

ONGERUBRICEERD

TNO-rapport**35495****Flexigas Simulator Architectuur (versie 1.0)****Technical Sciences**
Eemsgolaan 3
9727 DW Groningen
Postbus 1416
9701 BK Groningenwww.tno.nlT +31 88 866 70 00
F +31 88 866 77 57
infodesk@tno.nl

Datum	21 april 2011
Versie	1.0
Auteur(s)	Marc de Jonge, Jeroen Broekhuijsen, Matthijs Vonder
Reviewers	Kristian Helmholt, Allart Bastiaans
Aantal pagina's	34 (incl. bijlagen)
Opdrachtgever	Suzanne van Kooten (Innovatiegebied Energie Efficiëntie)
Projectnaam	Flexigas
Projectnummer	035.33938

Alle rechten voorbehouden.

Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt door middel van druk, foto-kopie, microfilm of op welke andere wijze dan ook, zonder voorafgaande toestemming van TNO.

Indien dit rapport in opdracht werd uitgebracht, wordt voor de rechten en verplichtingen van opdrachtgever en opdrachtnemer verwezen naar de Algemene Voorwaarden voor opdrachten aan TNO, dan wel de betreffende terzake tussen de partijen gesloten overeenkomst.

Het ter inzage geven van het TNO-rapport aan direct belang-hebbenden is toegestaan.

© 2011 TNO

ONGERUBRICEERD

Inhoudsopgave

1	Inleiding	3
1.1	Achtergrond informatie	3
1.2	Doel	5
1.3	Opbouw document	5
2	Eisen aan de architectuur.....	6
2.1	Woordenlijst	6
2.2	Functionele eisen	7
2.3	Non-functionele eisen.....	15
3	High-level ontwerp	18
3.1	Logische gezichtspunt	18
3.2	Proces gezichtspunt	23
3.3	Ontwikkel gezichtspunt.....	24
3.4	Fysieke gezichtspunt	25
3.5	Scenario's.....	26
4	Mogelijke toekomstige uitbreidingen.....	27
5	Bijlage A: Paper-prototype	28

1 Inleiding

Doel van dit document is om een (high level) architectuur te beschrijven aan de hand waarvan binnen het project Flexigas een simulator voor een Smart Biogas Grid ontwikkeld kan worden. Het document is primair geschreven voor de ontwikkelaars van de simulator, maar zal ook gebruikt worden in de discussie met de beoogde eindgebruikers.

In het project Flexigas (waarin wordt samengewerkt in verschillende thema's¹ met diverse partners) bouwt TNO kennis op op het gebied van een Smart Biogas Grid. In de volgende paragrafen wordt meer achtergrond informatie gegeven over het Flexigas project en het specifieke Thema A (ketenoptimalisatie) waarin TNO werkzaam is.

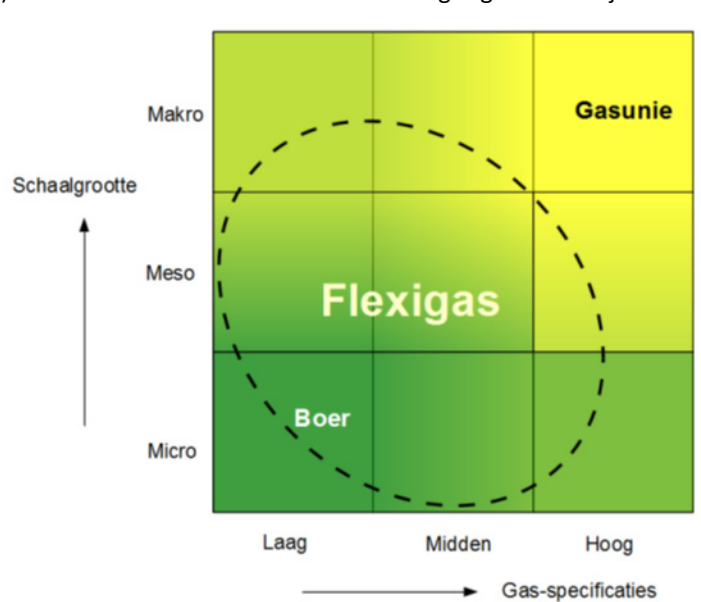
Het voorliggende document is een eerste versie van de afgesproken deliverable: *A3.1 Architectuur rapportage*. In verschillende iteraties wordt het document aangepast en/of uitgebreid.

1.1 Achtergrond informatie

Er wordt een Europees plan voorbereid dat inzet op een verdere aanscherping van de reductie van CO₂-emissie: 30% in plaats van 20% reductie in 2020 (t.o.v. 1990). Vóór de verzwaar van deze eis had het kabinet reeds als doel geformuleerd dat in 2020 jaarlijks 4 miljard m³ aardgas (zo'n 10% van het Nederlandse totaalverbruik) moet zijn vervangen door groen gas. Wetende dat de huidige productie op 10 á 15 miljoen m³ ligt is er nog een lange weg te gaan.

1.1.1 Flexigas project

Biogas wordt nu óf op kleinschalig niveau verstreemd tegen laag economisch- en milieurendement óf opgewerkt tot groen gas, een optie die kostentechnisch alleen rendeert bij grootschalige opwerkinstallaties (groen gas mag ingebracht worden in het aardgasnet). De inzet is te komen tot een smart biogas grid waarbij afstemming van vraag-



Figuur 1: Positionering Flexigas

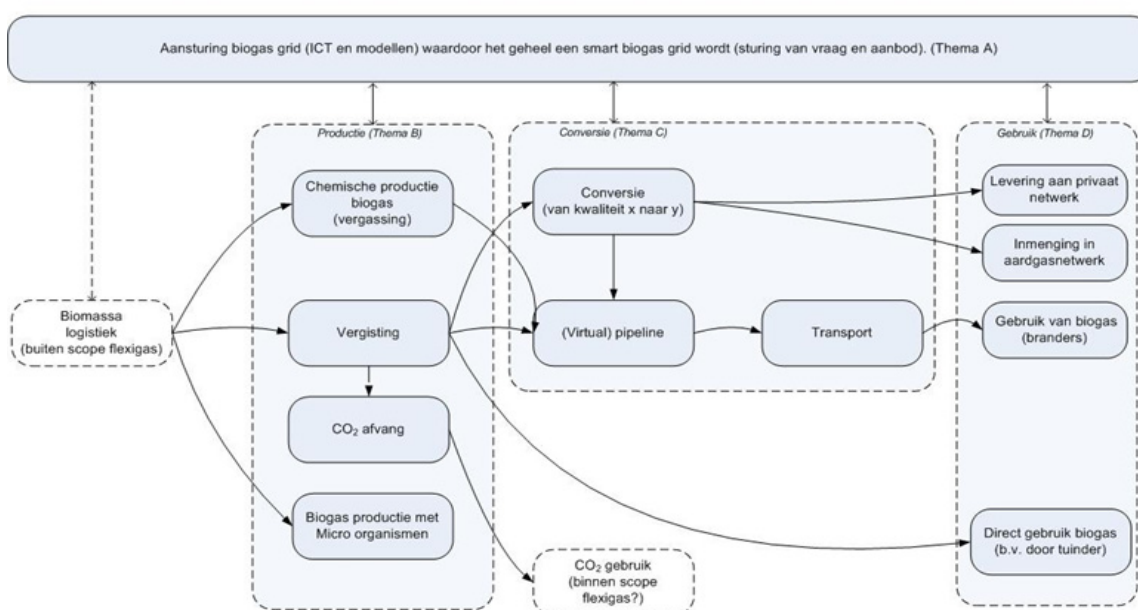
en aanbod patronen economisch geoptimaliseerd worden en waarbij niet (noodzakelijkerwijs) opgewerkt wordt tot groen gas kwaliteit. De positionering van Flexigas wordt geïllustreerd in Figuur 1.

1.1.2 Flexigas-thema's

De verschillende onderzoekslijnen binnen Flexigas zijn georganiseerd naar de volgende thema's:

- Thema A: Ketenoptimalisatie
- Thema B: Productie/vergisting
- Thema C: Conversie, opslag en transport
- Thema D: Gebruik biogas
- Thema E: Valorisatie (opleiding)

De samenhang tussen deze thema's is weergegeven in Figuur 2.



Figuur 2: Samenhang tussen de thema's in Flexigas

1.1.2.1 Thema A: Ketenoptimalisatie

TNO is werkzaam in Thema A (ketenoptimalisatie). Dit thema richt zich op de systeemoptimalisatie en integratie van een smart biogas grid (keten). Hierbij wordt onderzocht hoe de huidige en nieuwe technologieën optimaal op elkaar afgestemd kunnen worden. Bij optimaal kan worden gedacht aan: economisch, duurzaam, leveringsbetrouwbaar en technisch. Van de losse technologieën is kennis en ervaring, maar er is nog weinig inzicht hoe de ketens optimaal (zie boven) georganiseerd kunnen worden. Bovendien spelen er lange termijn zaken op technisch en maatschappelijk vlak, waarvan de impact op een smart biogas grid nog onzeker is. Doelstelling van het thema is: inzicht krijgen in toekomstige scenario's en ketenmodellen.

¹ Met thema in dit document wordt bedoeld een onderzoekslijn binnen Flexigas, niet te verwarren met de 7 marktthema's van TNO.

1.1.2.2 *Aandachtsgebieden TNO in Thema A*

Een dergelijk smart biogas grid bestaat nog niet en zal ontwikkeld worden, waarbij TNO zich zal richten op de ICT aspecten van dit grid. Ten behoeve van o.a. investeringsbeslissingen is er behoefte aan inzicht in de (on)mogelijkheden van een dergelijk grid en de dynamiek ervan wordt een simulator gemaakt waarin verschillende scenario's kunnen worden nagebootst. Om een dergelijke simulator te maken is eerst een architectuur beschrijving benodigd.

1.2 **Doel**

Doel van dit document is om een (high level) architectuur te beschrijven aan de hand waarvan de simulator voor een Smart Biogas Grid ontwikkeld kan worden. Op dit moment wordt nog open gehouden op welke wijze de ontwikkeling plaats gaat vinden. In verband met de kosten en beoogde samenwerking binnen en buiten Flexigas gaat de voorkeur uit naar het configureren van een bestaande simulatie omgeving (zoals bijvoorbeeld Open Modelica²) boven het zelf volledig uitprogrammeren van een simulator (in bijvoorbeeld Java of C#).

1.3 **Opbouw document**

In het volgende hoofdstuk worden zowel de functionele als non-functionele eisen beschreven die aan de architectuur van het simulatie systeem gesteld worden. De functionele eisen worden beschreven aan de hand van een aantal use-cases. De non functionele eisen worden beschreven aan de hand van het zogenaamde Quint model. Hoofdstuk 3 beschrijft het "high-level" ontwerp waarbij gebruik gemaakt wordt van het "4+1 view model". Hoofdstuk 4 bevat een opsomming van mogelijke toekomstige uitbreidingen van dit architectuur document. In de bijlage wordt ter illustratie een zogenaamd "paper prototype" getoond. Dit zijn getekende schermen die de simulator straks zou kunnen laten zien. Ze zijn gebruikt bij het uitwerken van de use-cases en geven de lezer een beter beeld. De schermen in de uiteindelijke versie zullen zeker afwijken van de schermen uit de bijlage.

² Zie <http://www.openmodelica.org>

2 Eisen aan de architectuur

Dit hoofdstuk beschrijft wat de eisen zijn aan het simulatie systeem dat gebouwd zal worden voor het Flexigas project. Eerst wordt een woordenlijst gegeven waarin verschillende termen beschreven staan. Daarna zullen de functionele specificaties opgesteld zijn in de vorm van use-cases. Het laatste deel zal de non-functionele eisen beschrijven.

In dit document wordt gesproken van (eind)gebruikers. Voorlopig zijn dit de bouwers van de simulator zelf. In tweede instantie wordt gedacht aan de partners binnen Flexigas die zelf bezig zijn met optimalisatie berekeningen (bijv. EKC via excel sheets). Daarna moet het geschikt zijn voor toepassers (zoals Rendo en BioBench) en studenten en promovendi (van RUG en EKC) en natuurlijk TNO collega's werkzaam voor/binnen het Thema Energie.

2.1 Woordenlijst

Connector	Aansluitpunt van een model, waarbij waardes overgedragen kunnen worden van een uitvoerconnector (van het ene model) naar een invoerconnector (van het andere model) via een connectie.
Connectie	Een verbinding tussen een uitvoerconnector en een invoerconnector (van twee modellen).
Interface	Datgene waar de gebruiker interactie mee heeft, om gegevens te kunnen lezen of te kunnen invoeren in het systeem. Soms ook wel aangegeven met UI (User Interface).
Model	Een model is een vereenvoudigde weergave van (een deel van) de werkelijkheid. Iets specifieker gaar het hier om een afgebakende invulling van functionaliteit. De afbakening kan zowel het hele scenario omvatten als het kleinste onderdeel ervan. Om een model op te bouwen kan er gebruik gemaakt worden van: connectoren, connecties, parameters, andere modellen, etc. Een model is mogelijk een invulling van een raamwerkmodel.
Modelverzameling	Een verzameling modellen welke van tevoren gedefinieerd zijn en herbruikbaar zijn voor in te voeren scenario's.
Opdracht	Een opdracht richting het systeem om een simulatie of een optimalisatie uit te voeren voor een geselecteerd scenario. Als de opdracht uitgevoerd is, zijn er resultaten aan gekoppeld.
Optimalisatie	Uitrekenen van een optimale combinatie van parameters waardoor een opgegeven functie gemaximaliseerd wordt. Voor een optimalisatie kan door de gebruiker worden aangegeven welke parameters een vaste waarde hebben en welke bepaald moeten worden.
Parameter	Een waarde (vaak een getal) die het gedrag van het model kan aanpassen. Bijvoorbeeld: het aantal kuub gas dat maximaal verwerkt kan worden.

Raamwerkmodel	Een generieke omschrijving van een model zonder implementatie. Deze bevat een opsomming van de connectoren, de parameters, de resultaatwaardes en eventueel een aantal voorwaarden, maar dus geen functionele invulling.
Raamwerkverzameling	Een verzameling raamwerkmodellen welke van tevoren gedefinieerd zijn en in te vullen zijn door modellen.
Scenario	De beschrijving van de door te rekenen samenstelling van modellen, welke wordt omschreven door het koppelen van modellen en het toekennen van waarden aan parameters.
Simulatie	Het doorrekenen van een (of meerdere) scenario('s) over een vastgestelde tijdsperiode. Het resultaat van een simulatie is de uitvoer van alle modellen over de tijd.
Systeem	Het gehele systeem dat door dit document beschreven wordt. Dit omvat zowel de interface om te bewerken, als alle achterliggende systemen om simulaties uit te voeren of om het systeem te beheren.
Template	Een scenario die als voorbeeld dient, om de gebruiker op weg te helpen bij het samenstellen van een specifiek scenario.

2.2 Functionele eisen

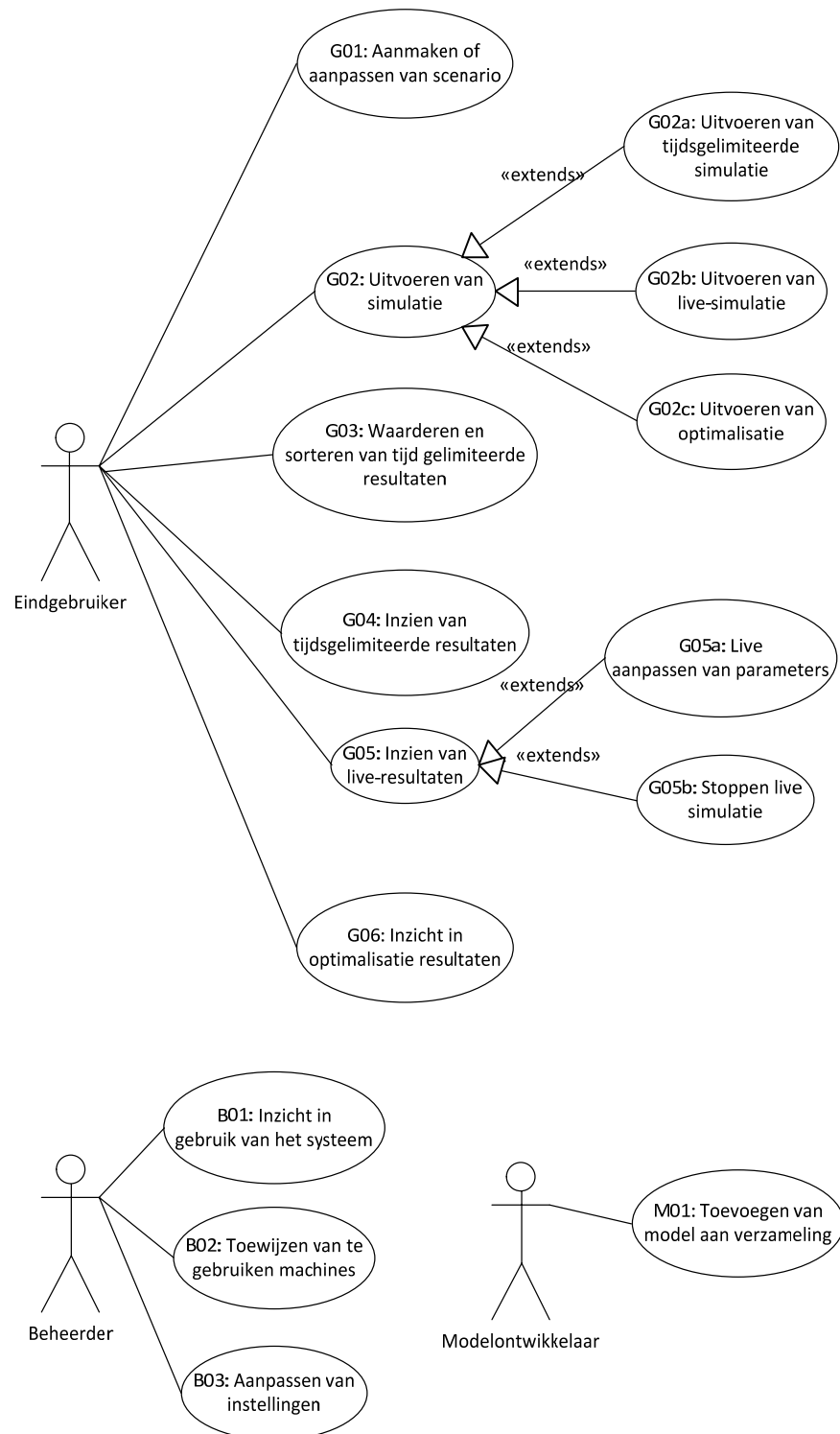
Deze paragraaf beschrijft de use cases van de Flexigas simulator. Hierin zijn drie type gebruikers te definiëren:

- De *eindgebruiker* is degene die de simulator gebruikt om scenario's door te rekenen, om hiermee inzicht te krijgen in de keten.
- De *modelontwikkelaar* is een expert in het maken van modellen in de specifieke taal die hiervoor benodigd is.
- De *beheerder* is een expert binnen het systeem die verantwoordelijk is voor de correcte werking van het systeem met voldoende rekenkracht om de simulaties te kunnen uitvoeren.

Alle use cases zijn eerst beschreven door middel van een overzichtsplaatje (zie Figuur 3), waarna elk van de use cases uitgewerkt is.

2.2.1 Use case Diagram

In Figuur 3 staat een diagram waarin alle use-cases omschreven zijn.



Figuur 3: Use case diagram

2.2.2 Use cases van de eindgebruiker

In bijlage A wordt ter illustratie een zogenaamd “paper prototype” getoond. Dit zijn getekende schermen die de simulator straks zou kunnen laten zien. Ze zijn gebruikt bij het uitwerken van de use-cases en geven de lezer een beter beeld. De schermen in de uiteindelijke versie zullen zeker afwijken van de schermen uit de bijlage.

UC G01	Aanmaken of aanpassen van een scenario
Omschrijving	De gebruiker kan op de UI een scenario in elkaar zetten of aanpassen aan de hand van template(s) of een eerder bewaard scenario (of scenario's).
Stappenplan	- De gebruiker selecteert een template of een opgeslagen scenario - De gebruiker geeft optioneel een unieke naam aan het scenario - De gebruiker past het scenario aan door modellen toe te voegen, te verwijderen, connecties te maken. - De gebruiker kiest bestaande parameter setting om aan te passen of maakt nieuwe setting aan - Bij parameters kan de gebruiker een waarde opgeven, maar ook een bereik (beginwaarde, eindaarde en stapgrootte) - Het geheel aan gekozen parameterwaarden kan worden bewaard als parametersetting. Gebruiker kiest optioneel een unieke naam. - De gebruiker geeft aan het scenario te willen bewaren
Rollen	Eindgebruiker
Pre-condities	Er zijn voldoende geïmplementeerd modellen in het raamwerk aanwezig om een volledig scenario te beschrijven Er zijn templates beschikbaar om uit te kiezen
Post-condities	Het scenario is bewaard en kan later opnieuw opgevraagd worden De parameters zijn ook bewaard en kunnen ook opgevraagd worden

UC G02	Uitvoeren van een simulatie
Omschrijving	Voor een gekozen scenario met parameters bepaalt de gebruiker dat deze gesimuleerd moet worden.
Stappenplan	- De gebruiker selecteert een scenario - De gebruiker selecteert een bijbehorend parameter-instelling
Rollen	Eindgebruiker
Pre-condities	Er is een eerder gedefinieerd scenario geselecteerd Alle parameters hebben een waarde en mogelijk een bereik
Post-condities	De opdracht is in behandeling genomen door het systeem en zal uiteindelijk gestart worden Het systeem zal na afloop van de uitvoering van de opdracht, de resultaten van de simulatie bevatten

UC G02a	Uitvoeren van tijd gelimiteerde simulatie
Basis-usecase	G02: Uitvoeren van een simulatie
Stappenplan	- De gebruiker geeft aan dat hij een tijd gelimiteerde simulatie wil uitvoeren - De gebruiker selecteert de gewenste tijdsspanne - De gebruiker geeft aan dat hij de simulatie wil starten
Post-conditie	Er zijn automatisch een of meerdere opdrachten gegenereerd, die in behandeling genomen zijn door het systeem. Het totaal aantal opdrachten hangt af van het aantal parameters waarvoor een bereik is opgegeven en hoe groot deze bereiken zijn.

UC G02b	Uitvoeren van live-simulatie
Basis-usecase	G02: Uitvoeren van een simulatie
Stappenplan	- De gebruiker kiest de gewenste simulatie snelheid (sneller of langzamer of gelijk aan realtime) - De gebruiker geeft aan dat hij een live simulatie wil uitvoeren
Post-condities	Het systeem zal na enige tijd de simulatie starten, waarna er continue resultaten beschikbaar zullen zijn

UC G02c	Uitvoeren van optimalisatie
Basis-usecase	G02: Uitvoeren van een simulatie
Stappenplan	- De gebruiker geeft aan dat hij een optimalisatie wil uitvoeren - De gebruiker geeft aan met welke waarderingsfunctie de optimalisatie uitgevoerd moet worden of maakt nieuwe waarderingsfunctie aan - De gebruiker selecteert de gewenste tijdsspanne - De gebruiker geeft aan dat hij de optimalisatie wil starten
Post-condities	De resultaten zullen bestaan uit een set van optimale parameters De resultaten zijn na enige tijd beschikbaar als nieuwe parameter instelling voor hetzelfde scenario

UC G03	Waarderen en sorteren van tijd gelimiteerde resultaten
Omschrijving	De gebruiker kan alle parameter instellingen van een tijd-simulatie-opdracht sorteren aan de hand van een functie die uitgevoerd wordt op de resultaten. Dit maakt het makkelijker om de resultaten te beoordelen.
Stappenplan	- De gebruiker selecteert een scenario waarvoor hij een tijd gelimiteerde opdracht wil opzoeken - De gebruiker selecteert een tijd gelimiteerde opdracht, waarvan hij de resultaten gesorteerd wil zien worden - De gebruiker kan een functie opgeven om de sortering uit te kunnen voeren - Het systeem zal na enige tijd de opdracht hebben verwerkt
Rollen	Eindgebruiker
Pre-condities	Er is een tijd gelimiteerde opdracht uitgevoerd voor het geselecteerde scenario
Post-condities	De parameter instellingen zijn nu gesorteerd aan de hand van de waarde die de functie oplevert op de gerelateerde resultaten

UC G04	Inzien van tijd gelimiteerde resultaten
Omschrijving	De gebruiker kan de resultaten van een tijdsgebonden simulatie opvragen en kan hier inzicht in verkrijgen
Stappenplan	- De gebruiker selecteert een scenario waarvoor hij resultaten wil bekijken - De gebruiker selecteert een specifieke parameter instelling en een tijd-simulatie-opdracht waarvoor hij resultaten wil bekijken - De gebruiker selecteert de parameters welke hij wil bekijken - De gebruiker krijgt een grafiek te zien waarin de geselecteerde resultaten getoond worden - De gebruiker kan de selectie op de volgende manier wijzigen, waarbij de grafiek direct bijgewerkt zal worden: - De gebruiker kan andere parameters aan of uit zetten - De gebruiker kan mogelijk een uitgevoerde sortering selecteren - De gebruiker kan resultaten van andere parameter instellingen aan of uit zetten - De gebruiker kan resultaten van andere tijd-simulatie-opdrachten aan of uit zetten
Rollen	Eindgebruiker
Pre-condities	Er is een tijd-simulatie-opdracht correct uitgevoerd door het systeem, waarvoor de resultaten beschikbaar zijn
Post-condities	De opgeslagen resultaten zijn niet gewijzigd

UC G05 Inzien van live resultaten	
Omschrijving	De gebruiker kan de resultaten van een live simulatie opvragen en kan hier inzicht in verkrijgen
Stappenplan	- De gebruiker selecteert scenario waarvoor een live opdracht op dit moment wordt uitgevoerd - De gebruiker selecteert de parameters die hij wil bekijken - De gebruiker krijgt een grafiek te zien waarin de geselecteerde resultaten getoond worden - Telkens wanneer er nieuwe resultaten ontvangen worden van de live-simulatie, wordt de grafiek bijgewerkt
Rollen	Eindgebruiker
Pre-condities	Er wordt op dit moment een live-simulatie uitgevoerd voor het geselecteerde scenario
Post-condities	Geen

UC G05a Live aanpassen van parameters	
Omschrijving	De gebruiker kan tijdens het bekijken van de live resultaten de parameters van het scenario aanpassen en daarvan de resultaten zien
Stappenplan	Als in G05, waarna: - De gebruiker past een van de parameters aan in het scenario - De grafiek wordt bijgewerkt wanneer de live scenario deze parameter heeft meegenomen
Post-condities	De aanpassingen van de parameters worden niet opgeslagen

UC G05b Stoppen van de live-simulatie	
Omschrijving	De gebruiker kan de live-simulatie stoppen vanuit UC G05 en UC G05a
Stappenplan	- De gebruiker geeft aan dat hij de live-simulatie wil stoppen - Het systeem vraagt aan de gebruiker of hij de aanpassingen wil opslaan als een nieuwe parameter instelling of niet - De gebruiker kiest een van deze opties
Post-condities	Als de gebruiker ervoor kiest op de aanpassing op te slaan, zal het systeem de nieuwe versie van de parameter instelling bewaard hebben De live-simulatie zal na enige tijd gestopt zijn De resultaten worden niet opgeslagen

UC G06	Inzicht in optimalisatie resultaten
Omschrijving	De gebruiker kan de resultaten van een optimalisatie opdracht inzien zoals verkregen uit G02c
Stappenplan	- De gebruiker selecteert een scenario waarvoor een optimalisatie opdracht is uitgevoerd - De gebruiker kiest een uitgevoerde optimalisatie opdracht - De gebruiker kan hierbij een overzicht krijgen van de optimale waarden voor alle vrije parameters (inclusief de oorspronkelijke bereiken) - de gebruiker kan de optimale parameters opslaan als parameter set voor een volgende simulatie.
Rollen	Eindgebruiker
Pre-condities	De optimalisatie voor het scenario is correct uitgevoerd
Post-condities	De resultaten zijn na enige tijd beschikbaar als nieuwe parameter instelling voor hetzelfde scenario.

2.2.3 Use cases van de beheerder

UC B01	Inzicht in gebruik van het systeem
Omschrijving	De beheerder kan inzicht verkrijgen in het gebruik van het systeem, zoals het aantal machines dat opdrachten aan het verwerken is en de verwerkingssnelheid van deze
Stappenplan	- De beheerder geeft aan dat hij inzicht in het gebruik wil krijgen - Het systeem geeft een overzicht van: - Het aantal machines dat actief is - Wat elke machine aan het doen is - Statistieken over het gebruik van elke machine - Welke opdrachten nog in een wachtrij staan - etc.
Rollen	Beheerder
Pre-condities	Er is tenminste een machine beschikbaar om opdrachten uit te voeren
Post-condities	Geen

UC B02	Toewijzen van te gebruiken machines
Omschrijving	De beheerder kan machines toevoegen of verwijderen, welke gebruikt mogen worden door het systeem om opdrachten op uit te voeren
Stappenplan	- De beheerder geeft de hostname van een machine op, welke het systeem kan gebruiken - Het systeem voegt deze toe aan de lijst
Rollen	Beheerder
Pre-condities	Geen
Post-condities	Er is tenminste een machine beschikbaar om opdrachten uit te voeren De nieuw toegevoegde machine zal werk krijgen van het systeem wanneer er een opdracht beschikbaar is

UC B03	Aanpassen van instellingen
Omschrijving	De beheerder kan instellingen van het systeem aanpassen
Stappenplan	- De beheerder verandert een van de volgende instellingen: - Welke gebruikers toegang hebben tot het systeem - etc.
Rollen	Beheerder
Pre-condities	Geen
Post-condities	Het systeem past zich aan, aan de hand van de nieuwe instellingen

2.2.4 Use cases van de modelontwikkelaar

UC M01	Toevoegen of wijzigen van een model aan verzameling
Omschrijving	De modelontwikkelaar kan bestaande modellen wijzigen of nieuwe modellen toevoegen aan de modelverzameling
Stappenplan	- De modelontwikkelaar zet het (nieuwe of gewijzigde) model klaar - Hij geeft deze een betekenisvolle naam - Hij geeft het model vrij voor gebruik.
Rollen	Modelontwikkelaar
Pre-condities	Er is een correcte beschrijving van het model in de gehanteerde simulatietaal.
Post-condities	Het model is opgenomen in de modelverzameling

2.3 Non-functionele eisen

Alle non-functionele eisen zullen beschreven worden aan de hand van het zogenaamde Quint³-model. Daarvan is hieronder het overzicht:

2.3.1 Functionality

Suitability	Voor de eerste versie is het van belang dat tenminste 5 van de 15 use cases geïmplementeerd zijn, te weten G01, G02, G02a, G03 en G04. In de daarop volgende versie moeten use cases G02b, G05, G05a en G05b geïmplementeerd zijn.
Accuracy	Voor alle getallen die we terug krijgen, verwachten we tenminste een nauwkeurigheid van 3 cijfers achter de komma.
Interoperability	Aangezien het systeem vooralsnog intern zal werken (geen koppeling met andere systemen), definiëren we hiervoor nog geen eisen.
Compliance	We kiezen zoveel mogelijk voor standaard data formaten om communicatie tussen clients en server en clients onderling zo soepel mogelijk te laten verlopen (denk aan JSON, SOAP). Voor de beschrijving van de modellen gebruiken we OpenModellica.
Security	Aangezien het systeem vooralsnog intern zal werken, definiëren we hiervoor nog geen eisen.
Traceability	Het is belangrijk om te kunnen achterhalen of de aangeboden scenario's ook daadwerkelijk en correct zijn uitgevoerd. Daartoe eisen we dat de simulatie omgeving (cloudmanager en workers) via logging bijhouden wie wat heeft uitgevoerd.

2.3.2 Reliability

Maturity	Vooralsnog bouwen we een demonstrator, daarbij mag er af en toe wel eens iets misgaan, waarbij een herstart nodig is. De MTBF moet minimaal een uur zijn (zodat we zonder problemen een demo kunnen geven).
Fault tolerance	Bij een fout is het acceptabel dat het systeem down gaat en er een herstart nodig is.
Recoverability	Het systeem moet binnen 5 minuten weer gerecoverd kunnen worden om een demo te kunnen geven.
Availability	De beschikbaarheid van de processing (verwerking) hoeft niet heel hoog te zijn. Maar de beschikbaarheid van de data moet wel gegarandeerd zijn als het systeem on line is.
Degradability	Een minimale set beschikbaar krijgen is niet belangrijk: het systeem doet het wel of niet (via de eis aan recoverability is dat binnen 5 minuten).

³ Het Quint-model is een uitbreiding om de ISO 9126 standaard. De eerste versie is geschreven door CIBIT Adviseurs in 1996, in een document genaamd "Kwaliteit van softwareproducten".

2.3.3 *Usability*

Understandability	Met minimale uitleg door degene die de demonstratie geeft, moet de demo te begrijpen zijn. Een leek hoeft niet zelf de demonstrator te kunnen geven en te begrijpen.
Learnability	Hier stellen we geen eisen aan.
Operability	Hier stellen we geen eisen aan.
Explicitness	Hier stellen we geen eisen aan.
Customisability	Hier stellen we geen eisen aan.
Attractivity	De demonstratie moet aantrekkelijk zijn om naar te kijken.
Clarity	Hier stellen we geen eisen aan.
Helpfulness	Hier stellen we geen eisen aan.
User-friendliness	De simulator moet voor ingewijden (zoals Flexigas projectleden) te bedienen zijn.

2.3.4 *Efficiency*

Time behaviour	Hier stellen we geen harde eisen aan, al zal het systeem wel zo snel mogelijk vele scenario's kunnen uitvoeren. (Denk aan minimaal 1000 scenario's binnen 10 minuten.)
Resource behaviour	Het systeem mag alle resources die hij tot zijn beschikking heeft gebruiken om de time behaviour zo optimaal mogelijk te krijgen.

2.3.5 *Maintainability*

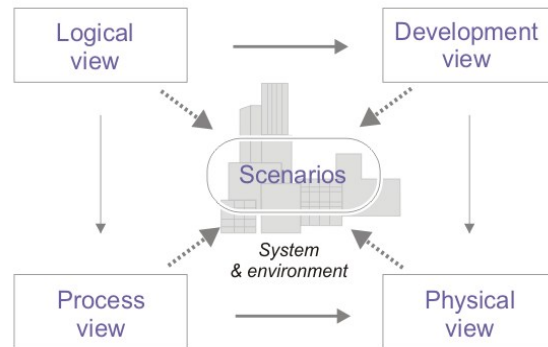
Analysability	Wanneer een fout zich voordoet, moet het mogelijk zijn om binnen 10 minuten er achter te komen waarom deze fout zich voor heeft gedaan.
Changeability	Wanneer een fout in de software geïdentificeerd is, moet het mogelijk zijn om binnen 60 minuten de fout te herstellen en opnieuw het systeem te starten. Uitzondering hierop zijn fouten in het ontwerp.
Stability	Per nieuwe revisie van de software mag er maximaal 1 nieuwe niet-kritische bug geïntroduceerd worden.
Testability	De verschillende onderdelen moeten los van elkaar te testen zijn. Hierbij mag maximaal dezelfde hoeveelheid tijd zitten in het maken van de tests als in het maken van de software zelf.
Manageability	Zoals eerder omschreven, moet het voor een beheerder mogelijk zijn om het systeem binnen 5 minuten te resetten.
Reusability	Dit systeem hoeft niet hergebruikt te worden voor andere systemen, echter moet er wel zo efficiënt mogelijk worden omgegaan om hergebruik binnen het systeem te bevorderen.

2.3.6 *Portability*

Adaptability	Hier stellen we geen eisen aan.
Installability	Hier stellen we geen eisen aan.
Conformance	Hier stellen we geen eisen aan.
Replaceability	Hier stellen we geen eisen aan.

3 High-level ontwerp

Bij het high-level ontwerp maken we gebruik van het "4+1 view model" ontworpen door Philippe Kruchten in 1995⁴ voor software-intensieve systemen, gebaseerd op het idee van meerdere concurrerende aanzichten.



Figuur 4: Principe van de 4+1 view

De views worden gebruikt om het systeem te beschrijven vanuit gezichtspunt van de verschillende belanghebbenden, zoals eindgebruikers, ontwikkelaars en projectmanagers. De vier gezichtspunten van het model zijn het: logische -, ontwikkelings-, proces- en fysieke gezichtspunt. De additionele geselecteerde usecases of scenario's worden gebruikt om de architectuur te illustreren als het "+1" gezichtspunt.

In de volgende paragrafen zullen deze 5 views kort en algemeen worden toegelicht (met verwijzingen naar Wikipedia) en waar relevant wordt voor de simulator specifiek ingevuld. In de eerste paragraaf zullen we een logisch aanzicht beschrijven, wat zowel inzicht geeft in het datamodel als in de volgorde waarin de verschillende componenten elkaar zullen gaan aanroepen. Daarna zal de tweede paragraaf op hoog niveau het proces omschrijven om van een scenario-omschrijving te komen tot inzicht in resultaten. De gedachte van de "Development View" wordt kort toegelicht in paragraaf 3 maar is niet concreet ingevuld omdat het daarbij op te stellen package diagram in dit geval geen toegevoegde waarde heeft (bij complexere projecten is dat wel het geval). Paragraaf vier omschrijft de fysieke aanblik, waarin de verschillende fysieke systemen staan omschreven met hun rollen en componenten die daarop staan. Aan de scenario-view van paragraaf vijf is al eerder invulling gegeven door middel van de de use-cases uit hoofdstuk 2.

3.1 Logische gezichtspunt

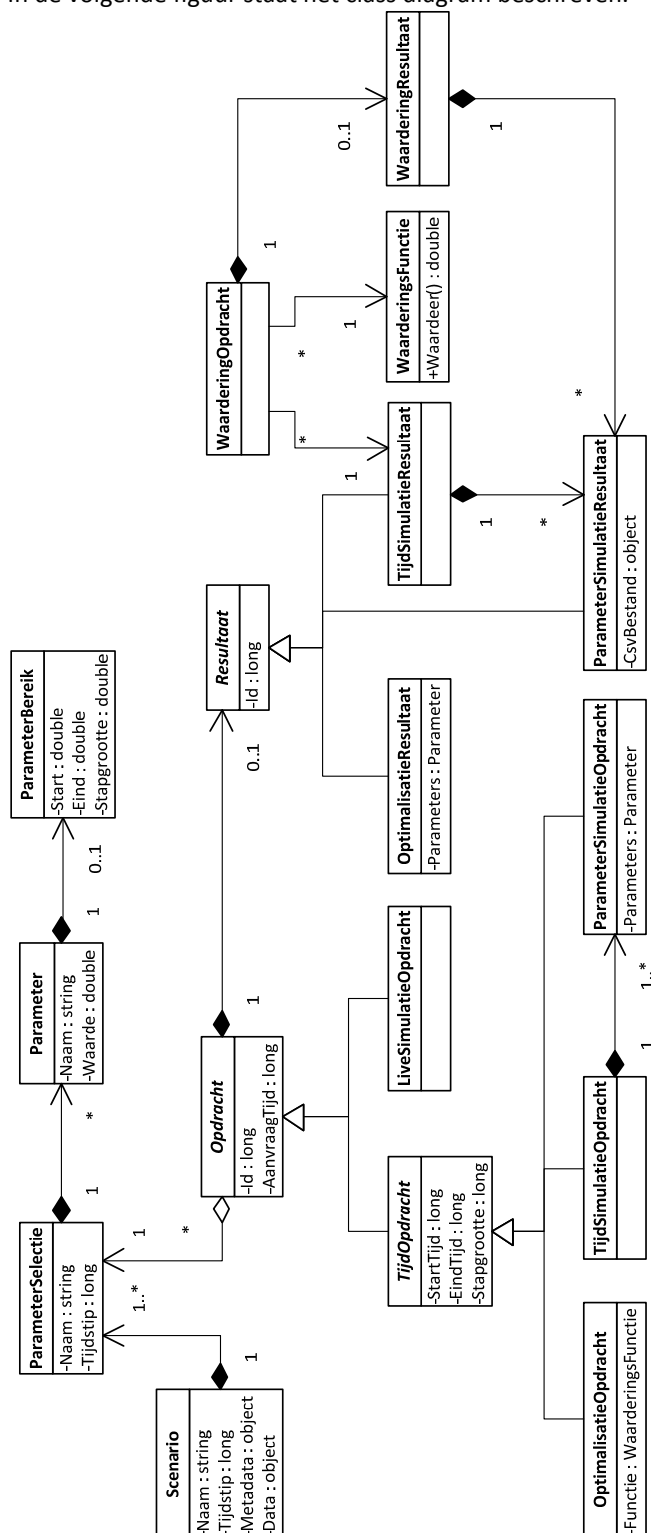
In het logische gezichtspunt worden de functionaliteit beschreven die het systeem de eindgebruikers biedt. Voor het vastleggen van dit gezichtspunt in UML wordt gebruik gemaakt van het Class diagram en het Sequence diagram. Zie voor een uitgebreidere beschrijving:

- http://en.wikipedia.org/wiki/Class_diagram
- http://en.wikipedia.org/wiki/Sequence_diagram

⁴Zie <http://www.cs.ubc.ca/~gregor/teaching/papers/4+1view-architecture.pdf> en <http://en.wikipedia.org/wiki/4%2B1>

Class diagram

In de volgende figuur staat het class diagram beschreven.

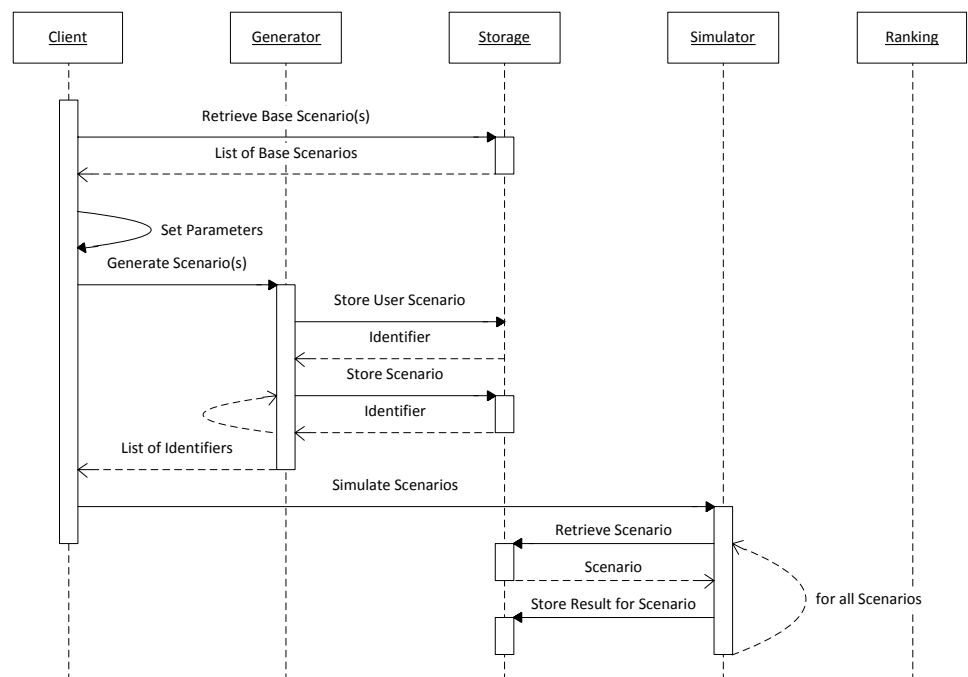


Figuur 5 Class diagram: Datamodel

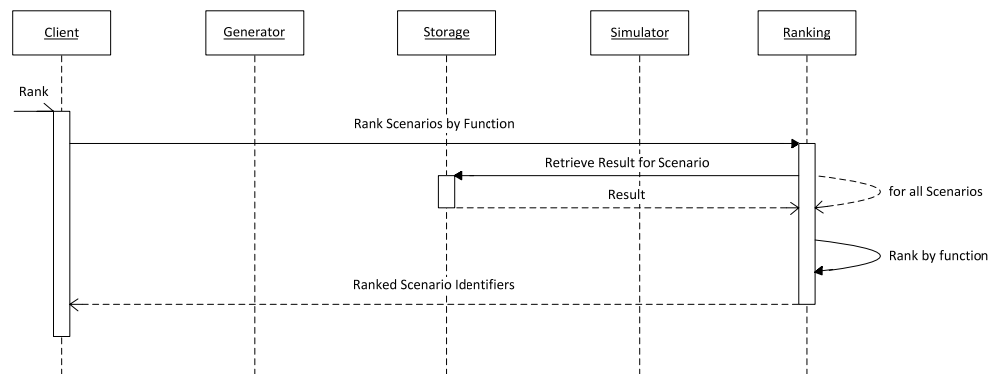
Sequence diagrammen

Voor de simulator van het smart biogas grid zijn de volgende sequence diagrammen opgesteld:

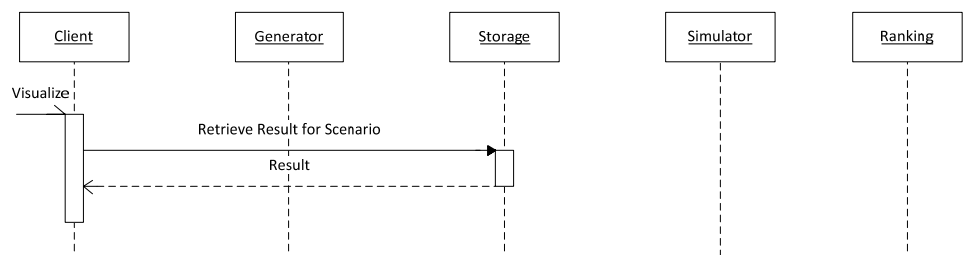
- Simulate: deze beschrijft hoe de client applicatie een scenario kan laten simuleren.
- Rank: deze beschrijft hoe de client applicatie de resultaten van een aantal scenario's kan rangschikken.
- Visualize: deze beschrijft hoe de client applicatie de resultaten kan ophalen voor de visualisatie.
- Live: deze beschrijft hoe de client applicatie de live resultaten kan ophalen voor de visualisatie.
- Optimize: deze beschrijft hoe de client applicatie een opdracht tot optimalisatie kan aanvragen



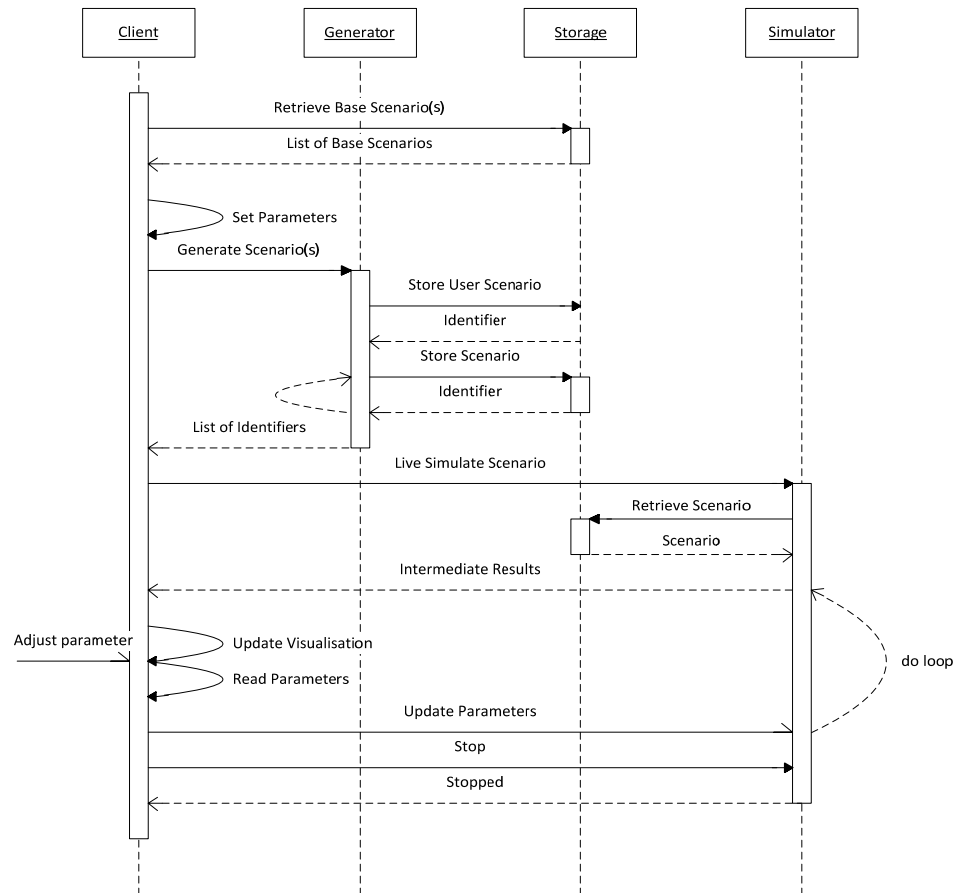
Figuur 6 Sequence diagram "Simulate"



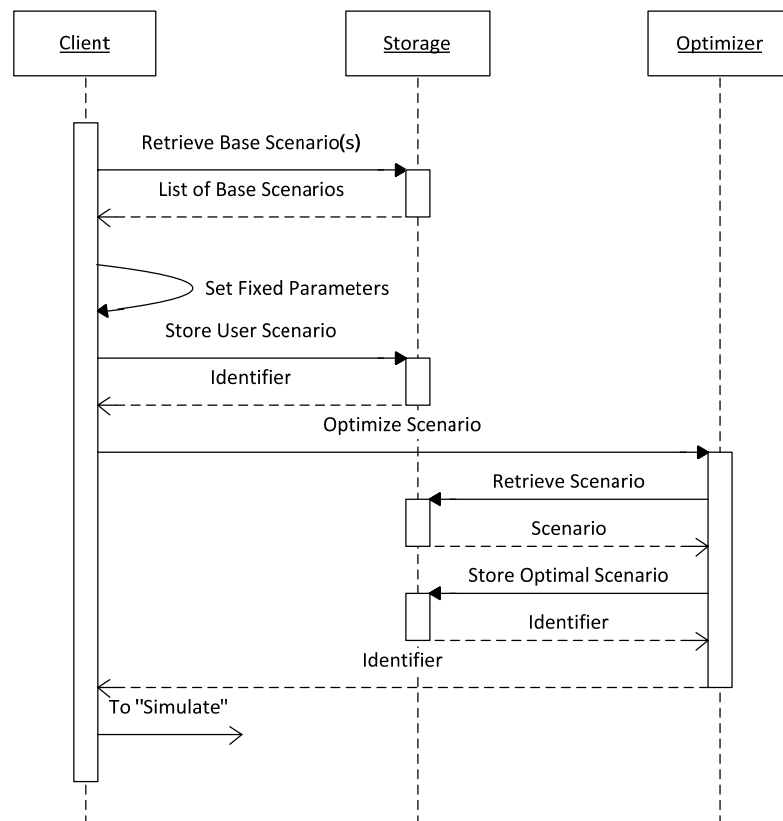
Figuur 7 Sequence diagram "Rank"



Figuur 8 Sequence diagram "Visualize"



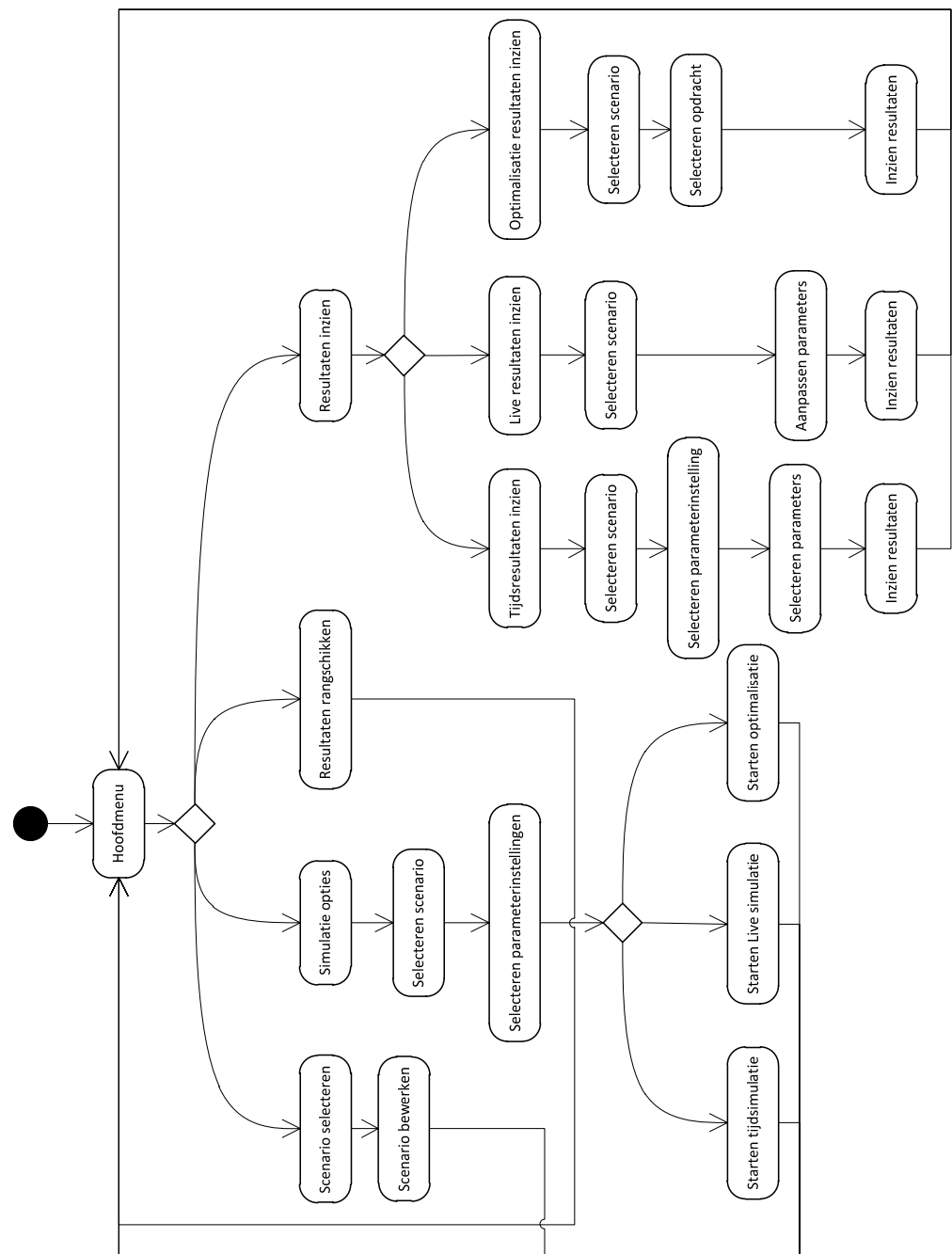
Figuur 9 Sequence diagram "Live"



Figuur 10 Sequence diagram "Optimize"

3.2 Proces gezichtspunt

Dit gezichtspunt omvat de dynamische aspecten van het system, de processen en hoe ze communiceren en focussed op het runtime gedrag van het system. In feite hoe de functionaliteit in werkelijkheid gebruikt wordt bij een echte run. Het gaat in op distributie, concurrency, integrators, performance, en schaalbaarheid. Voor de weergave van het gezichtspunt wordt gebruik gemaakt van het Activity diagram. Zie http://en.wikipedia.org/wiki/Activity_diagram voor een uitgebreidere beschrijving.



Figuur 11 Activity diagram

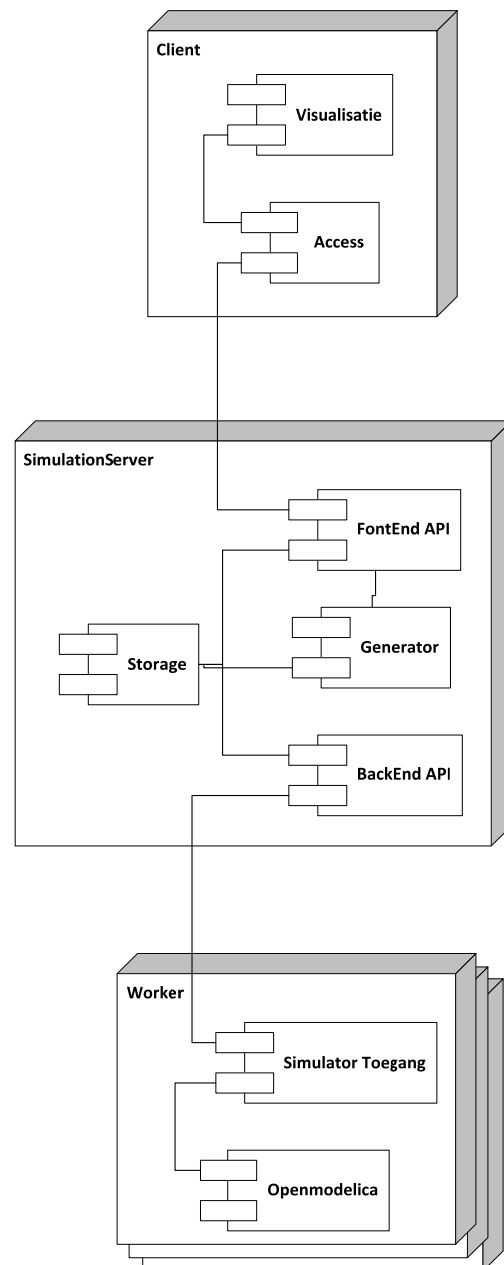
3.3 Ontwikkel gezichtspunt

Dit betreft het gezichtspunt vanuit de programmeur en betreft software management. Een en ander wordt vastgelegd in het package diagram. Zie http://en.wikipedia.org/wiki/Package_diagram voor een uitgebreidere beschrijving.

Deze view wordt hier niet concreet ingevuld omdat deze view niets toevoegt aan dit document (bij complexere projecten is dat wel het geval).

3.4 Fysieke gezichtspunt

Dit is de beschrijving van het system vanuit het gezichtspunt van de system engineer. Het bevat de topologie van software componenten op de fysieke laag alsook de fysieke verbindingen tussen deze componenten. Dit wordt vastgelegd in component- en deployment diagram (zie http://en.wikipedia.org/wiki/Component_diagram en http://en.wikipedia.org/wiki/Deployment_diagram voor een uitgebreidere beschrijving). Voor de simulator krijgen we het volgende.



Figuur 12: Fysieke gezichtspunt

Client

Het Client-component is het component waarmee de eindgebruiker toegang heeft tot de simulatie omgeving. Voor de interactie wordt een Visualisatie-component gemaakt. Voor het doorgeven van bijvoorbeeld parameter wijzigingen of ophalen simulatie resultaten maakt het Visualisatie component gebruik van het Acces-component.

SimulationServer

De SimulationServer-component bevindt zich als “ontkoppel laag” tussen de Client (voor interactie met de gebruiker) en de Worker (waar de werkelijke simulaties gaan plaats vinden).

Voor interactie met de omgeving wordt gebruik gemaakt van een *FrontEnd API* en een *BackEnd API* voor communicatie met respectievelijk de Client en de Worker.

Via *FrontEnd API* kan de *Generator* aangesproken worden, is er toegang tot de *Storage* en kan via de *Backend API* de Worker worden benaderd.

Storage wordt gebruikt voor de opslag van templates, modellen, scenario's, parameters, opdrachten en resultaten.

De *Generator* genereert, op basis van door de gebruiker opgegeven parameter instellingen, de uit te voeren simulaties. Via de *Backend API* worden de simulatie settings naar de Worker gestuurd.

Worker

Binnen de Worker-component worden de daadwerkelijke simulaties uitgevoerd. Aanleveren van simulatie instellingen vindt plaats via de *Simulator Toegang*. Die stuurt het door naar de uiteindelijke *Simulator*. Hier wordt vooralsnog uitgegaan van OpenModelica. Maar door de gekozen opbouw kan deze “simulatie engine” relatief eenvoudig vervangen worden door een andere omgeving, waarbij de Client-component en SimulationServer-component niet of nauwelijks aangepast hoeven te worden.

3.5 Scenario's

Voor de beschrijving van de architectuur wordt gebruik gemaakt van een beperkte set use cases (of scenario's) die het vijfde gezichtspunt vormen. De scenario's beschrijven een sequentie van interacties tussen objecten en tussen processen. Ze worden gebruikt om architectuur elementen te identificeren en het architectuur ontwerp te illustreren en valideren. Ze dienen ook als vertrekpunt voor het testen van een prototype-architectuur. Use-case diagrammen worden gebruikt voor de weergave. Zie http://en.wikipedia.org/wiki/Use_case_diagram voor een uitgebreidere beschrijving.

De scenario's zijn uitgebreid beschreven middels de use cases van hoofdstuk 2. In bijlage A wordt ter illustratie een zogenaamd “paper prototype” getoond, gebaseerd op de use cases.

4 Mogelijke toekomstige uitbreidingen

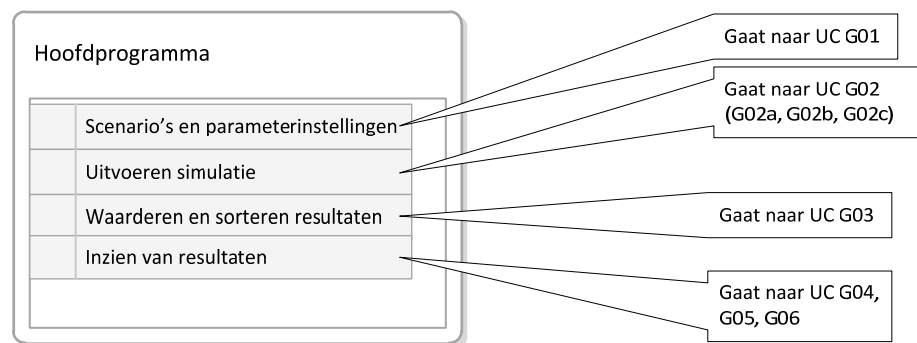
In onderstaande lijst staan mogelijke uitbreidingen van dit architectuur document die in een volgende versie uitgewerkt kunnen worden.

- De relatie tussen de definities uit de woordenlijst (zie 2.1) in een figuur helder toelichten.
- Toevoegen Use Case voor het maken van een template.
- Bewaren van visualisatie instellingen (bijvoorbeeld geselecteerde parameters voor grafieken).
- Inzicht in correlatie tussen simulatie resultaten.
- Uitbreiden van de Usability eis “Customisability” (zie 2.3.3) tav het kunnen maken en aanpassen van templates en andere generieke componenten .
- Uitbreiden van de Functionality eis “Interoperability” (zie 2.3.1) tav de koppeling met de database van BioBench.
- Ook de koppeling met die BioBench database beschrijven.
- Ook nog externe input in het algemeen meenemen.
- Uitbreiden van het document met technische (implementatie) keuzen tav database, machines, verbindingen, simulator, cloud, etc.

5 Bijlage A: Paper-prototype

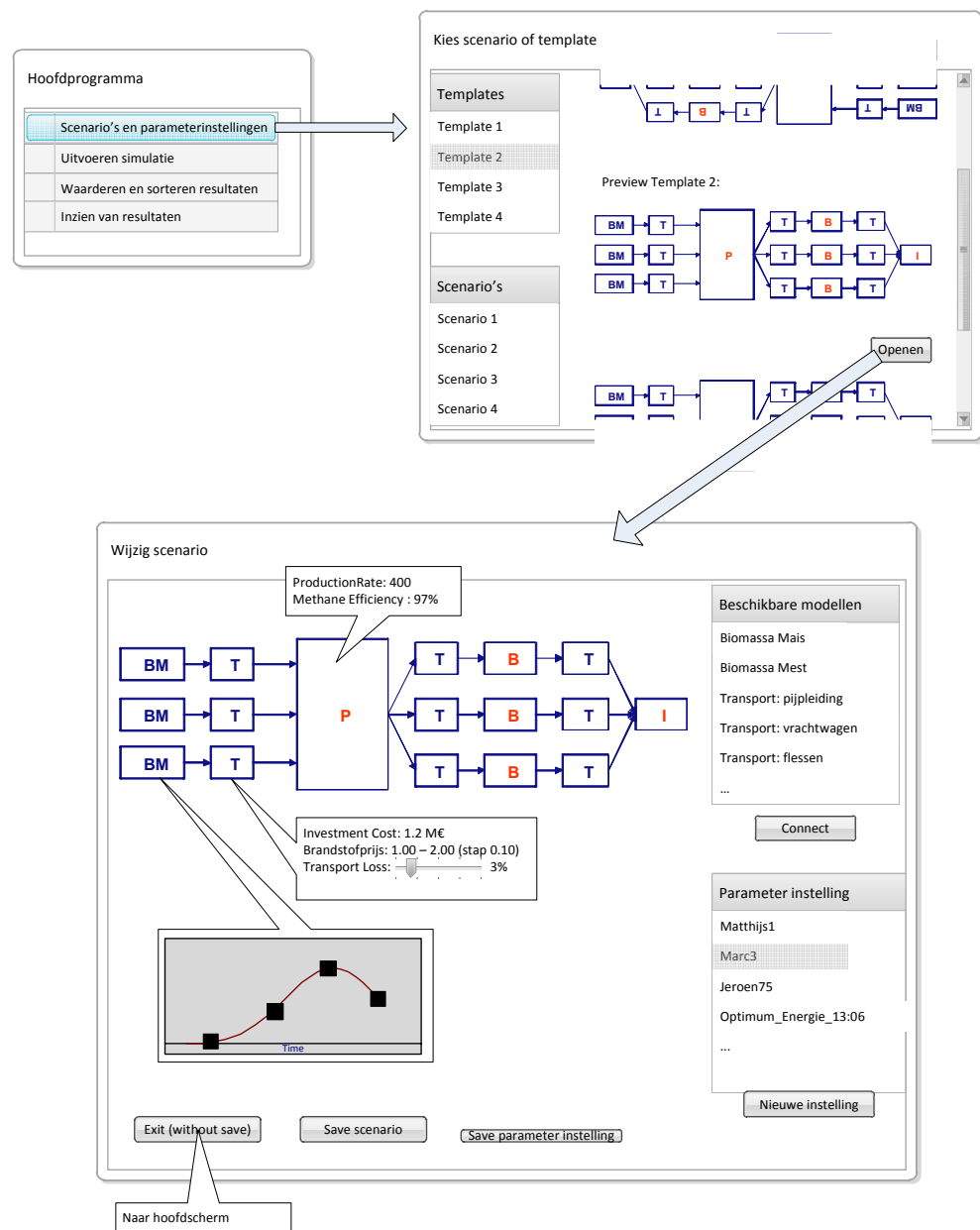
In deze bijlage wordt ter illustratie een zogenaamd “paper prototype” getoond. Dit zijn getekende schermen die de simulator straks zou kunnen laten zien. Ze zijn gebruikt bij het uitwerken van de use-cases voor de eindgebruiker en geven de lezer een beter beeld. De schermen in de uiteindelijke versie zullen zeker afwijken van de schermen zoals die hier getoond worden.

Hoofdmenu



Figuur 13: Hoofdmenu

UC G01 Aanmaken of aanpassen van een scenario



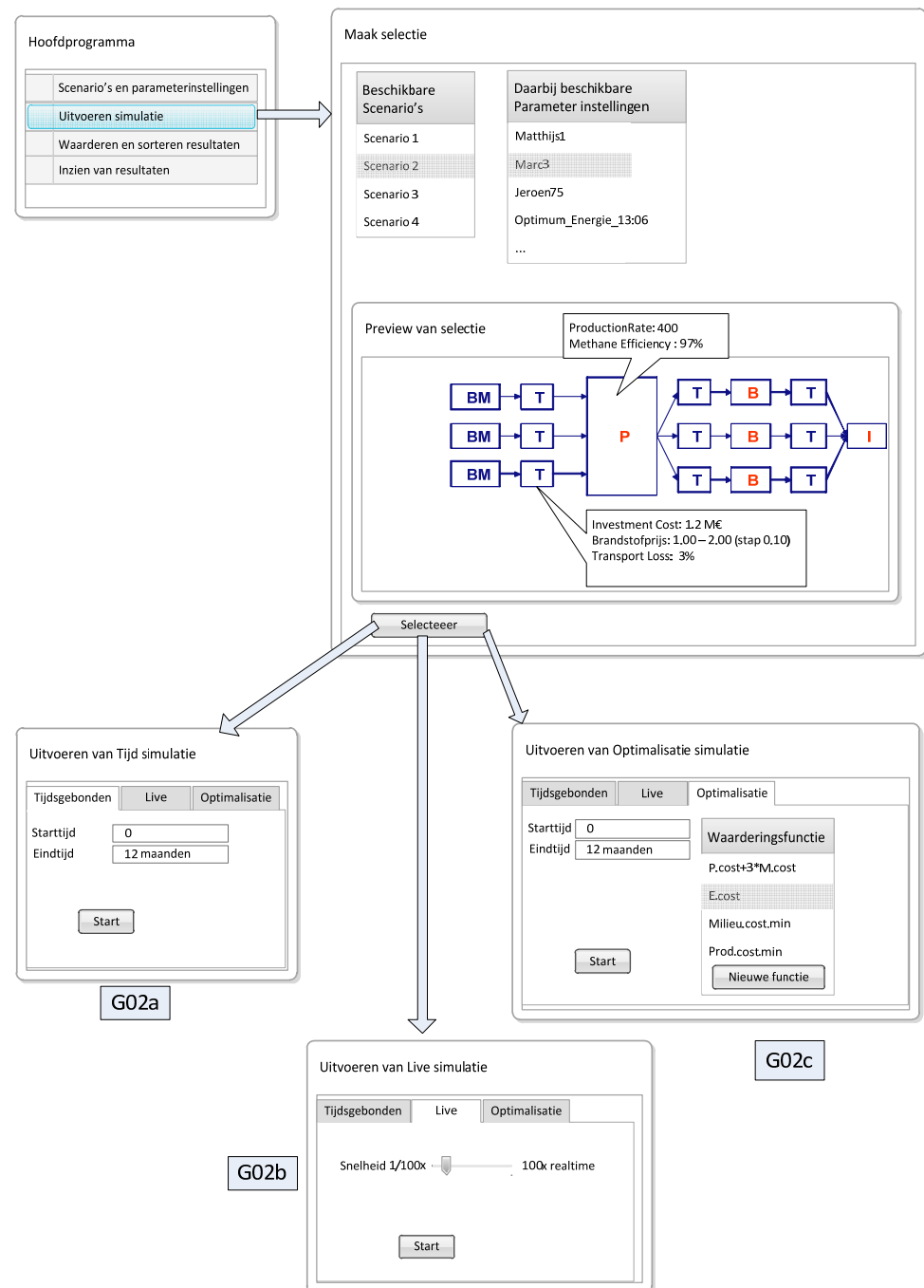
Figuur 14 Aanmaken of aanpassen van een scenario

UC G02 Uitvoeren van een simulatie

UC G02a Uitvoeren van tijd gelimiteerde simulatie

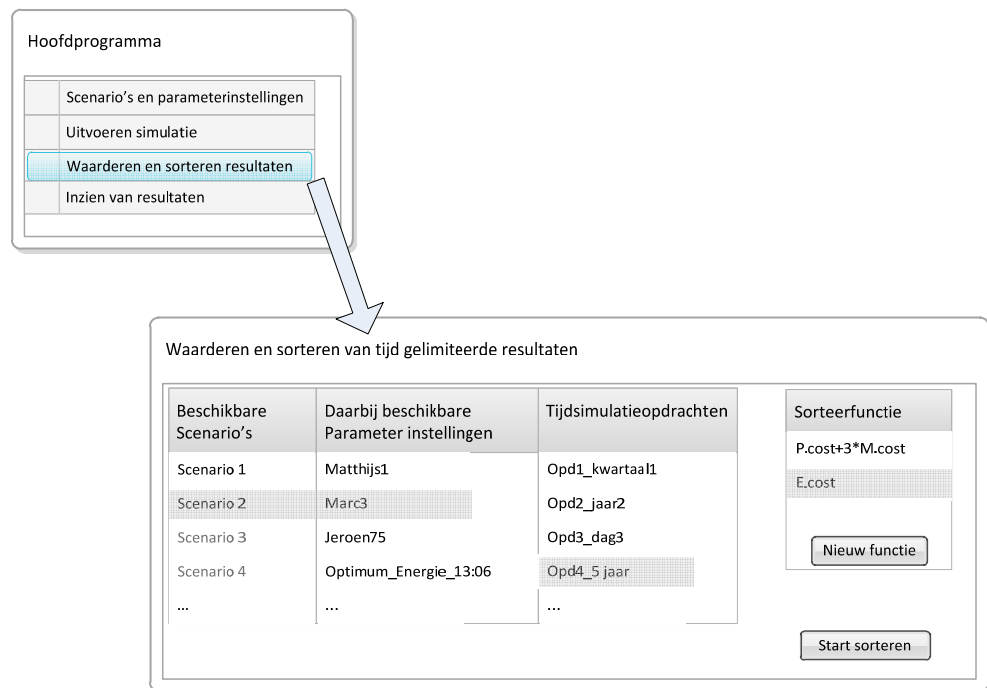
UC G02b Uitvoeren van live-simulatie

UC G02c Uitvoeren van optimalisatie

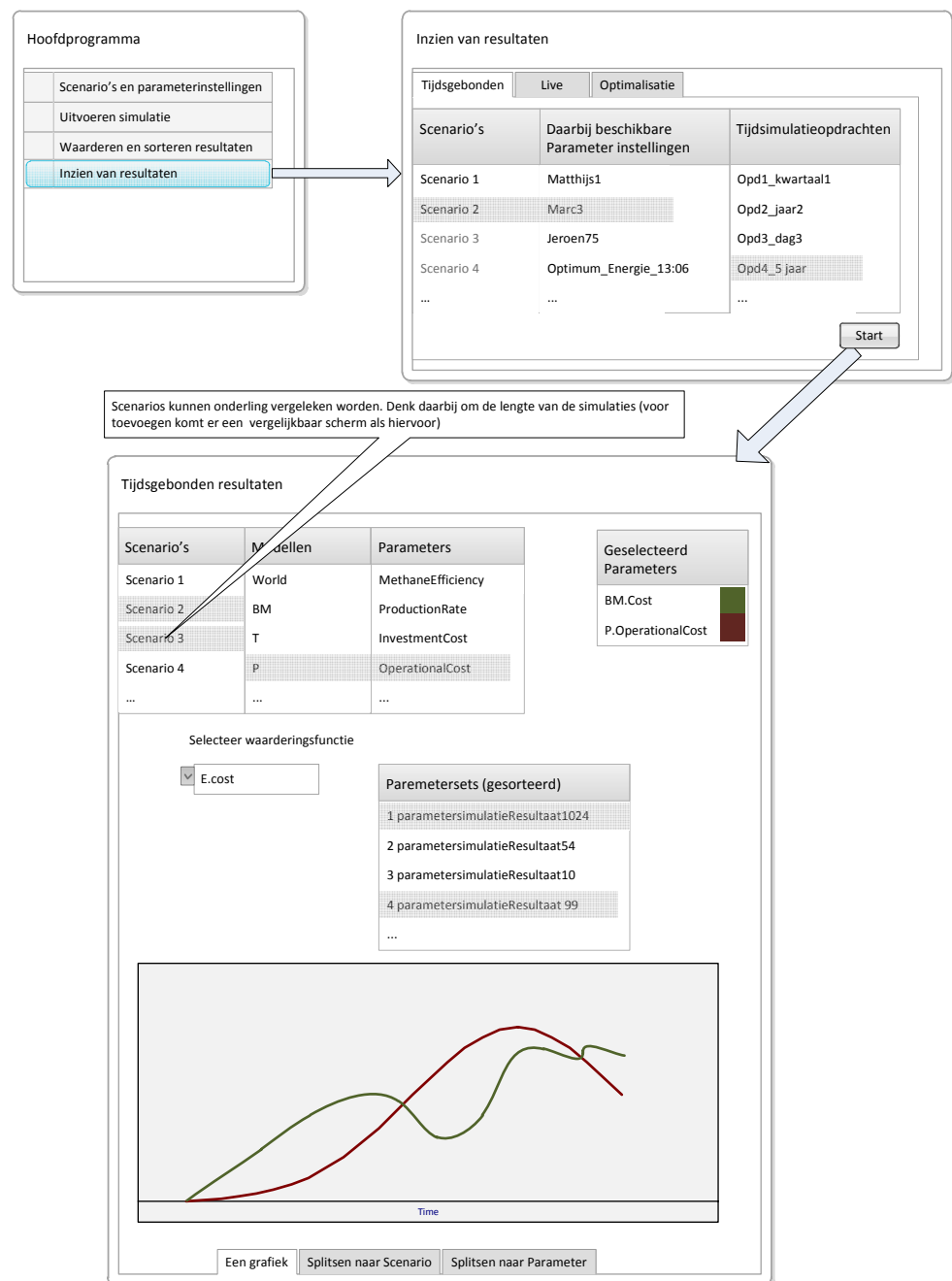


Figuur 15 Uitvoeren van een simulatie

UC G03 Waarderen en sorteren van tijd gelimiteerde resultaten



Figuur 16 Waarderen en sorteren van tijd gelimiteerde resultaten

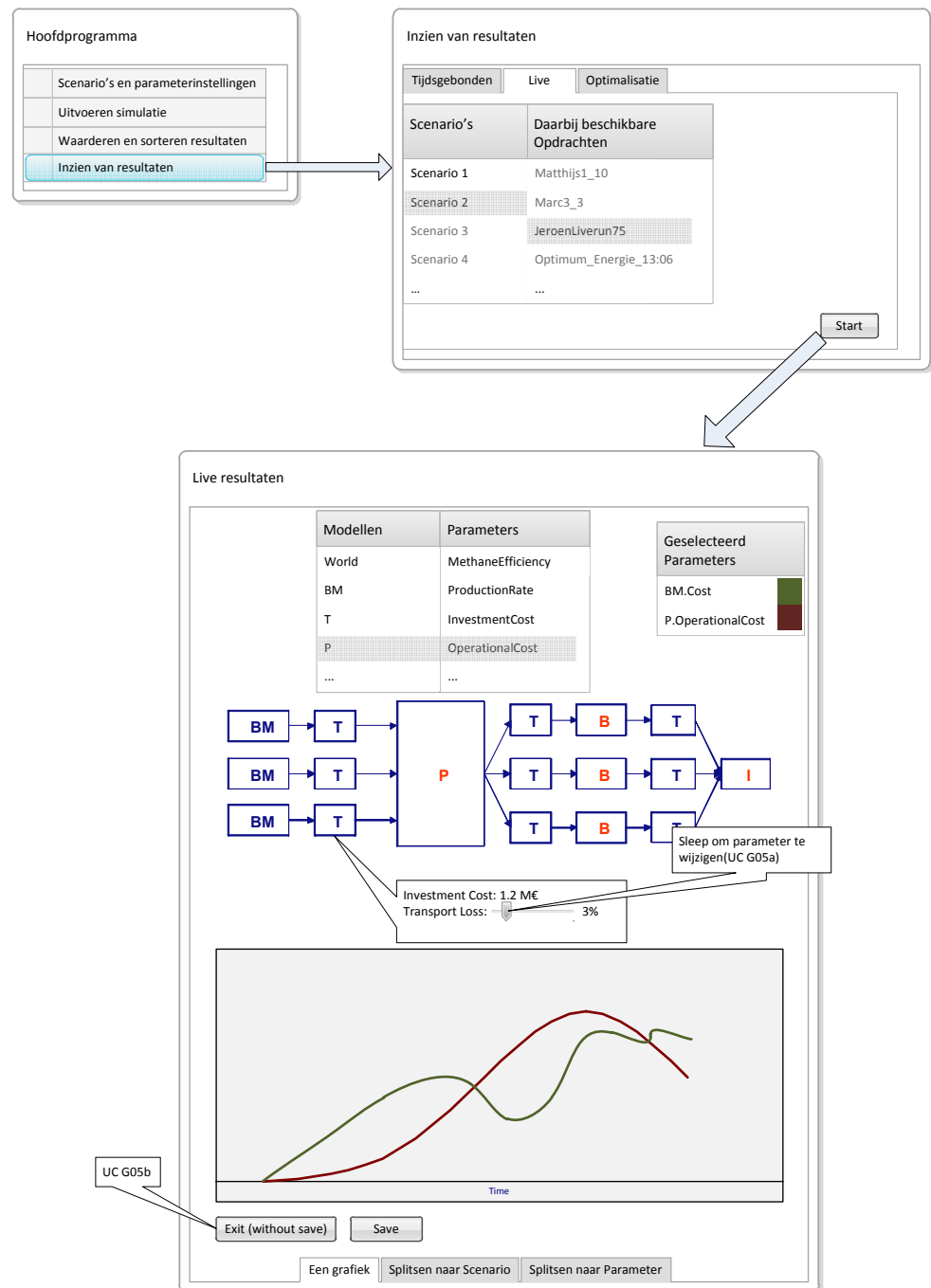
UC G04 Inzien van tijd gelimiteerde resultaten

Figuur 17 Inzien van tijd gelimiteerde resultaten

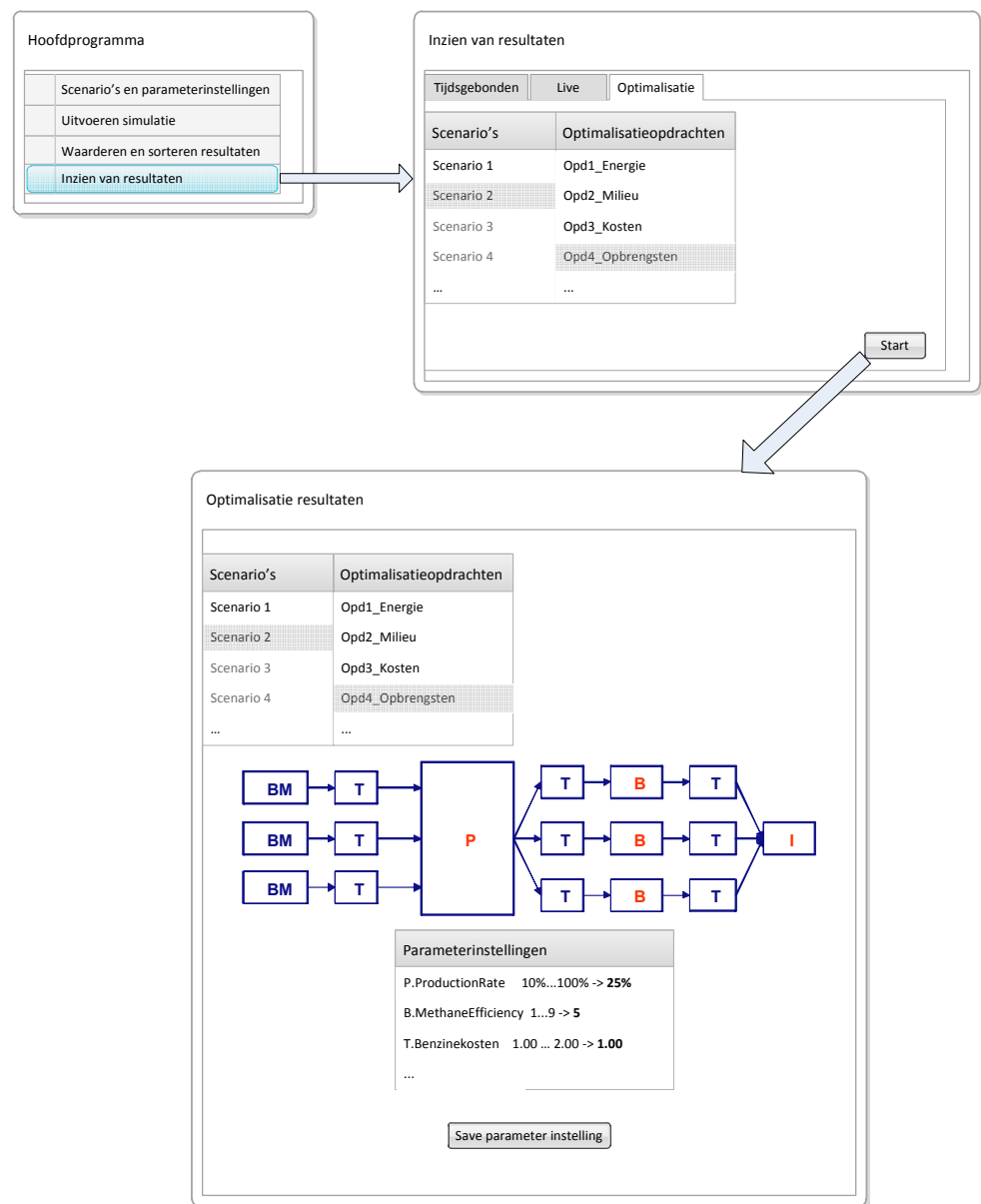
UC G05 Inzien van live resultaten

UC G05a Live aanpassen van parameters

UC G05b Stoppen van de live-simulatie



Figuur 18 Inzien van live resultaten, aanpassen en stoppen

UC G06 Inzicht in optimalisatie resultaten

Figuur 19 Inzicht in optimalisatie resultaten