende nieuwe ID's overgenomen worden in de originele gedetachte graaf.

Dit is op zich niet onmogelijk. Maar dit is wel wat werk. Het is niet zeker of het de moeite waard is om dit te implementeren; het is het al zeker niet voor beta-1.

Voorlopig denk ik dat we er ook geraken als we de 'gele' oplossing gebruiken, en geconcentreerd werken.

DummyDAO's in unit tests

Als business methods die iets verwijderen direct moeten persisteren, en als na het persisteren verder gewerkt moet worden met wat het DAO oplevert, hebben we een probleem bij het unit testen van business methods.

Voor de unittests in de businesslaag mocken we de data access. We geven bijvoorbeeld aan de CategorieenManager het DummyCategorienDao mee als ICategorieenDao.

De method CategorieenManager.LeegMaken zal ICategorieenDao.Bewaren(categorie, cat=>cat.GelieerdePersoon) aanroepen, en het is dus belangrijk dat de DummyCategorienDao iets zinvol oplevert.

Hiervoor heb DummyDao.Bewaren aangepast.

(<https://develop.chiro.be/trac/cg2/browser/trunk/Solution/TestProjecten/Chiro.Gap.Dummies/DummyDao.cs#L51>). De nieuwe Bewaren heeft als resultaat de originele data, maar dan zonder de TeVerwijderen items, en vervangt ID's 0 door niet nulle ints. Ik doe iets analoog als AttachObjectGraph om de 'paths' te doorlopen, en het lijkt te werken.

2. Implementatie van functies op lidniveau

Nationaal bepaalde functies en 'eigen' functies

In de database is een eigen functie een functie die gekoppeld is aan een bepaalde groep. Functies die aan geen enkele groep gekoppeld zijn, worden beschouwd als nationaal bepaalde functies, of kortweg 'nationale functies'.

Dit impliceert dat bij het ophalen van een functie uit de database, steeds zijn koppelingen met een groep bij opgehaald moeten worden. Want het moet te allen prijze vermeden worden dat een eigen functie van een groep plots als nationaal beschouwd wordt.

We gaan dat oplossen door in de tabel 'lid.Functie' een 'computed column' toe te voegen: `IsNationaalBepaald`. Deze zal `true` zijn, als `groep = null`. (Zie #364.)

Als handigheidje heb ik een method

`FunctiesManager.NationaalBepaaldeFunctiesOphalen()` gemaakt, die de lijst van nationaal bepaalde functies oplevert. Deze lijst wordt gecacht.

Overzicht nationaal bepaalde functies

ID	Code	Omschrijving	MinAantal	MaxAantal	Lidsoort	Werkjaar Van	Werkjaar Tot
1	CP	Contactpersoon	1	1	Leid	NULL	NULL
2	GL	Groepsleiding	0	NULL	Leid	NULL	NULL
3	VB	VB	0	NULL	Leid	NULL	NULL
4	FV	Fin. verantwoordelijke	1	1	Leid	NULL	NULL
5	JR	Vertegenw. jeugdraad	0	NULL	Leid	NULL	NULL
6	CK	Coördinator kookploeg	0	NULL	Leid	NULL	NULL
7	PR	Proost	0	NULL	Leid	NULL	NULL

Opmerkingen hierbij:

1. Als `MaxAantal`, `WerkjaarVan`, `WerkjaarTot` niet van toepassing zijn, krijgen ze de waarde `NULL` (#365, #366).
2. Als `MaxAantal` of `WerkjaarTot` 0 is, dan mag je dat als 'oneindig' beschouwen (zijnde er is geen `MaxAantal` of `WerkjaarTot`)
3. De codes zijn overgenomen uit Kipadmin, maar GG1 en GG2 vraagt wat extra uitleg. Vroeger was het zo dat het oude Chirogroepprogramma 1 of 2 groepsleid(st)ers toeliet. De eerste had functie GG1, de tweede GG2. Ondertussen is het zo dat er meerdere GG2's mogelijk zijn, maar een GG1 is er hoogstens 1, en dat blijft de contactpersoon. Vandaar dat ik voorstel om GG1 contactpersoon te noemen, en GG2 'groepsleiding'. Dat de contactpersoon zowel functies GG1 als GG2 heeft, is wat mij betreft geen bezwaar.
4. We pasten de functiecodes aan, zodat ze duidelijker zijn. (Zie #367.)

Nationale functies gebruiken in code

In sommige gevallen lijkt het me nuttig om bijvoorbeeld in de businesslaag de contactpersoon (of een andere nationale functie) op te vragen. Bijvoorbeeld om in het overzicht van de groep te tonen. Voor dit soort situaties heb ik een enum `NationaleFunctie` gedefinieerd.

```
public enum NationaleFunctie
{
    Geen = 0,
```

```

        ContactPersoon = 1,
        GroepsLeiding = 2,
        Vb = 3,
        FinancieelVerantwoordelijke = 4,
        JeugdRaad = 5,
        KookPloeg = 6,
        Proost = 7
    };

```

De entiteit `Functie` heeft een property `VoorgedefinieerdeFunctie` gekregen, die het juiste `VoorgedefinieerdeFunctieType` oplevert als de functie gepredefinieerd is, en anders `VoorgedefinieerdeFunctieType.Geen`.

```

public VoorgedefinieerdeFunctieType VoorgedefinieerdeFunctie
{
    get
    {
        // Constructie met linq, en niet gewoon conversie,
        // om default (0, 'Geen') te krijgen
        // als de functiecode niet overeenkomt met
        // een Voorgedefinieerde functie.
        return (from VoorgedefinieerdeFunctieType f in
Enum.GetValues(typeof(VoorgedefinieerdeFunctieType))
                where (int)f == ID
                select f).FirstOrDefault();
    }
}

```

Dit is niet zo interessant, omdat bij het accessen van deze property iedere keer een loopje uitgevoerd wordt. Beter is iets in deze stijl: (Zie #368.)

```

// Declaratie:
bool Functie.Is (NationaleFunctie vgf)
{
    return (ID == (int)vgf)
}

// Bijv:
if (f.Is(NationaleFunctie.ContactPersoon))
{
    // blablabla
}

```

Controle aantallen

In de vorige vergadering spraken we af dat we het toekennen/afnemen van een functie niet zouden tegenhouden als `MaxAantal` of `MinAantal` niet gerespecteerd wordt. Er is een method `'AantallenControleren'` die de aantallen nakijkt, en inidien van toepassing een lijst met problemen oplevert

(<https://develop.chiro.be/trac/cg2/browser/trunk/Solution/Chiro.Gap.Workers/FunctiesManager.cs#L189>). Op het moment dat ik dit schrijf, is deze method nog niet af.

Functies vervangen

Tegen de vergadering hoop ik de service method af te hebben die de functies van een persoon vervangt door een lijstje functies.

Varia

- Instellingen zoals
 - o Minimumchiroleeftijd leden (6 jaar)
 - o Dagen instapperiode
 - o ...

We definiëren deze in de settingsfile van een project. (Relevante tickets: #329, #271, #244)
- Adressensysteem (stilaan een vast puntje :))
 - o We kunnen de tabellen van Peter (uit de database Crab_Koen) gebruiken voor het adressensysteem dat we vorige keer afspraken:
 - created_peterpostomschrijving
 - PeterStraat
 - o Wat moeten we nog doen:
 - aan beide tabellen een ID toevoegen
 - In PeterStraat een postnummer toevoegen (er is al een pkanid, en gelukkig bestaat er een 1-1 relatie pkanid-postnummer)
 - o Wat is het verschil met de plaatsnamentabel van Tommy:
 - 2020 Antwerpen – 2020 Antwerpen (Kiel)
 - o Wat kunnen we wel recupereren uit Tommy's bestand:
 - De gemeentes, en bijhorende postnummers. (kan misschien ook uit Crab; Peter bekijkt dit.)
 - De koppeling gemeente-provincie.
 - Extra 'typische' postnummers (vrt, nato,...)
- Cachekeys vastleggen in constanten; definiëren in klasse/scope waar ze nodig zijn
- Overgang naar het nieuwe werkjaar (ook een bijna-vast puntje)
 - o We hadden het document van Peter en Broes verkeerd geïnterpreteerd
 - o Het moest zijn:
 - Overgang moet gebeuren voor 1 oktober
 - Instapperiode = max(nu + 3 weken, 21 oktober)
 - (Let op de verschillende data in 1ste en 2de puntje)
- Publieke testsite
 - o Testaccount maken voor beta 2
 - Jiska
 - Hans Bouwen kj /0202
 - o Peter mailen: Hoe aanmelden op publieke test
 - o Nakijken toegang Broes publieke test

Programmeerweekend

- Datum: 22 en 23 mei
- Plaats: Indien mogelijk Heidepark, anders Heibrand. Ik laat iets weten.
- Johan zorgt voor vpn-router en een databaseserver voor het geval de connectie naar devserver te traag is.

Openstaande tickets voor beta-1

Ticket	Summary	Owner	Component	Blocked By
#237	Adressensysteem (opnieuw)	--	adressen	232, 332
#300	Dubbele deelgemeentes in testdatabase	--	adressen	
#303	Dubbele adressen bij een persoon	--	adressen	
#332	Trigger voor controle postnummer in woonplaats en adres	haepeto	adressen	
#289	Controle op geboortejaren bij activeren afdelingen (van, tot)		afdelingen	
#325	Duidelijke foutmelding als nieuwe afdeling maken mislukt	--	afdelingen	
#326	Afdeling bewerken doet niet dezelfde controles als afdeling activeren	--	afdelingen	
#295	Export naar Excel	--	algemeen	
#351	Stroomlijnen 'OphalenMet...'-functies van workers		architectuur	352
#353	LedenManager.Ophalen(funcielID, groepsWerkJaarID) -> FunctiesManager.LedenOphalen		architectuur	
#336	Documentatie LaTeX (voor coding standard)		documentatie	
#348	Problemen met toegekende functies tonen		functies	
#356	Controle juiste groep bij functies koppelen		functies	
#103	functies		functies	328, 330, 337, 347, 348
#179	Bij een lid op 'afdelingen' klikken crasht als er geen afdelingen zijn		ledenbeheer	
#13	view voor lid bewerken	decatbr	ledenbeheer	329
#244	leeftijdsgrenzen afdelingsjaren		ledenbeheer	
#272	Instapperiode op lidniveau en niet op kindniveau		ledenbeheer	
#329	Hardgecodeerde leeftijden in ledenmanager.	--	ledenbeheer	
#93	Extra info van communicatievormen ergens weergeven	--	personenbeheer	
#220	Is dit de voorkeurscommunicatie voor dit communicatietype?		personenbeheer	
#221	Vakje 'extra informatie' voor communicatievorm		personenbeheer	
#308	Iedereen tegelijk selecteren in de personenpagina's lukt niet		personenbeheer	
#331	SecurityProbleem in LedenManager.AanpassenAfdelingenVanLid		security	
#245	Links naar ledenlijsten per afdeling		technisch	
#271	Aantal dagen instapperiode hoort thuis in settings ipv resources	decatbr	technisch	
#263	WatinTest LidMakenPersonen failt		tests	
#344	DummyData-klasse moet niet statisch zijn	vervijo	tests	

Als je aan iets werkt, ken die taak dan aan jou toe, zodat we geen dubbel werk doen!