



## Service Supply Chain Game: deliverable 2

Onderwerp: Dit document bevat de deliverable van fase 2 van de service supply chain game (conceptueel ontwerp).  
Datum: 29 november 2016  
Aan: Herman Wagter  
Van: Maarten Driessen, Joost Florie, Jos Thalen  
CC: Jasper de Graaf, Jan Willem Rustenburg, Jürgen Donders

---

### 1 Inleiding

In opdracht van Connekt ontwikkelt Gordian, samen met Invoke en Joost Florie (NedTrain), een service supply chain game. Onderstaande tabel schetst de activiteiten en deliverables van fase 2. Deze fase is afgerond en de deliverable van deze fase is te vinden in paragraaf 2 en 3 van dit document.

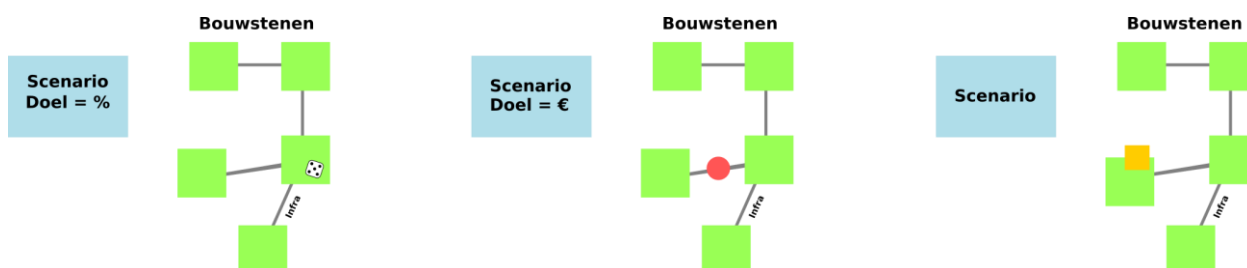
Onderwerp	Activiteiten	Deliverable(s)
2. Conceptueel ontwerp	<ul style="list-style-type: none"><li>▪ Selecteren geschikte voorwaarden en wensen om op te nemen in het spelontwerp</li><li>▪ Optimaliseren spelmechanieken</li><li>▪ Vaststellen spelregels &amp; vormgeving</li></ul>	<ul style="list-style-type: none"><li>▪ Eerste ontwerp van het bordspel en applicatie</li></ul>

## 2 Conceptueel ontwerp

### 2.1 Van gewenste interventies naar spelelementen

In fase 1 leverden we een long list op van interventies die we met de service supply chain game willen simuleren. We gebruikten deze interventies (innovatieve onderhoudsconcepten, technieken, etc.) als uitgangspunt voor de uitwerking van bouwstenen, infrastructuur en scenario's. Per interventie bekeken we hoe we de effecten van de interventie willen modelleren, zoals:

- Toevoegen van bouwstenen
- Toevoegen/veranderen van eigenschappen/karakteristieken aan bestaande bouwstenen
- Toevoegen/veranderen van eigenschappen aan de infrastructuur



Afbeelding 1: Voorbeeld vertaling van interventies naar bouwsteeneigenschap (links), infrastructuur (midden) en een nieuwe bouwsteen (rechts).

Paragraaf 2.2 beschrijft de eigenschappen die we aan het spel willen toevoegen teneinde alle interventies met het spel te kunnen simuleren.

### 2.2 Eigenschappen spelelementen

Onderstaande tabel bevat de variaties in eigenschappen van de spelelementen die we opnemen in het spel.

Categorie	Spelelement	Eigenschappen
1.	Bouwstenen	<ul style="list-style-type: none"> <li>▪ Eén vs. meerdere locaties</li> <li>▪ Eén vs. meerdere assettypen</li> <li>▪ Variaties in aantal assets</li> <li>▪ Vaste doorlooptijd onderhoud vs. werkelijke doorlooptijd o.b.v. beschikbaarheid parts en monteurs</li> <li>▪ Inzicht in en digitale registratie status van ieder asset (bv. in gebruik, in onderhoud, wacht op parts)</li> <li>▪ Variaties in onderhoudsconcept (run to failure, preventief vervangen, conditiegestuurd vervangen)</li> <li>▪ Variaties in doelstelling gewenste beschikbaarheid assets</li> <li>▪ Variaties in onderhoudsplanning mogelijk</li> </ul>
	Centrale/lokale voorraadlocatie	<ul style="list-style-type: none"> <li>▪ Eén vs. meerdere locaties</li> <li>▪ (On)beperkte opslagcapaciteit</li> <li>▪ Opslag vuile en/of schone delen</li> <li>▪ Inzicht in en digitale registratie van voorraad(waarde)</li> </ul>



		Componentenfabriek	<ul style="list-style-type: none"> <li>▪ Eén vs. meerdere componentenfabrieken</li> <li>▪ Reguliere reparatie vs. spoedreparaties</li> <li>▪ Vaste doorlooptijd vs. werkelijke doorlooptijd o.b.v. beschikbaarheid monteurs en parts (behoefte bekend na inspectie).</li> <li>▪ Inzicht in en digitale registratie status van iedere reparatieorder (bv. wacht op vrijgave, in inspectie, in reparatie, wacht op parts)</li> <li>▪ Onderscheid verschillende processtappen mogelijk (vuil, inspectie, reparatie, schoon)</li> <li>▪ Variatie mogelijk in capaciteit (aantal monteurs)</li> <li>▪ Generieke vs. specialistische skills monteurs</li> <li>▪ Variaties in openingstijden</li> </ul>
		OEM / Leverancier	<ul style="list-style-type: none"> <li>▪ Eén vs. meerdere leveranciers</li> <li>▪ Vaste doorlooptijd vs. variabele doorlooptijd</li> <li>▪ Variaties in doorlooptijd (1,2,3,... beurten)</li> <li>▪ Onzekerheid (bv. obsolescence)</li> </ul>
		Spare parts	<ul style="list-style-type: none"> <li>▪ Eén vs. meerdere verschillende repareerbare delen</li> <li>▪ Schone vs. vuile delen</li> <li>▪ Variëren in omloopvoorraad</li> <li>▪ LRU's vs. SRU's</li> <li>▪ Generieke vs. facultatieve SRU's</li> <li>▪ Hoge vs. lage vraagonzekerheid</li> </ul>
<b>2.</b>	Infrastructuur	Transport	<ul style="list-style-type: none"> <li>▪ Heen- (schoon) en retourstroom (vuil) van parts</li> <li>▪ Reguliere vs. spoedtransporten</li> <li>▪ Laterale transporten</li> <li>▪ Vaste doorlooptijd vs. variabele doorlooptijd</li> <li>▪ Variaties in doorlooptijd (1,2,3,... beurten)</li> <li>▪ Minimale/maximale batchgrootte per transportbeweging</li> <li>▪ Onzekerheid uitval (bv. transportschade)</li> </ul>
<b>3.</b>	Scenario's	Speldoelen	<ul style="list-style-type: none"> <li>▪ Gericht op inzicht in interventies op kosten, beschikbaarheid of werkkapitaal</li> <li>▪ Generiek vs. detailniveau simulatie van een proces</li> </ul>
		Beslissingen	<ul style="list-style-type: none"> <li>▪ Afzonderlijke targets en beslissingen per bouwsteen vs. control tower (centrale aansturing)</li> <li>▪ Tijdens of tussen rondes</li> </ul>
		Gebeurtenissen	<ul style="list-style-type: none"> <li>▪ Dobbelstenen</li> <li>▪ Gebeurteniskaarten</li> <li>▪ Interventies tussen twee spelrondes vs. interventies op gezette tijdstippen</li> </ul>

Naast de eigenschappen van de spelelementen variëren we in het detailniveau waarop we iedere processtap afbeelden in het spel. Het detailniveau bepaalt de complexiteit van het spel. Wanneer het speldoel is om het effect van beslissingen in proces 1 te simuleren, dan is het verstandig om complexiteit van de overige processtappen zo laag mogelijk te houden. Op die manier houden we het spel simpel en begrijpelijk voor alle spelers.



### 3 Conceptueel ontwerp

Onderstaande foto's geven een eerste beeld van het conceptueel ontwerp:

