



2. Contextdiagrammen en Data Flow Diagrammen

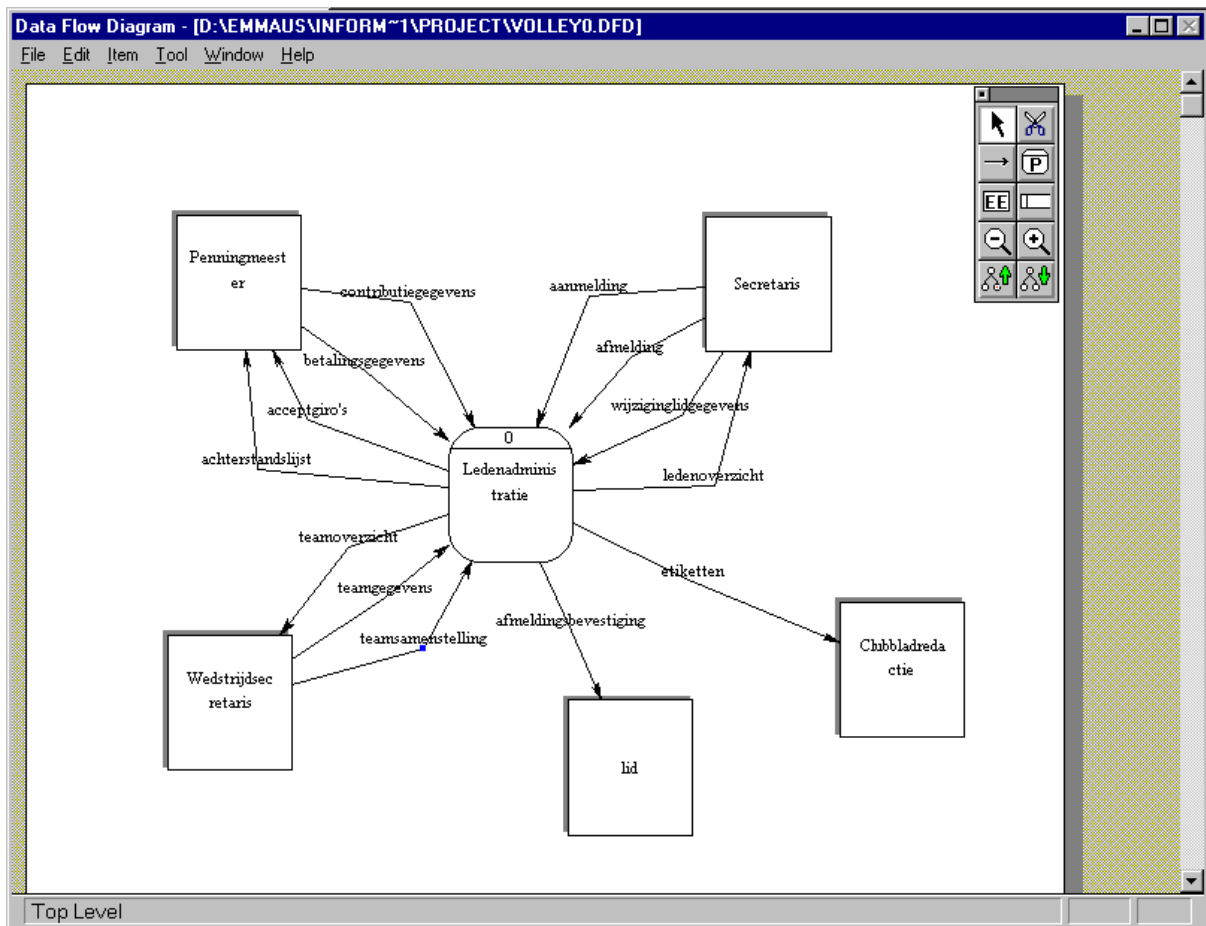
Met behulp van Contextdiagrammen en Data Flow Diagrammen beschrijven we globaal via welke processen de systeemfuncties een plaats krijgen in het informatiesysteem.

2.1 Contextdiagram

Het contextdiagram geeft weer welke gegevensstromen er tussen de buitenwereld en het informatiesysteem bestaan. Daarmee beschrijft het contextdiagram tevens de systeemgrenzen: wat het systeem wél moet kunnen en wat het systeem niet moet kunnen.

Als uitgangspunt bij het maken van een contextdiagram dient het systeem van eisen.

We maken met behulp van het programma DFD(.exe) het contextdiagram bij de uitgewerkte CASE opdracht “de volleybalvereniging”. Het resultaat is het volgende contextdiagram:



Hieronder is beschreven hoe het diagram is gemaakt.



Met de werkset van het programma DFD maak je de volgende onderdelen van het contextdiagram:


- het proces dat je bekijkt (in het contextdiagram is dat de ledenadministratie)
- de externe entiteit (in het contextdiagram is dat de component uit de buitenwereld)
- de gegevensstroom

Het enige dat je bij elk van de onderdelen hoeft in te vullen is de naam die je aan het betreffende onderdeel van het diagram geeft (*External Entity Name*, *Process Name* en *Name* van de gegevensstroom of Input Dialog). Slechts de naam wordt zichtbaar in het diagram.

**Opmerking:**

Het contextdiagram geeft een oppervlakkige beschrijving van het informatiesysteem. Het contextdiagram wordt ook wel een Data Flow Diagram van het 0-de niveau genoemd. Dit is er ook de reden van dat bij het tekenen van het informatiesysteem (hier: de ledenadministratie) als *Process Number 0* is ingevoerd

Let op:

Let op het volgende als je een gegevensstroom tekent: na het kiezen van de knop  uit de werkset moet je eerst op de component waarvandaan de pijl vertrekt en daarna op de component waar de pijl aankomt te klikken.

Tip:

Het programma DFD biedt geen mogelijkheden om de gemaakte diagrammen in een BMP of JPG formaat te exporteren of zelfs via knippen en plakken over te brengen naar een tekstverwerkingsprogramma of tekenprogramma. Met <Alt> + <PrintScrn> kan je echter het complete venster van het DFD programma (en overigens ook van elk ander programma!) op het prikbord zetten. Met de optie <Plakken> kan je het diagram vervolgens in een tekstverwerkingsprogramma of tekenprogramma opvragen.

2.2 Functionele decompositie

De systeemeisen kunnen worden vertaald in processen. De processen geven een omschrijving van de elementaire handelingen die met het systeem verricht moeten kunnen worden.

Uitgaand van de elementaire processen die voortkomen uit het systeem van eisen vindt clustering tot hoofdprocessen plaats. De structuur van hoofdprocessen en daaronder vallende elementaire processen wordt ook wel de functionele decompositie van het informatiesysteem genoemd.

De elementaire processen die we bij de ledenadministratie van de volleybalvereniging onderscheiden zijn:

1. verwerk aanmelding
2. verwerk wijziging lidgegevens
3. maak ledenoverzicht
4. leg teamgegevens vast
5. leg teamsamenstelling vast
6. maak teamoverzicht
7. leg hoogte contributie vast
8. verwerk betaling
9. maak achterstandslijst
10. maak acceptgiro's
11. maak etiketten
12. verwerk afmelding
13. maak bevestiging afmelding

De elementaire processen kunnen worden geclusterd of samengevoegd tot hoofdprocessen. De belangrijkste reden om processen te clusteren is de functionele samenhang van de processen. De processen hebben onderling namelijk wel iets met elkaar te maken. Enerzijds hoort bij elk proces uit het informatiesysteem een bepaalde gebruiker. Er zijn dus groepen of clusters van processen die bij eenzelfde gebruiker horen. Anderzijds maakt elk proces gebruik van een (aantal) gegevensbestand(en) uit het systeem. Er zijn dus groepen of clusters van processen die van dezelfde gegevensbestanden gebruik maken.



Het is dus van belang om in deze fase van de modellering van het informatiesysteem een grove afbakening te maken van de gegevensbestanden die in het systeem nodig zijn. Het ligt voor de hand om de lidgegevens (naam, adres, etc.) vast te leggen in een aparte tabel, de contributiegegevens (categorie, bondscontributie, verenigingscontributie) vast te leggen in een aparte tabel en de teamgegevens (team, klasse, trainer, coach) vast te leggen in een aparte tabel. Dat betekent dat er, in het informatiesysteem dat we gaan bouwen, drie gegevensbestanden (ook wel buffers genoemd) nodig zijn:

- A. lidgegevens
- B. contributiegegevens
- C. teamgegevens

Hieronder is per elementair proces aangegeven welke gegevensbestanden het proces nodig heeft:

proces	gebruiker	gebruikt gegevensbestand
1. verwerk aanmelding	secretaris	A
2. verwerk wijziging lidgegevens	secretaris	A
3. maak ledenoverzicht	secretaris	A
4. leg teamgegevens vast	wedstrijdsecretaris	C
5. leg teamsamenstelling vast	wedstrijdsecretaris	A
6. maak teamoverzicht	wedstrijdsecretaris	A, C
7. leg hoogte contributie vast	penningmeester	B
8. verwerk betaling	penningmeester	A
9. maak achterstandslijst	penningmeester	A
10. maak acceptgiro's	penningmeester	A, B
11. maak etiketten	secretaris	A
12. verwerk afmelding	secretaris	A
13. maak bevestiging afmelding	secretaris	A

Op basis van de gebruikers en gebruikte gegevensbestanden zijn de elementaire processen geclusterd tot de volgende hoofdprocessen:

1. verwerk lidgegevens
 - (1. verwerk aanmelding, 2. verwerk wijziging, 3. maak ledenoverzicht, 11. maak etiketten, 12. verwerk afmelding, 13. bevestiging afmelding)
2. verwerk contributiegegevens
 - (7. leg hoogte contributie vast, 8. verwerk betaling, 9. maak achterstandlijst, 10. maak acceptgiro's)
3. verwerk teamgegevens
 - (4. leg teamgegevens vast, 5. leg teamsamenstelling vast, 6. maak teamoverzicht)

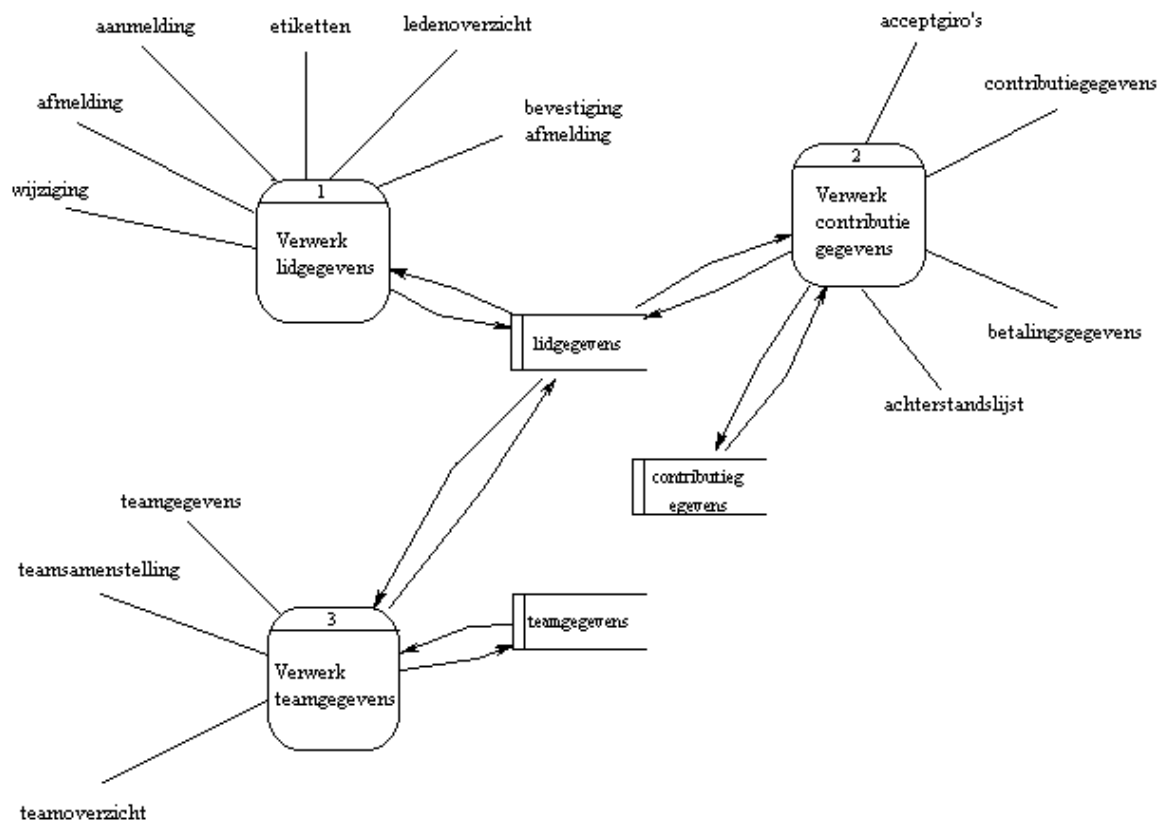
N.B. Aangezien de buitenwerelden "lid" en "clubbladredactie" in werkelijkheid niet om zullen gaan met het informatiesysteem zijn de processen "maak bevestiging afmelding" en "maak etiketten" ondergebracht in het hoofdproces "verwerk lidgegevens", het proces dat wordt aangestuurd door de secretaris.



2.3 Data Flow Diagram van het eerste niveau

We kunnen het contextdiagram nu verder gaan detailleren op basis van de gemaakte functionele decompositie. Dat klinkt allemaal heel ingewikkeld maar het komt hierop neer dat we verder gaan inzoomen op de processen die binnen de ledenadministratie.

Daarbij geef je aan welke gegevensbestanden de verschillende (hoofd)processen gebruiken. Wat ontstaat is een diagram waarin is weergegeven op welke wijze de processen gebruik maken van de gegevensbestanden. We noemen zo'n diagram een Data Flow Diagram (DFD). Omdat we bij het inzoomen starten met het weergeven van de hoofdprocessen noemen we het diagram ook wel een DFD van het eerste niveau.



Uit het DFD kan je bijvoorbeeld aflezen dat voor het maken van acceptgiro's de tabellen met lidgegevens en contributiegegevens dienen te worden geraadpleegd.

Nu met DFD een beschrijving is gegeven van de processen die in het informatiesysteem opgenomen dienen te worden en van de gegevenstabellen die bij deze processen gebruikt dienen te worden kan een begin worden gemaakt van een meer gedetailleerd ontwerp van het informatiesysteem.