

SAMENVATTING BUSINESS ANALYSE

Dit vak is onderverdeeld in twee delen: Part I Business procesmanagement en Part II Business analyse

BUSSINES ANALYSE

BA1. Introduction to business analyse

1. Wat is business analyse ?

"Bedrijfsanalyse is de praktijk van het mogelijk maken van **verandering** in een onderneming door het definiëren van **behoeften** en het aanbevelen van **oplossingen** die **waarde** leveren aan **belanghebbenden**."

→ Praktijkdomein dat focust op veranderingstrajecten binnen bedrijven.

Bedrijven hebben noden die oplossingen vereisen en die veranderingen in het manier van werken met zich meebrengen.

Business analysten = Team die gaat nadenken over die nood en voorstel tot oplossing gaat doen en uiteindelijk ook die oplossing uitwerken

Vb.: Nood aan geïntegreerd communicatiesysteem naar de klanten toen die informatie aanbiedt zoals wanneer heb ik de klant het laatst gecontacteerd? Welk bericht? Wanneer laats iets gekocht? Hoe?... Deze informatie is belangrijk op operationele beslissingen te maken.

Proliferatie van IT-systemen (jaren 80): Boom bedrijven hebben nood aan digitale transformatie/ digitale intelligente oplossingen

→ Grote invloed op sterke groei van business analyse

Project successen:

CHAOS RESOLUTION BY PROJECT SIZE			
	SUCCESSFUL	CHALLENGED	FAILED
Grand	2%	7%	17%
Large	6%	17%	24%
Medium	9%	26%	31%
Moderate	21%	32%	17%
Small	62%	16%	11%
TOTAL	100%	100%	100%

The resolution of all software projects by size from FY2011-2015 within the new CHAOS database.

Hoe groter het project hoe groter de faalkansen. De projecten zijn vaak gefocust. Kleine projecten zijn succesvoller maar de waard toevoeging is wel kleiner.

Business Analyse Core Concept Model : BACCM

→ Business model

Basismodel dat geïntroduceerd wordt in een van de belangrijkste business analyse standaarden.



→ Karakteriseert de belangrijkste concepten van business analyse

BABOK (Business Analysis Body of Knowledge)

= Een verzameling concepten, activiteiten, te leveren prestaties, competenties, beginselen enz. die binnen een beroep als standaard wordt beschouwd.

- Identificeert momenteel geaccepteerde praktijken
- Erkent dat bedrijfsanalyse niet synoniem is met software-eisen
- Gedefinieerd en verbeterd door professionals die het toepassen
- Bevat de som van de kennis die nodig is voor de uitoefening van bedrijfsanalyse als beroep.
- BABOK is geen methodologie en schrijft ook geen methodologie voor.
- Het is geen "hoe te" handleiding voor bedrijfsanalyse

→ Enumeratie van alle technieken die binnen business praktijken gebruikt kunnen worden! Het is een samenvatting, het gaat dus niet diepgaand op de technieken.

BABOK Knowledge Areas: domeinen



2. Rol van een business analyst

Volgens het International Institute of Business Analysis (IIBA) werkt een bedrijfsanalist als: *tussenpersoon tussen belanghebbenden om eisen voor wijzigingen in bedrijfsprocessen, -beleid en -informatiesystemen te verzamelen, analyseren, communiceren en valideren.*

- ⇒ Het belangrijkste element is de zakelijke/ business focus; ervoor zorgen dat de zakelijke behoeften worden begrepen en gecommuniceerd, zodat de uiteindelijke oplossing voldoet aan de zakelijke behoeften.
- ⇒ Oplossingen kunnen IT-gerelateerd, niet-IT-gerelateerd of procesverbetering zijn. De bedrijfsanalist is verantwoordelijk voor het achterhalen van de werkelijke behoeften van belanghebbenden, niet alleen hun uitgesproken wensen, en speelt vaak een centrale rol bij het afstemmen van de behoeften van bedrijfsonderdelen op de mogelijkheden die de informatietechnologie biedt.

Verschillende types business analist:

- **Bedrijfsanalisten** met zeer sterke bedrijfsvaardigheden en inzicht in het bedrijfsdomein die als voornaamste taak hebben bedrijfsprocessen, procedures, enz. te analyseren om problemen te identificeren en oplossingen te bepalen. Deze analisten zijn meer betrokken bij wat het IIBA definieert als bedrijfsanalyse en zijn waarschijnlijk betrokken vóór de start van een IT-project.
 - Gaat zich focussen op de requirements

- **IT-businessanalist** die zich richt op het verzamelen en analyseren van eisen en het oplossen van problemen met behulp van IT-oplossingen. Deze analist dient als een brug tussen business en IT en begint over het algemeen te werken nadat een project is opgestart. Deze analist specificeert "wat" het systeem moet doen.
 - o Wat moet IT doen? Wat moeten er gebruikt worden?
- **Systeemanalist/ Functioneel analist** is een IT-bedrijfsanalist die zich meer richt op het systeemontwerp en de technische aspecten van de oplossing. Deze analist neemt de vereisten en creëert functionele specificaties betreffende "hoe" een systeem het "wat" zal doen;
 - o Hoe moet het systeem uitgewerkt worden? Hoe moet die werken?
- Vele andere titels worden gebruikt, waaronder de **businesssysteemanalist**, die is beschreven als een combinatie van de IT-businessanalist en de systeemanalist.

Rollen:

- Probleemoplosser
- Faciliteerder: positieve, voortdurende discussie en vooruitgang
- Onderhandelaar: bemiddelt tussen cliënten, belanghebbenden en andere betrokken partijen
- Architect: oplossingen die het volgende omvatten:
 - o Systeemontwikkeling
 - o Procesverbetering
 - o Organisatieverandering
 - o Modellen
 - o Bedrijfsprocessen
 - o Ontwerpen enzovoort
- Planner: Nauwkeurig geïdentificeerd, vastgelegd en gevolgd gedurende de gehele levenscyclus van het project; definiëren, organiseren, plannen, enz.
- Communicator: Discussies, gesprekken en interviews gebruiken om het begrip te vergroten en met anderen te communiceren.
- Strategisch: Lange termijn, visie, doelstellingen, tactische plannen
- Expert: Ken uw bedrijf, branche en domein

3. Competenties van een bedrijfsanalist

Competentie = het vermogen om een reeks verwante kennis, vaardigheden en bekwaamheden toe te passen of te gebruiken die nodig zijn om met succes "kritische werkfuncties" of taken in een bepaalde werkomgeving uit te voeren.

Belangrijkste competentiegebieden: *(Je kan niet expert zijn in alle gebieden)*

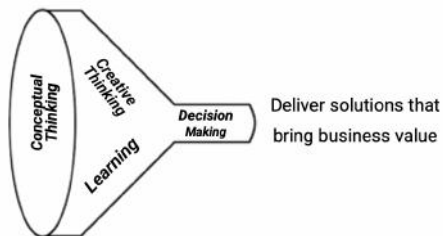
1. Zakelijke kennis:

- o Externe bedrijfskennis:
 - Zakelijk inzicht / Business Acumen: Kennis van en inzicht in oplossingen die eerder in andere organisaties zijn overwogen en/of toegepast.
 - Beste praktijken
 - Rapporten en studies
 - Casestudies
 - Gespecialiseerde aanbieders van oplossingen
 - Via ervaring
 - Industrie kennis: kennis over de sector
 - Belangrijke spelers en concurrenten
 - Huidige praktijken
 - Trends
 - Belangrijkste processen

- Branchespecifieke termen
- Regelgeving
- Informatietechnologie: Inzicht in informatietechnologie
 - Hoe ze werken
 - Ontwikkelingsprocessen
 - Methodologieën
 - Technologische vooruitgang – Trends
- Interne Bedrijfskennis:
 - Bedrijfsmodel en strategieën: de manier waarop een bedrijf waarde creëert en levert aan zijn klanten.
 - Organisatorische kennis: Interne structuren van
 - Eenheden
 - Afdelingen
 - Invloed
 - Communicatiekanalen

2. Analytisch denken/ logisch redeneren:

- Ontdekken
- Samenvatten
- Analyseren
- Identificeren
- Evalueren



3. Organiseren en tijdsbeheer

- Organiseren ...
- Verschillende soorten vergaderingen
- Complexe plannen beheren
- Veelvuldige veranderingen beheren

4. Communicatie en interactie: terwijl ze neutraal zijn, vergemakkelijken...

5. Hulpmiddelen en technieken

→ BABOK

In deze cursus sterke focus op BPMN

4. BA Planning en toezicht.

(Zeer belangrijk: wat gaan we doen? Hoe gaan we het doen, wat zijn de KPI's die we daarop willen zetten? Etc. Dit wordt op voorhand gedaan. Essentiële manier om te verhinderen dat je te maken krijgt met bijvoorbeeld heel sterk veranderende requirement over tijd, runover van je project, dat er andere stakeholders opduiken...)

Kennisgebieden:



Kennisgebieden en taken:

Business Analysis (BA) Planning & Monitoring	Elicitation	Requirements Management & Communication	Enterprise Analysis	Requirements Analysis	Solution Assessment & Validation
<ul style="list-style-type: none"> Plan BA Approach Conduct Stakeholder Analysis Plan BA Activities Plan BA Communication Plan Requirements Management Process Manage BA Performance 	<ul style="list-style-type: none"> Prepare for Elicitation Conduct Elicitation Activity Document Elicitation Results Confirm Elicitation Results 	<ul style="list-style-type: none"> Manage Solution Scope & Requirements Manage Requirements Traceability Maintain Requirements for Re-use Prepare Requirements Package Communicate Requirements 	<ul style="list-style-type: none"> Define Business Need Assess Capability Gaps Determine Solution Approach Define Solution Scope Define Business Case 	<ul style="list-style-type: none"> Prioritize Requirements Organize Requirements Specify and Model Requirements Determine Assumptions & Constraints Verify Requirements Validate Requirements 	<ul style="list-style-type: none"> Assess Proposed Solution Allocate Requirements Assess Organizational Readiness Define Transition Requirements Validate Solution Evaluate Solution Performance

BA Planning Parameters: ONSAACA → Houvast/ vaak gebruikte model

Objectief/ doel:

- Wat is het resultaat van de bedrijfsanalyse?
- Wat moet er worden opgeleverd?

Behoeftte: wat is het waargenomen probleem, de behoefte of de kans die moet worden onderzocht?

Gemeenschappelijk begrip van ...

- Wat zijn de percepties van het probleem/de behoefte/de kans?
- Waarom is het een probleem?
- Hoe beïnvloedt het bedrijf?
- Wat zijn de gewenste effecten of waarde van een oplossing?

Toepassingsgebied: Wat is de reikwijdte van het project?

De grenzen van het project bepalen:

- Duidelijkheid over welke aspecten en gebieden in aanmerking moeten worden genomen
- Duidelijkheid over waar geen rekening mee moet worden gehouden
- Planning van de activiteiten mogelijk maken

Aanpak: Wat is de belangrijkste aanpak (voorspellend versus adaptief)?

→ Intern beleid van softwareontwikkelingsmethode bepalen of volgen (voorspellende of agile aanpak)

Aanpak voor het verkrijgen van de resultaten aanpassen aan de context van

- De verandering
- Betrokken middelen
- Standaarden en methoden
- Beperkingen

Activiteiten: Hoe moet het resultaat worden bereikt (activiteiten, planning, tijdschema, methoden, instrumenten enz.)

Wat is nodig om het resultaat te bereiken gezien

- Activiteiten
- Hulpmiddelen/methoden
- Sequentiële afhankelijkheid
- Eenmalig versus iteraties
- Beschikbaarheid van belangrijke middelen
- Prioriteit

Overweeg ook, indien van toepassing,

- Afhankelijkheden
- Beperkingen
- Beperkingen

Complexiteit en risico's: Hoe complex is het project? Wat zijn de risico's van het project?

Complexiteit: Houd bij het maken van het plan ook rekening met de complexiteit die wordt beïnvloed door bijvoorbeeld

- Omvang van de verandering
- Aantal gebieden en systemen beïnvloed
- Aantal locaties
- enz

Risico: Houd bij het maken van het plan ook rekening met het risico dat wordt beïnvloed door bijvoorbeeld

- Ervaring met soortgelijke problemen
- Algemeen sentiment
- Ervaring en kennis van belanghebbenden enz.

⇒ Hoe haalbaar is het project ?

Goedkeuring: Goedkeuring verkrijgen en zorgen voor de nodige middelen om het werk uit te voeren.

Een plan dat kan worden goedgekeurd door de belangrijkste belanghebbenden en middelen die zijn toegewezen om met de bedrijfsanalyse te beginnen

- Tijd
- Kosten
- Middelen
- Resultaten

Andere kennisgebieden en taken

1. Bedrijfsanalyse:

→ Het "grote geheel" begrijpen

- Zakelijke doelen definiëren waaraan de oplossing moet voldoen
- Vereisten integreren in een grotere bedrijfsarchitectuur
- Ondersteunen van initiatieven en langetermijnplanning
- Strategische planning, business caseontwikkeling, kosten-batenanalyse, haalbaarheidsstudies.

→ "Waarom doen we dit?"

2. Elicitatie

- Focus op het verzamelen van eisen van verschillende groepen belanghebbenden
- Identificeer de taken, kennis en technieken voor het vastleggen van vereisten → In kaart brengen

→ "Wat hebben de Stakeholders nodig?"

Technieken:

- Brainstormen
- Focusgroepen
- Interviewen
- Observatie
- Prototypering
- Requirements Workshop
- Enquête/vragenlijst
- Analyse van documenten
- Analyse van de interface

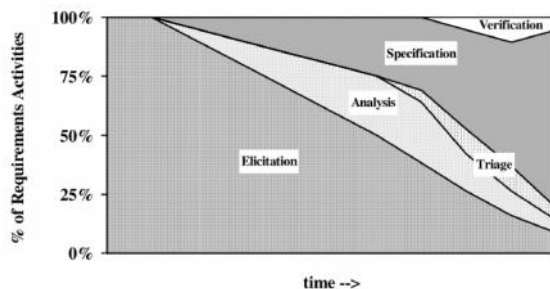
3. Vereistenbeheer en communicatie

Vereisten analyse

- Richt zich op het analyseren van de gegevens
- Definieert de methoden, instrumenten en technieken om de ruwe gegevens die tijdens de elicatie zijn verzameld te structureren
 - Alle requirements samenbrengen in een coherent geheel (verschillende stakeholders hebben verschillende noden, tegensprekingen, etc.)
- Identificeert gaps in de vereisten
- Definieert de "oplossing" mogelijkheden en kan dienen als de basis voor het selecteren van oplossingsalternatieven.

→ "Wat moet de oplossing doen?"

% van vereiste activiteiten



Wordt later in detail besproken

Vereistenbeheer en communicatie:

- Focus op het presenteren en communiceren van gedocumenteerde vereisten aan alle belanghebbenden, inclusief projectteamleden, om de groep tot consensus te brengen over het projectbereik.
- Identificeren en beheren van verandering

→ "Begrijpt en is iedereen het ermee eens?"

4. Beoordeling en validatie van oplossingen

- Focus op het verzekeren dat de beste aanpak wordt gekozen, dat de oplossing voldoet aan de doelstellingen van de belanghebbenden, dat de oplossing haalbaar is, en leidt de "verificatie" van de oplossing.
- Hoe bepalen we dat een BA-project succesvol is?

→ "Doet de oplossing wat het moet doen?"

5. Stakeholder engagement

Stakeholders:

= een persoon of groep met een relatie tot de verandering of oplossing.

→ De meest algemene versie van dit idee is "relevante mensen". BA's moeten zich meestal niet richten op mensen die geen relatie hebben met de oplossing of de verandering, want die zijn meestal niet relevant. Maar wat maakt een persoon relevant voor een verandering of voor een oplossing? Wat maakt van een persoon een stakeholder?

- Een persoon is een stakeholder als hij door een verandering of oplossing een verandering in waarde ervaart, of als hij de door een stakeholder ervaren waarde kan beïnvloeden.
- Het ervaren van een waardeverandering betekent niet dat u een change agent bent, of direct betrokken bij een gecontroleerde transformatie van organisatorische systemen.
 - o Denk bijvoorbeeld aan een project dat personeel gaat verhuizen naar een nieuw kantoor. De families van de mensen die verhuizen zijn stakeholders, niet omdat ze deel uitmaken van de verandering, maar omdat ze invloed hebben op de waardeverandering die de medewerkers ervaren.
- Stakeholders maken deel uit van het systeem, en zijn waardevolle bronnen van informatie voor de Bedrijfsanalyse.

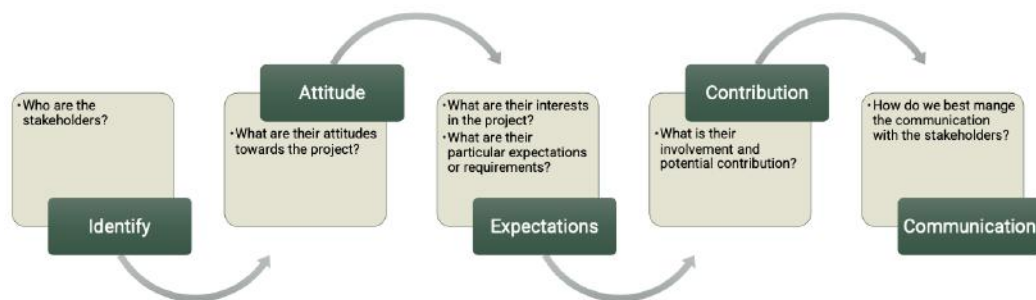
→ Belangrijkste stakeholders: businessponsors: degene die het project financieren. Je gaat voor deze stakeholder het project in kaart brengen en managen.

Belang van betrokkenheid van belanghebbenden:

Voor een gezonde en productieve samenwerking en om ervoor te zorgen dat het analyseproces vordert

- ✓ Efficiënte communicatiestrategie (steun verzamelen in plaats van weerstand bestrijden).
- ✓ Betere tevredenheid van belanghebbenden (meer participatie)
- ✓ Beperking van scope creep (meer duidelijkheid vermindert conflicten)

Wat we willen weten :



Stakeholders zijn niet enkel de businesssponsor ! Soms heb je voor de verwezenlijking van jouw project ook stakeholders nodig, hun input nodig...

→ Op basis van de categorie van stakeholders (afhankelijk van hun contributie) ga je een aangepaste communicatiestrategie uitwerken!

Stakeholder engagement:



Wie zijn de stakeholders?



- Interne: binnen het bedrijf
 - Externe: buiten het bedrijf
- Moeilijkste groep om te managen: ze zijn minder betrokken. Je hebt ze voor het project nodig maar ze staan niet dicht bij jou.

Vb. Partners, suppliers...

1. Bedrijfsanalist

De bedrijfsanalyse is per definitie een belanghebbende bij alle bedrijfsanalyseactiviteiten.

→ De BABOK® Gids is geschreven met het vermoeden dat de bedrijfsanalyse verantwoordelijk en aanspreekbaar is voor de uitvoering van deze werkzaamheden. In sommige gevallen kan de bedrijfsanalist ook verantwoordelijk zijn voor het uitvoeren van activiteiten die onder een andere stakeholderrol vallen.

2. Klanten

Een klant is een belanghebbende buiten de grenzen van een bepaalde organisatie of organisatie-eenheid. Klanten maken gebruik van door de organisatie geproduceerde producten of diensten en kunnen contractuele of morele rechten hebben waaraan de organisatie moet voldoen.

3. Domein Subject Matter Expert (MKB)

Een domeinexpert is elke persoon met diepgaande kennis van een onderwerp dat relevant is voor de zakelijke behoefte of de reikwijdte van de oplossing. Deze rol wordt vaak vervuld door mensen die ook eindgebruikers zijn of mensen die indirecte gebruikers van de oplossing zullen zijn, zoals managers, proceseigenaren, juridische medewerkers (die kunnen optreden als volmachten voor regelgevende instanties), consultants en anderen.

4. Eindgebruiker

Eindgebruikers zijn belanghebbenden die rechtstreeks met de oplossing zullen communiceren.

De term wordt het meest gebruikt in de context van softwareontwikkeling, waarbij eindgebruikers degenen zijn die de softwaretoepassing die wordt ontwikkeld daadwerkelijk zullen gebruiken, maar in de bredere context van een oplossing kunnen ze alle deelnemers aan een bedrijfsproces omvatten.

5. Implementatie Subject Matter Expert (MKB)

- Ontwikkelaars/Software Engineers
- Organizational Change Management Professionals
- Systeemarchitecten

- Trainers
- Bruikbaarheid Professionals

6. Projectmanager

Projectmanagers zijn verantwoordelijk voor het managen van het werk dat nodig is om een oplossing te leveren die voldoet aan een zakelijke behoefte, en om ervoor te zorgen dat de projectdoelstellingen worden bereikt, terwijl ze de projectbeperkingen in evenwicht houden, waaronder reikwijdte, budget, planning, middelen, kwaliteit, risico's en andere.

7. Tester

Testers zijn verantwoordelijk voor het bepalen hoe ze kunnen verifiëren dat de oplossing voldoet aan de oplossingsvereisten die zijn gedefinieerd door de bedrijfsanalist, en voor het uitvoeren van het verificatieproces. Testers proberen er ook voor te zorgen dat de oplossing voldoet aan de geldende kwaliteitsnormen en dat het risico op defecten of storingen wordt begrepen en geminimaliseerd.

8. Regelaar

Toezichthouders zijn verantwoordelijk voor het definiëren en handhaven van normen. Normen kunnen normen zijn waaraan het team dat de oplossing ontwikkelt moet voldoen, normen waaraan de oplossing moet voldoen, of beide. Regelgevers kunnen wetgeving, normen voor corporate governance, auditnormen of normen die zijn gedefinieerd door competentiecentra van de organisatie afdwingen.

9. Sponsor

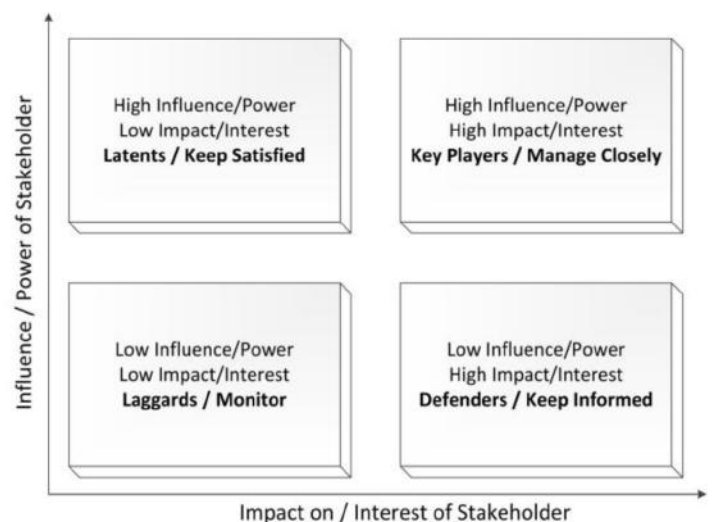
Sponsors zijn verantwoordelijk voor het initiëren van de inspanning om een zakelijke behoefte te definiëren en een oplossing te ontwikkelen die in die behoefte voorziet. Ze geven toestemming voor het uitvoeren van de werkzaamheden en controleren het budget voor het initiatief.

10. Leverancier

Een leverancier is een stakeholder buiten de grens van een bepaalde organisatie of organisatieonderdeel. Leveranciers leveren producten of diensten aan de organisatie en kunnen contractuele of morele rechten en verplichtingen hebben waarmee rekening moet worden gehouden.

Stakeholder analyse:

- Stakeholdermatrix:
Gebaseerd op
 - Kracht van invloed
 - Interesse in het verbeterinitiatief
- ⇒ Gaat de stakeholder verdelen in 4 categorieën:
 - Keep Satisfied/ Latents
 - Key players/ Manage Closely
 - Laggards/ Monitor
 - Defenders/ Keep informed

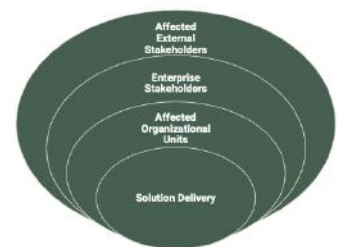


Stakeholder	Invloed	Impact	Stakeholder management
Tevreden houden	Hoog	Laag	Behouden en zorgen dat de stakeholder tevreden blijft
Sleutelspeler	Hoog	Hoog	Werk nauw samen om ervoor te zorgen dat ze het eens zijn over de wijzigingen
Monitor	Laag	Laag	Houd de belangen of invloed van de stakeholders in de gaten bij wijzigingen
Houd geïnformeerd	Laag	Hoog	Zorg voor een goede informatiestroom om hen op de hoogte te houden

→ Afhankelijk van de groep ga je een bepaalde manier van management gebruiken

- Onion diagram: breder tool, multidimensionaal
 - o Hier bekijk men meer wie dicht en wie ver van het project ligt.

- Solution Delivery: BA-team
- Affected Organizational Units: zijn betrokken of worden geïmpacteerd maar zitten niet rechtstreeks in het team
- Enterprise Stakeholders: meer high level mensen die niet betrokken zijn maar wel geïmpacteerd kunnen worden door het project
- Affected External Stakeholders: externe



- RACI Matrix: meest populaire model
 - o Je neemt ook de stappen/ taken in jouw analyse
 - Zeer gedetailleerd
 - o 4 categorieën:
 - Responsible: persoon die eindverantwoordelijk neemt
 - Accountable: persoon die medeverantwoordelijk is en er kan beroep op gedaan worden wanneer er iets fout gaat. (Niet rechtstreeks leidend)
 - Consulted: je bent op de hoogte en er wordt ook iets actief verwacht van jou zoals feedback
 - Informed: Je bent op de hoogte
 - (Support)

- Responsible (R), Accountable (A), Consulted (C), Informed (I)
- Sometimes RASCI with S for Support

	Business Analyst	Subject Matter Expert	Developer	Tester	System Administrator	Project Manager	User
Define Problem	R	C				A	C
Analyze Current Situation	R	C				A	C
Define Future State	R	I				A	C
Design Solution	R		I	I		A	C
Develop Solution	I		R	I		A	I
Test Solution	C, I		I	R	I	A	R
Install Solution	I				R	A	
Training	C					A	C, I

Deliverable/Phase/Activity	R: Responsible	A: Accountable	C: Consulted	I: Informed
Current State Analysis	Business Analyst	Sponsor	Subject Matter Expert	Team
Project Plan	Project Manager	Sponsor	Subject Matter Expert	Team
Requirements Analysis	Business Analyst	Project Manager/ Business Analyst	Subject Matter Expert/ Business Users	Project Manager

Stakeholder management:

Managen van Stakeholders gaat over communicatie (plan)

- **Waarom** – reden om te communiceren
 - o de stakeholder tevreden houden,
 - o ervoor zorgen dat ze het initiatief blijven steunen,
 - o om ze op de hoogte te houden voor hun aanstaande grotere betrokkenheid of
 - o is om een belangrijke beslissers op de hoogte te houden om de te nemen beslissingen te vergemakkelijken?
- **Wat** – informatie om te communiceren
 - o Verschillende belanghebbenden zullen geïnteresseerd zijn in verschillende soorten informatie.
- **Wanneer** – frequentie
 - o van een schaal vanaf dag/week tot kwartaalrapportages
- **Hoe** – manieren van communiceren
 - o van het verzenden van e-mails en statusupdatevergaderingen tot persoonlijke ontmoeting
 - Email soms niet voldoende als het gaat om accountable personen.

Voorbeeld communicatieplan:

Stakeholder	Why	What	When	How
CIO	Sponsor	Progress, Solutions, Challenges, Requirements, Budget	Weekly	Status Meetings
Project Manager	Will take over delivery	Progress, solutions, requirements	Monthly (weekly)	Project Meetings
Legal	Verify	Legal aspects	On needs basis	Meetings
Old CRM	Maintenance	Migration	Weekly	Project Meetings
Developer	Informed	Solution, Requirements	Monthly	Progress Meetings
Advisors	Verify	Solutions	On needs basis	Meetings/Worksh ops
End Users	Requirements	Solutions, Requirements	Weekly	Dedicated workshops
Customers	Informed	Relevant updates	On needs basis	Email

Conclusie:

- Bedrijfsanalyse is een breed domein dat is gericht op het identificeren en ontwikkelen van oplossingen binnen organisaties voor specifieke behoeften of veranderingen die bedrijfswaarde kunnen stimuleren
- Bedrijfsanalisten zijn doorgaans generalisten die verschillende rollen in een project vervullen, doorgaans met een gevarieerde vaardigheden set
- Een belangrijke succesfactor voor BA-projecten is goed stakeholdermanagement, gericht op analyse, management en communicatie

BUSINESS PROCES MANAGEMENT

BPM1. Introduction to BPM

1. The world of Business Process Management

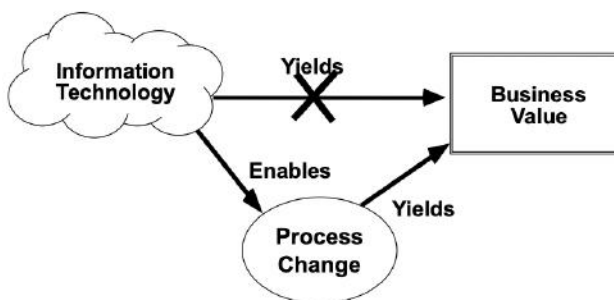
BPM/ Bedrijfsprocesmanagement: **wat** is het?

= Lichaam van principes, methoden en hulpmiddelen voor het ontwerpen, analyseren, uitvoeren en bewaken van bedrijfsprocessen.

→ Oorspronkelijk in manufacturing bedrijven : om de processen te industrialiseren

→ Nu alles omvormen naar de digitale wereld

Waarom BPM?



“De eerste regel van elke technologie die in een bedrijf wordt gebruikt, is dat automatisering toegepast op een efficiënte operatie de efficiëntie zal vergroten. De tweede is dat automatisering toegepast op een inefficiënte operatie de inefficiëntie zal vergroten.”

Belangrijkste doel: efficiëntie vergroten! Niet enkel het digitaliseren.!

Hoe deel te nemen aan BPM?

- Continue procesverbetering (CPI)
 - o Stelt de huidige processtructuur niet ter discussie
 - o Probeert problemen te identificeren en stapsgewijs op te lossen, stap voor stap en één oplossing tegelijk
- Business Process Re-Engineering (BPR)
 - o Stelt de fundamentele aannames en principes van de bestaande processtructuur ter discussie
 - o Streeft naar doorbraak, bijvoorbeeld door het schrappen van kostbare taken die niet direct waarde toevoegen

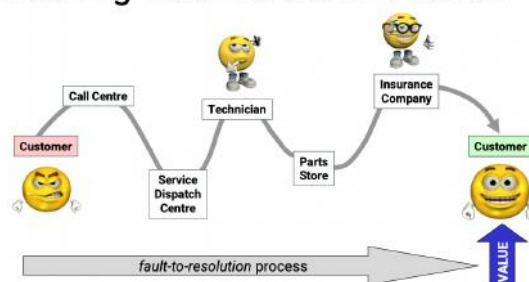
Onze interessante fenomenen: Business processen

- Verzameling van gerelateerde gebeurtenissen, activiteiten en beslissingen, waarbij een aantal actoren en objecten betrokken zijn, en die gezamenlijk leiden tot een resultaat dat van **waarde** is voor een organisatie of haar **klanten**.

Voorbeelden:

- o Order-naar-contant
- o Offerte op bestelling
- o Procure-to-Pay
- o Aanvraag tot goedkeuring
- o Fault-to-Resolution (Uitgifte-tot-Resolutie)
- o Claim-to-Settlement

“My washing machine doesn’t work...”

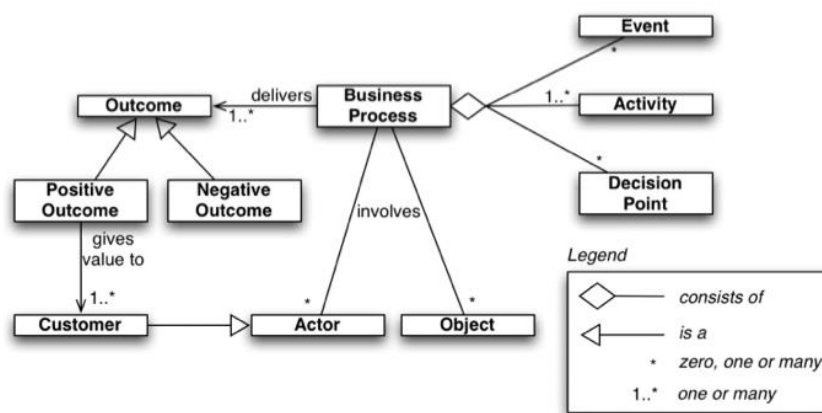


Processen en resultaten

- Elk proces leidt tot één of meerdere uitkomsten, positief of negatief
 - o Positieve uitkomsten leveren waarde op
 - o Negatieve uitkomsten verlagen waarde

We verwachten minimum 1 positieve uitkomst in de bussinesproces anders betekent het dat er geen weg is om waarde te creëren.

- Uitkomsten van het fout-tot-oplossingsproces:
 - o Storing gerepareerd zonder tussenkomst van een technicus
 - o Storing gerepareerd met kleine tussenkomst van een monteur
 - o Storing gerepareerd en volledig gedekt door garantie
 - o Storing gerepareerd en deels gedekt door garantie
 - o Storing gerepareerd maar niet gedekt door garantie
 - o Storing niet gerepareerd (klant heeft verzoek ingetrokken)



Customer is niet noodzakelijk buiten het bedrijf. Je kan een klant hebben binnen het bedrijf.

De kernelementen van een proces

- Activiteiten
 - o Actieve elementen (bijv. 'verkooporder invoeren')
 - o Tijdrovend, veeleisend
 - o Toestandsveranderend
- Evenementen
 - o Passieve elementen (bijv. 'verkooporder is ingevoerd')
 - o Vertegenwoordigen voorwaarden / omstandigheden
 - o Atomair, onmiddellijk
- Bedrijfsobjecten (of gegevens)
 - o De organisatorische artefacten die toestandsveranderingen ondergaan
 - o Fysieke of elektronische informatie
 - o Voorbeelden:
 - Verkooporder, digitaal object, adviesvoorstel
- Acteurs (of bronnen)
 - o De entiteiten die procesactiviteiten uitvoeren en gebeurtenissen genereren • mens en systemen
 - o Voorbeelden:
 - Financieel medewerker, magazijnmedewerker • ERP, CRM, SAP, applicatie X...

Hoe combineren we deze? => door te modelleren

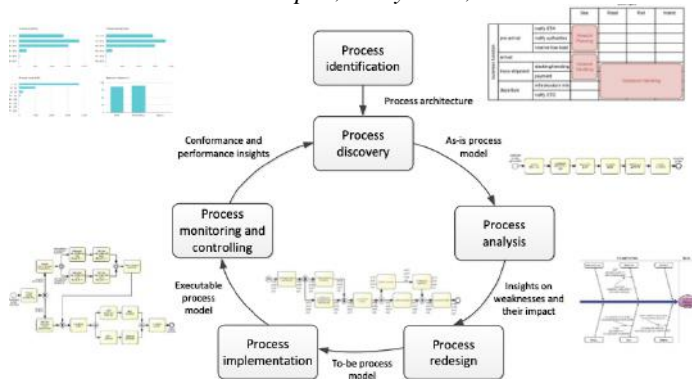
1. Wat moet er gebeuren en wanneer? – Control flow
2. Waar moet ik aan werken? – Gegevens/ data
3. Wie doet het werk? - Middelen (mensen en systemen)

Procesperspectieven:

- Controle stroom perspectief:
 - o Wat er moet gebeuren en wanneer”
 - o Voorganger/opvolger relatie tussen activiteiten en gebeurtenissen
 - o De centrale informatie weergegeven in een procesmodel
- Data perspectief:
 - o “Waar moeten we aan werken”
 - o Input/output data naar activiteiten
 - o Complementeert de control flow
- Bronperspectief
 - o "wie doet het werk"
 - o Menselijke deelnemers en systemen die controlestroomactiviteiten uitvoeren en gebeurtenissen genereren
 - o Complementeert de regelstroom

2. De BPM-Levenscyclus

...ontwerpen, analyseren, uitvoeren en bewaken van bedrijfsprocessen...



Process identificatie

- ⇒ *Het grootste verschil met alle andere stappen in de lifecycle: het focust niet op 1 proces maar op meerdere processen. Hier willen we juist één proces identificeren dat je als business analyst team wil verbeteren.*
- Wat?
 - o Identificeer de bedrijfsprocessen van een organisatie
 - o Geef prioriteit aan hun beheer op basis van bepaalde criteria
- Waarom?
 - o Begrijp de organisatie
 - o Maximaliseer de waarde van BPM-initiatieven

Stappen:

1. Aanwijzingsfase
 - o Hoofdprocessen opsommen
 - o Processcope bepalen
 - ➔ Proces architecture je hebt een goede overzicht van alle processen dat bestaan
2. Evaluatiefase (a.k.a process selectie)
 - o Prioriteer het proces op basis van:
 - Belang
 - Gezondheid
 - Haalbaarheid
 - ➔ Geprioriteerde procesportfolio

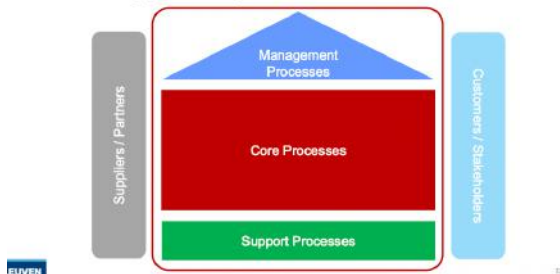
Proces opsomming:

“De meeste bedrijven hebben slechts drie kernprocessen:

1. Dingen verkopen

2. Lever spullen
 3. Zorgen dat je spullen hebt om te verkopen en af te leveren”
- ⇒ Maar te simplistisch

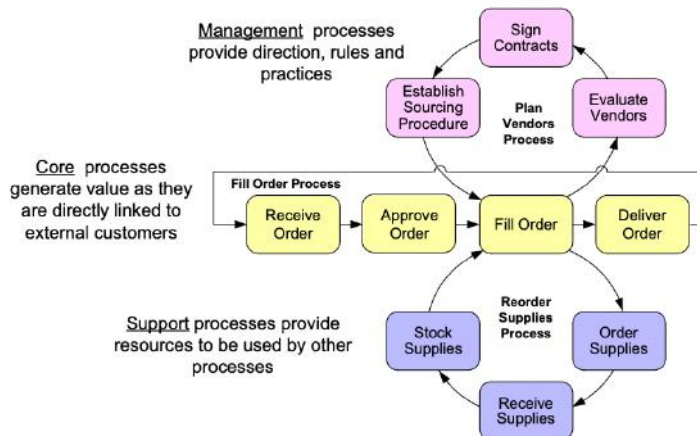
Porter: Types of processes



Voorbeeld: de kern-, ondersteunings- en managementprocessen van een groothandelaar.
en managementprocessen

- Kernprocessen
 - o Verkoop (lead-to-quote, offerte-to-order, order-to-cash)
 - o Purchase-to-Pay (directe inkoop, bijv. bevoorrading)
 - o ...
- Ondersteunende processen
 - o Purchase-to-pay (indirecte inkoop, bijv. aanvulling van onderdelen, operationele middelen aanvulling...)
 - o HR (bijwerken van beleid, werving, introductie, proeftijd...)
 - o ...
- Beheerprocessen
 - o Leveranciersbeheer (leveranciersplanning, leverancierswerving...)
 - o Logistiek beheer (logistieke planning, logistieke controle...)
 - o ...

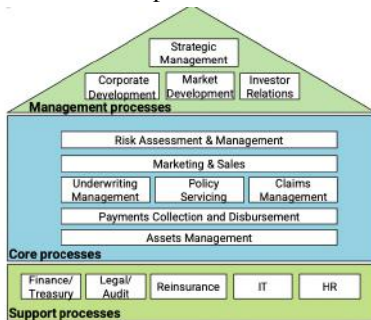
Relaties tussen kern-, ondersteunings- en managementprocessen



Voorbeeld: procesarchitectuur van de groothandel

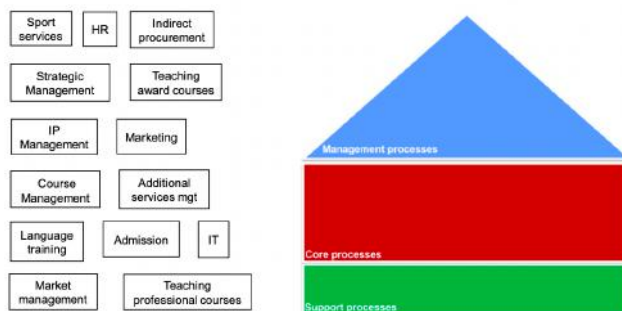


Voorbeeld: de procesarchitectuur van een verzekeringsmaatschappij

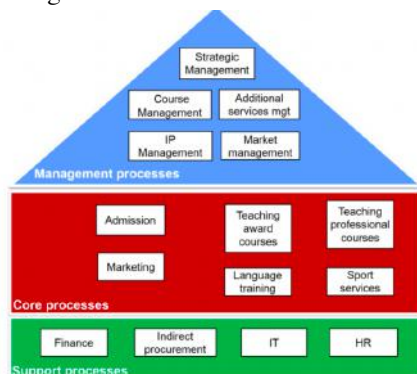


Oefening: indelen per procestype

- Deze groepen processen worden typisch uitgevoerd aan een universiteit. Classificeer elke procesgroep als kernproces, ondersteunend proces of managementproces. (Kan verschillend zijn voor elke bedrijf)



Oplossing:



Scoping van processen

Processen zijn onderling afhankelijk => inzicht in onderlinge relaties vereist

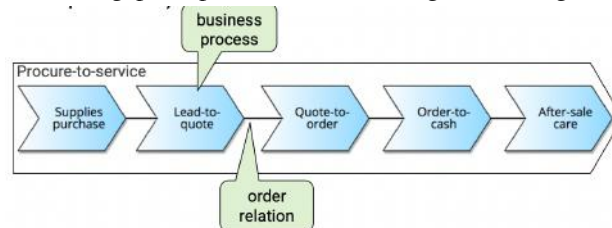
- Specialisatie: algemeen - speciaal product/dienst
- Horizontaal: upstream - downstream processen en hun waardeketens.

- Hoe zetten we deze bedrijfsprocessen naast elkaar. Hoe zijn de processen op elkaar afgestemd in termen van waarde
- Verticaal: hoofdprocessen - deelprocessen.
 - Moeilijker.
 - ➔ Procesarchitectuur



Modellering van de waardeketen

- Keten van processen die een organisatie uitvoert om waarde te leveren aan klanten en belanghebbenden.
- Meer in het algemeen, een mechanisme om bedrijfsprocessen op hoog niveau te groeperen volgens een orderrelatie (kan worden toegepast op kern-, ondersteunings- en managementprocessen)

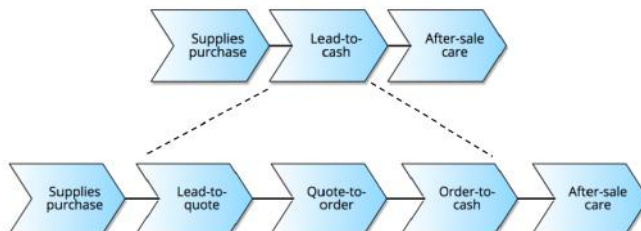


Richtlijnen voor het vaststellen van horizontale grenzen in waardeketens

1. Verandering van een belangrijk bedrijfsobject in het proces
2. Wijziging van de korrelgrootte van het hoofdzakelijke voorwerp
3. Verandering in frequentie/tijd
4. Wijziging in tussentijds resultaat/oplossing/doelstelling

Voorbeeld: waardeketen van de groothandelaar

- Kernprocessen



Typische waardeketens voor kernprocessen

- Denk rond drie hoofdstappen:
 - Bedenk het (ontwerp nieuw product/dienst)
 - Bouw het (product/dienst inkopen, assembleren, leveren)
 - Verkoop het (product/dienst op de markt brengen, verkopen, service verlenen)
- Voorbeeld: producent
 - Specialisaties

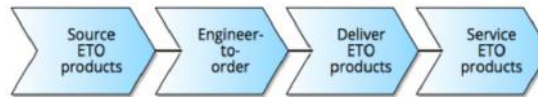
- Stocked products



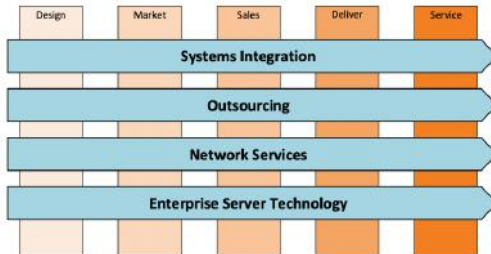
- MTO products



- ETO products

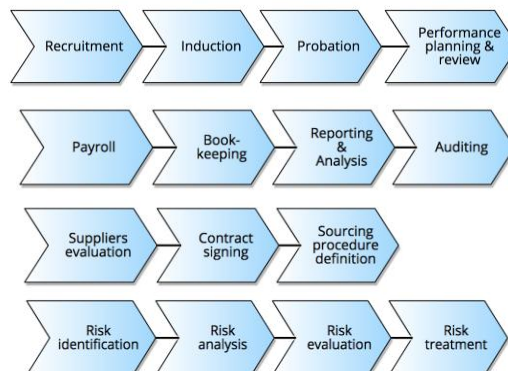


Voorbeeld: waardeketens voor een IT-dienstverlener

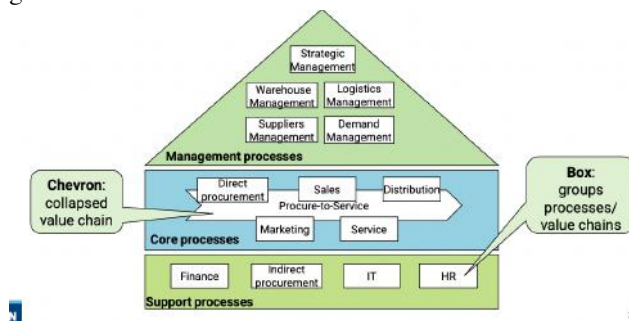


Voorbeeld: waardeketen van niet-kernprocessen

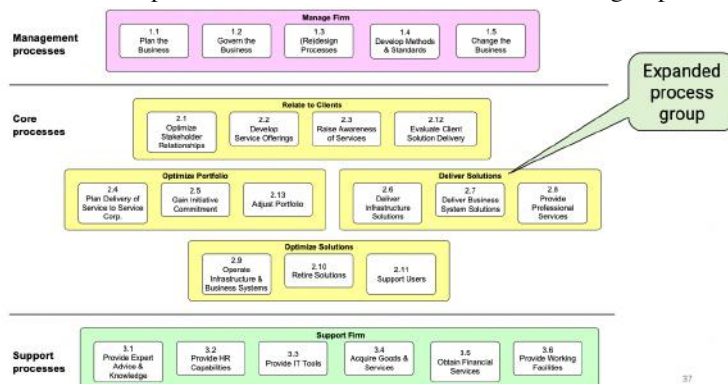
- Support processes
 - o HR
 - o Accounting
- Management processes
 - o Suppliers management
 - o Risk management



Voorbeeld: procesarchitectuur & waardeketens van de groothandelaar



Alternatief: de procesarchitectuur van een adviesbureau – groepen

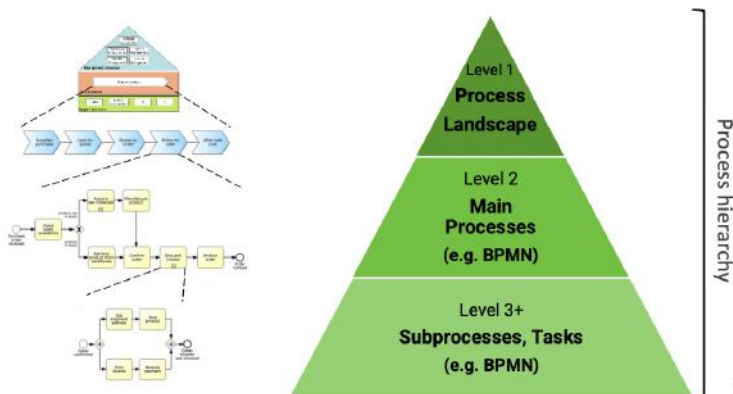


Typische artefacten voor verticale scoping

- Waardeketens
 - o Ketens van processen. Blijf op een hoog niveau. Vuistregel: 3-7 processen
 - Bv. Procure-to-service, Risicomanagement

- (Hoofd)processen
 - o bouwen waardeketens op en beïnvloeden elkaar. Ze zijn abstract
 - Bijv. Lead-to-quote, Quote-to-order, Order-to-cash
- ⇒ Typische focus van proces opsomming
- Subprocessen
 - o Opbouwende processen. Ze zijn gedetailleerd, omvatten meerdere activiteiten en kunnen gelaagd zijn op verschillende abstractieniveaus (d.w.z. sub-subprocessen).
 - Bijv. orderverzending, facturering
- Procestaken
 - o Processen en subprocessen opbouwen. Ze zijn atomair en worden uitgevoerd door mensen, IT-systemen of apparatuur.
 - Bijv. goedkeuring van de factuur

Procesarchitectuur: hiërarchische visie



Voorbeeld: hiërarchische procesarchitectuur van een verzekeringsmaatschappij

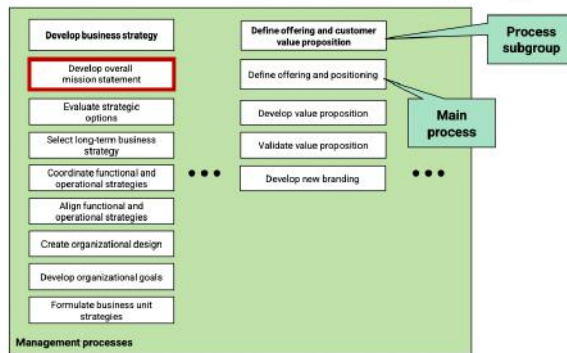
Level 1



• Level 2

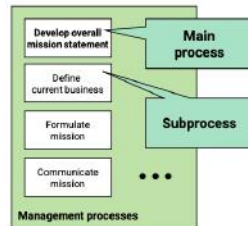


•Level 3



VEN

•Level 4



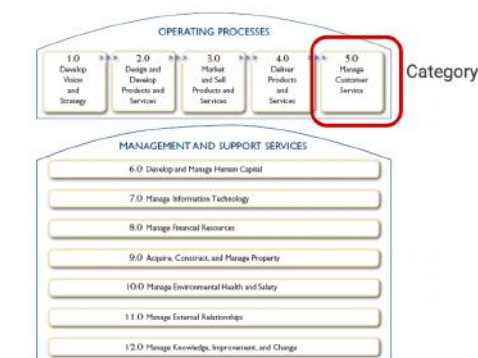
Designation via referentiemodellen

- Een referentiemodel wordt gebruikt als sjabloon om de procesarchitectuur te ontwerpen.
- Voorbeelden:
 - o Information Technology Infrastructure Library (ITIL)
 - o Referentiemodel voor Supply Chain Operations (SCOR)
 - o Procesclassificatiekader (PCF)
 - o Control Objectives for Information Technology (COBIT)
 - o Value Reference Model (VRM)
 - o Vrijwillige interindustriële handelsooplossingen (VICS)
 - o eTOM kader voor bedrijfsprocessen
 - o Prestatiekader

Voorbeeld: APQC Process Classification Framework (PCF)

- Industrie-neutraal bedrijfsmodel
- Open standaard voor benchmarking
- Vier niveaus
 - o Categorieën
 - o Procesgroep
 - o Proces
 - o Activiteit

APQC PCF Overview



APQC Classification framework

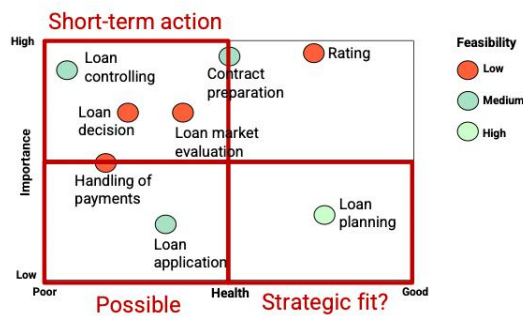


De evaluatiefase (ook wel processelectie)

1. Belang
 - Welke processen hebben de grootste impact op de strategische doelen van de organisatie?
2. Gezondheid (of disfunctie)
 - Welke processen zitten het diepst in de problemen?
3. Haalbaarheid
 - Welk proces is het meest vatbaar voor succesvol procesmanagement?

➔ Prioritized Process Portfolio

Voorbeeld: geprioriteerde procesportefeuille van een financiële instelling

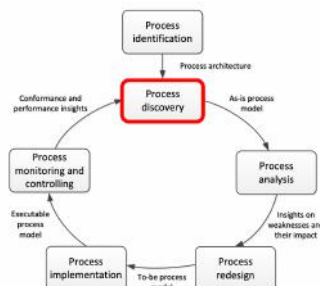


- Possible kwadrant: meeste berdjiven gaan ze outsourcen
- Strategic fit: zijn deze processen nog steeds belangrijk ?

3. Process discovery: as-is process modelling

BPM levenscyclus

...bedrijfsprocessen ontwerpen, analyseren, uitvoeren en controleren...



Proces ontdekking

1. Definieren van de setting
 - Deze fase is gewijd aan het samenstellen van een team in een bedrijf dat verantwoordelijk zal zijn voor het werken aan het proces.
2. Informatie verzamelen

In deze fase gaat het erom inzicht te krijgen in het proces. Verschillende ontdekkingsmethoden kunnen worden gebruikt om informatie over een proces te verzamelen.

3. Het uitvoeren van de modelleringstaak

Deze fase betreft het organiseren van de creatie van het procesmodel. De modelleermethode geeft een leidraad voor het systematisch in kaart brengen van het proces.

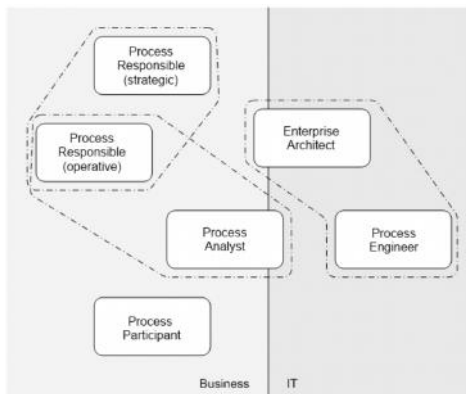
4. Het waarborgen van de kwaliteit van het procesmodel (zie later)

Deze fase heeft tot doel te garanderen dat de resulterende procesmodellen aan verschillende kwaliteitscriteria voldoen. Deze fase is belangrijk voor het vertrouwen in het process model.

Wie is betrokken ?

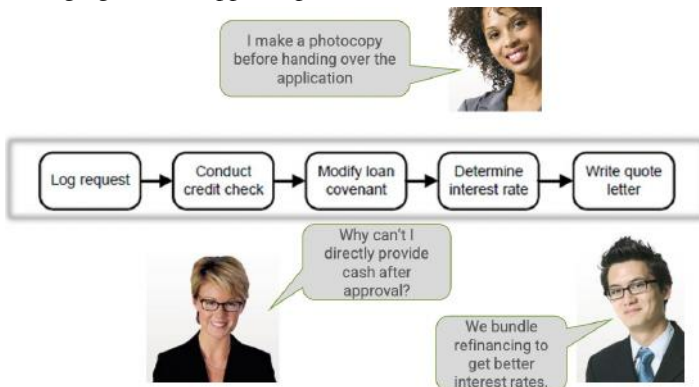
- Domeinexpert
- Procesanalist

Belanghebbenden/ Stakeholders in detail



→ Het is niet zwart wit: soms overlapping

Uitdaging 1: Versnipperde proceskennis



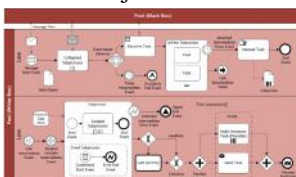
Ze kennen alleen hun deel en niet hoe het fit in de gehele proces. Ze hebben een lokale zicht. Als een business model heb je alle perspectieven nodig om 1 model te maken.

Uitdaging 2: Domeinexperts denken op instance-niveau

- "Elke reis is anders."
 - "Je kunt niet echt vergelijken. Onze klanten gaan naar verschillende plaatsen in verschillende seizoenen met verschillende vervoersmiddelen."
 - "We kunnen nooit iets precies op dezelfde manier doen. Er zijn zoveel speciale omstandigheden."
- Als business analyse moet je de gemeenschappelijke kenmerken zoeken.

Uitdaging 3: Kennis over procesmodellering is zeldzaam

- "Kunt u mij vertellen of dit diagram uw proces correct weergeeft?"



Elicitatietechnieken gebruikt in de fase van de procesontdekking (procesmodellering)

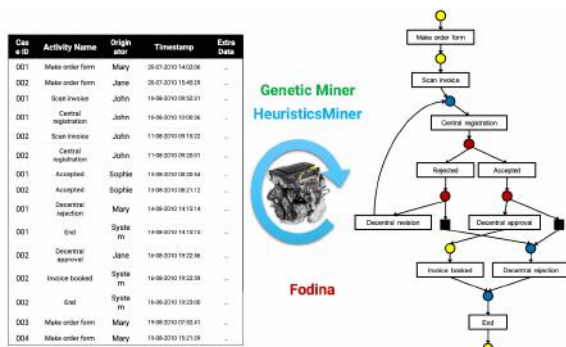
1. Documentanalyse
 - Documenten wijzen op bestaande rollen, activiteiten en bedrijfsobjecten
 - Formele documentatie in termen van
 - o Organogram
 - o Werkgelegenheidsplan
 - o Verslag van het kwaliteitscertificaat
 - o Intern beleid
 - o Woordenlijsten en handboeken
 - Formulieren
 - Werkinstructies
 - Gemakkelijk te gebruiken, snel te vinden, veel up to date informatie beschikbaar...
2. Observatie
 - Observeren wat mensen op hun werkplek doen
 - Traceer bedrijfsobjecten in de loop van hun levenscyclus
 - Inspecteer de werkomgeving

Probleem:

- o Observation bias → wanneer mensen weten dat ze worden geobserveerd gaan ze anders handelen.
- o Tijd consumerend
- o Je kan maar enkele cases observeren

Maar zeer gedetailleerd en geeft toegang tot veel informatie

3. Processmining: geautomatiseerd procesonderzoek



= je observeert digitale sporen die je hebt bijgehouden in de informatiesystemen die worden gebruikt om een bedrijfsproces uit te voeren.

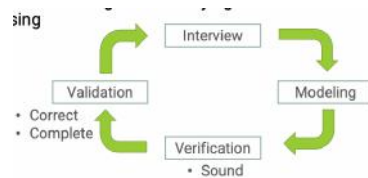
→ event-logs worden gevonden

Bv. Algoritmes die duizenden historische cases automatisch vertalen door middel van hun gedetailleerde logs in visuele representaties.

- VD: zeer objectief, gedetailleerd & zeer strek in het abstraheren het laag drempelige gedrag naar hogere level inzicht.
- ND: niet gemakkelijk toegang tot data & kwaliteit van data is in sommige bedrijven zeer laag.

→ 1, 2 & 3: Evidence based

4. Op interviews gebaseerde (Meest gebruikt)
 - Gestructureerde vs. ongestructureerde interviews
 - Veronderstelling: analist en belanghebbende delen terminologie
 - Vervolgens zijn de vragen gericht op het identificeren van afwijkingen van de standaardverwerking
- ND: tijdconsumerend, gefragmenteerde interview (je moet alle informatie van alle interviews samenbrengen), mensen kunnen liegen (=BIAS)
- VD: Je kan kiezen welke richting de interview gaat, kaan zeer gedetailleerd zijn



5. Workshop-based

- Alle belangrijke belanghebbenden samenbrengen
- Eén procesanalist, meerdere domeinexperts
- Deelnemers interageren om een gedeeld begrip te creëren
- Vaak: software ondersteund, een model wordt direct gecreëerd tijdens de workshop (aparte rol)
- Model is referentiepunt voor discussies - Alternatief: brown-paper workshops

VD: minder tijdconsumerend voor de business analyst, directe conflict resolution, na de workshop ben je zeer dicht bij het eindresultaat

ND: zeer moeilijk te plannen

Sterke en zwakke punten

Techniek	Sterke	Zwaktes
Document analysis	<ul style="list-style-type: none"> - Gestructureerde informatie - Onafhankelijk van beschikbaarheid van belanghebbenden 	<ul style="list-style-type: none"> - Verouderd materiaal - Verkeerd niveau van abstractie
Observatie	<ul style="list-style-type: none"> - Contextrijk inzicht in het proces 	<ul style="list-style-type: none"> - Mogelijk indringend - Belanghebbenden zullen zich waarschijnlijk anders gedragen - Slechts enkele gevallen
Automatisch ontdekken	<ul style="list-style-type: none"> - Uitgebreide reeks gevallen - Objectieve gegevens 	<ul style="list-style-type: none"> - Mogelijke problemen met de gegevenskwaliteit
Interview	<ul style="list-style-type: none"> - Gedetailleerd onderzoek naar het proces 	<ul style="list-style-type: none"> - Vereist weinig tijd van belanghebbenden in het proces - Meerdere iteraties nodig voor ondertekening
Workshops	<ul style="list-style-type: none"> - Rechtstreekse oplossing van tegenstrijdige standpunten 	<ul style="list-style-type: none"> - Synchrone beschikbaarheid van verschillende belanghebbenden

Organisatie van het verzamelde materiaal en uitvoering van de modelleringstaak

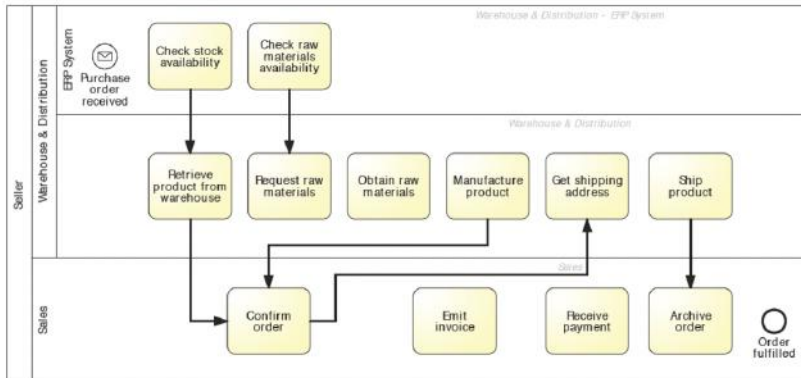
1. Identificeer de procesgrenzen
 - Onder welke voorwaarde begint het proces?
 - Met welk resultaat eindigt het?
 - Welk perspectief neem je aan?
2. Identificeer activiteiten en gebeurtenissen



3. Identificeer middelen en hun overdrachten



4. Identificeer de controlestroom



5. Identificeer aanvullende elementen

BPM2. Essential Process Modelling

1. Process modeling languages

Verschillende notaties

- OvergangssystemeN
- **Petri-netten (d.w.z. workflow-netten)**
- **BPMN**
- YAWL
- EPC's
- UML activiteitendiagrammen
- Causale netten
- BPEL
- IDEF3
- veel meer

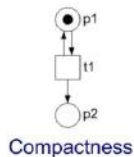
Petri-netten: Waarom ?



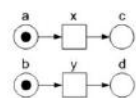
Math. foundation



Graphical notation



Compactness



Concurrency, locality



Analysis techniques

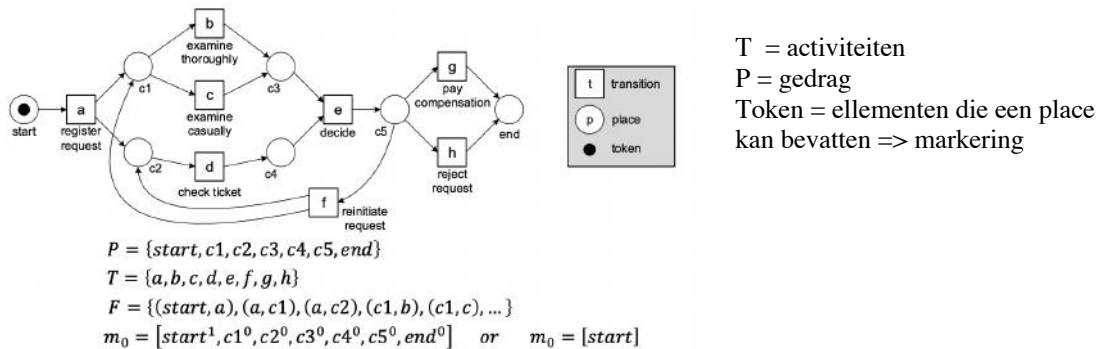


Tool support

BPM gebaseerd op Petri-netten

- petri-nets heeft een zeer sterke mathematische onderbouw. Als je iets met petri-net modeleert dan is er een 1 op 1 match tussen het systeem en het gedrag dat het systeem gaat modelleren. Er is dus geen misverstand mogelijk terwijl BPM wel vrijheid/ subjectiviteit geeft.
- Grafische notatie bij petri-nets is veel rudimentairder
- Zeer sterke analyse technieken

⇒ Een Petri net systeem (P, T, F, m_0) bestaat uit een Petri net (P, T, F) en een onderscheiden markering m_0 , de initiële markering

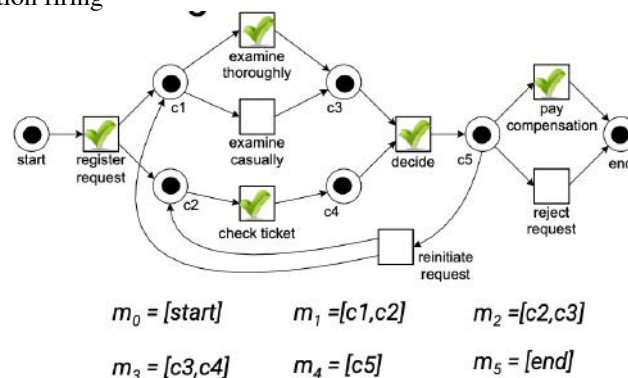


Uitvoeringssemantiek

- Een overgang kan afgaan/ afgevuurd worden wanneer deze ten minste één token bevat in elk van zijn invoerplaatsen.
- Wanneer een overgang afgaat:
 - o verbruikt hij één token van elk van zijn invoerplaatsen,
 - o produceert hij één token in elk van zijn uitvoerplaatsen.



Overgangsvuur / Transition firing



Only one of infinitely many possible firing sequences!

Je kunt gekke moeilijke dingen modelleren met Petri-netten

- Tunnelsysteem, één richting tegelijk, richting kan worden omgeschakeld, capaciteit is beperkt tot 10 auto's

→ In het kader van deze cursus beperken we ons tot werknetten

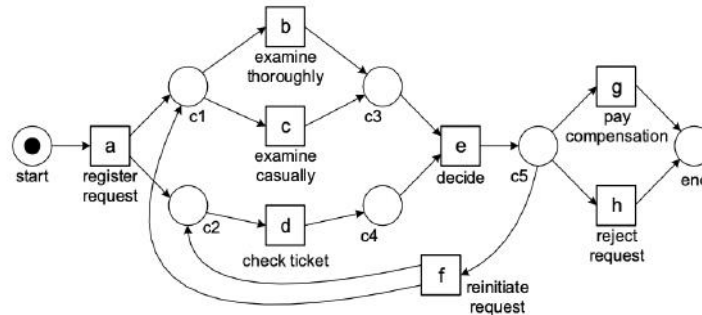
Werkstroomnetten (moeten we kunnen herkennen)

= Subklasse van Petri-netten, typisch gebruikt voor het modelleren van processen.

Kenmerken:

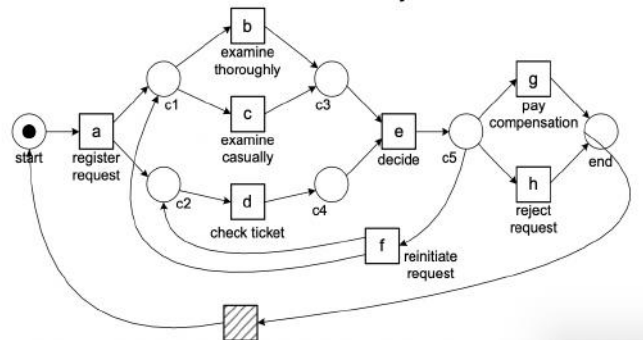
- Een invoerplaats (bronplaats genoemd): een plaats met alleen uitgaande bogen
- Een uitvoerplaats (sinkplaats genoemd): een plaats met alleen inkomende bogen

- Het net is sterk verbonden: er is een gericht pad tussen elk paar van knooppunten

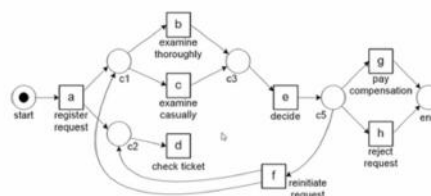


Sterk verbondenheid

- Het net is sterk verbonden als er een gericht pad is tussen twee willekeurige knooppunten.
 - o Vb. van a tot g is er een directe pad
 - o Vb. van g tot h is er geen direct pad dus we voegen een onzichtbare overgang toe
- Stel je een onzichtbare overgang voor tussen "einde" en "begin" (de zogenaamde "reset"-overgang)
 - o Heeft enkel het doel om te resetten maar is geen onderdeel van workflownet ! Het is enkel om te kunnen nakijken ofdat het sterk verbondenheid correct is.
- Dan moet elk paar elementen verbonden zijn door ten minste één gericht pad



Tegenvoorbeeld:



Business Process Modeling Notation (BPMN)

- OMG Standaard, ondersteund door vele tools:
 - o Bizagi Process Modeller
 - o Signavio (<http://www.signavio.com/>)
 - o TIBCO Business Studio (gratis download, vrij groot)
 - o IBM Websphere Business Modeler
 - o ARIS
 - o Oracle BPA
 - o Business Process Visual Architect (Visual Paradigm)
 - o Progress Savvion Business Modeller

Een BPMN procesmodel is een grafiek die bestaat uit vier soorten elementen (onder andere):



Event



Task



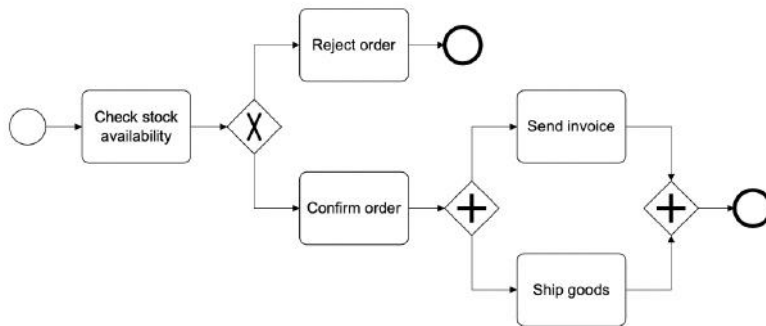
Flow



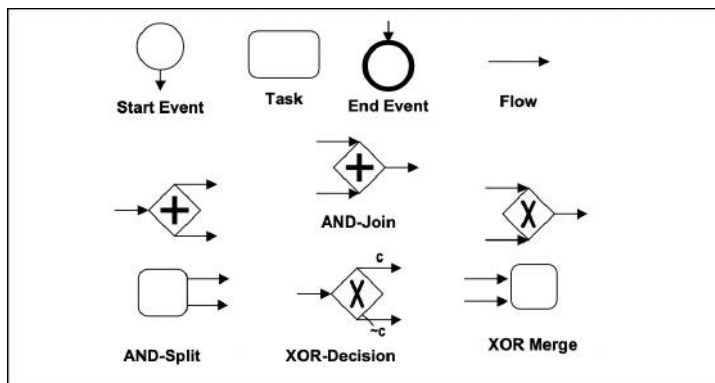
Gateway

Je krijgt BPMN poster op het examen !

Voorbeeld: orderbeheerproces

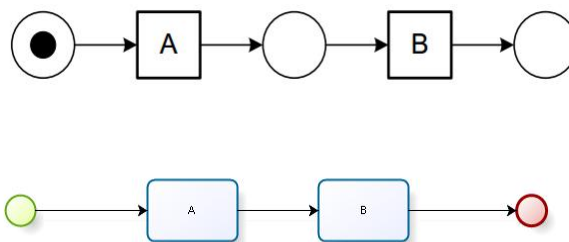


BPMN: basisstroomelementen

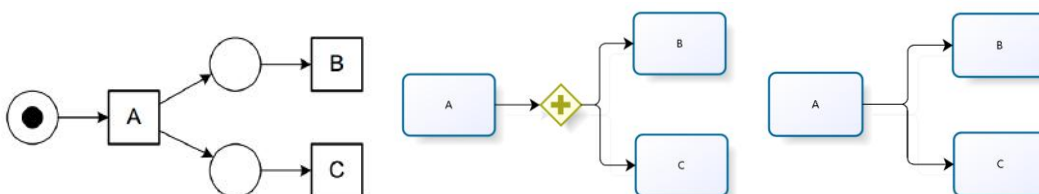


Basisrouteringspatronen

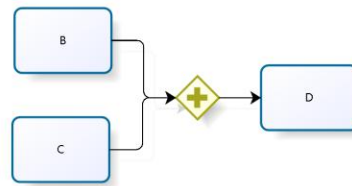
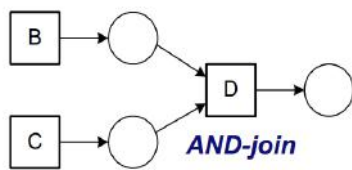
- Sequentieel
 - o "Eerst A dan B" (sequentieel patroon)



- Parallel
 - o "A en B tegelijkertijd of in willekeurige volgorde"
 - o AND-split (parallel splitspatroon)

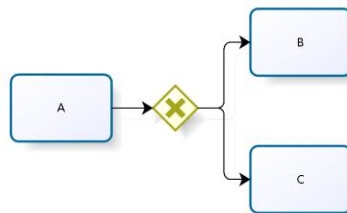
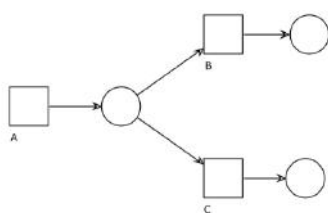


- AND-join (synchronisatiepatroon)



- Keuze

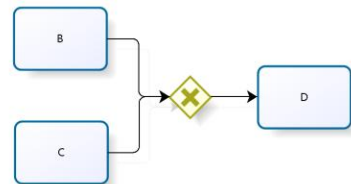
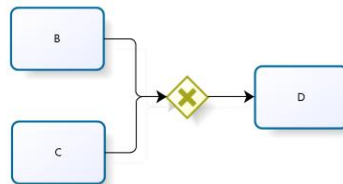
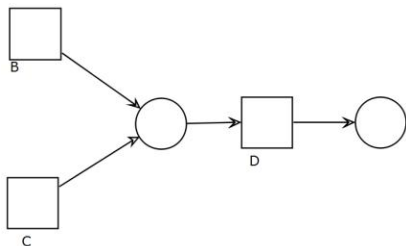
- XOR-split (exclusief keuzepatroon)
- "A of B"



Je MOET de condities duidelijk maken in de model !

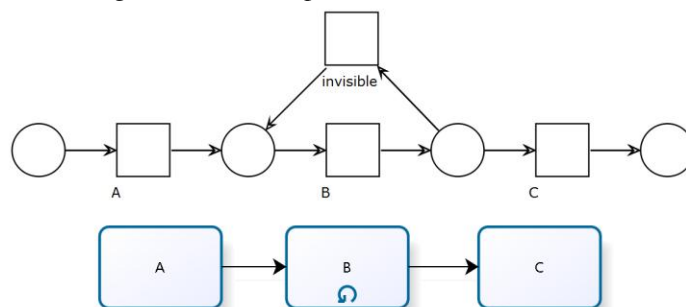
- XOR-join (eenvoudig samenvoegpatroon)

- "D na B of C"



- Iteratie

- "Meerdere A's" (gestructureerd luspatroon)



Andere procesmodelleringstalen:

- Yet Another Workflow Language
- EPCs
 - Event-driven Process Chains
 - Ontwikkeld door A.W. Scheer (ARIS)
 - Oorspronkelijk gebruikt in SAP R/3

Tuisoefening:

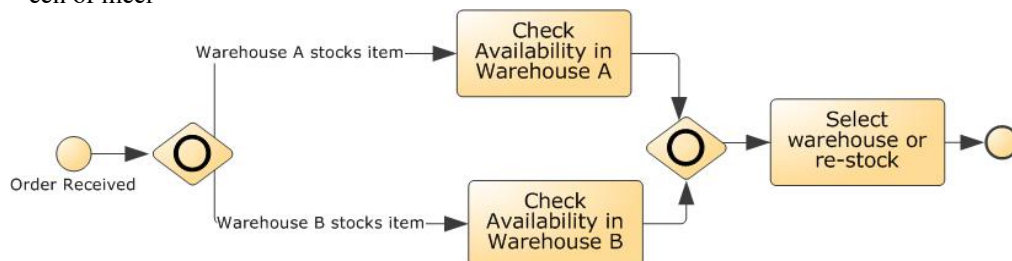
- Teken het Petri net (genaamd PN1) dat geformaliseerd is als: $PN1 = (P, T, F)$ met
 $P = \{\text{start}, p1, p2, p3, p4, \text{end}\}$
 $T = \{A, B, C, D, E\}$
 $F = \{(\text{start}, A), (A, p1), (A, p2), (p1, B), (p2, C), (B, p3), (C, p4), (p3, D), (D, p1), (p3, E), (p4, E), (E, \text{end})\}$
 Opmerking: Je kunt WoPeD gebruiken voor het modelleren van Petri-netten.
- Is Petri net $PN1 = (P, T, F)$ een workflow net? Leg uit.
- Beschouw de volgende markering als beginmarkering: $M0 = [\text{start}]$. Noem drie verschillende mogelijke ontstekingsreeksen (ontstekingsvolgorde = opeenvolging van markeringen). Welke activiteitenreeksen komen overeen met elk van de drie ontstekingsreeksen?
- Beschouw de volgende markering als beginmarkering: $M0 = [\text{start}]$. Hoeveel verschillende ontstekingsreeksen
- Gegeven PN1 met markering $M[p1, p4]$. Welke overgangen zijn ingeschakeld? Welke niet?
- Gegeven PN1 met markering $M[p2, p3]$. Wat is de resulterende markering na het afvuren van D?
- Teken een BPMN-model dat hetzelfde gedrag weergeeft als PN1. Voor het tekenen van het model kun je Signavio gebruiken

2. Process modeling with BPMN: control-flow

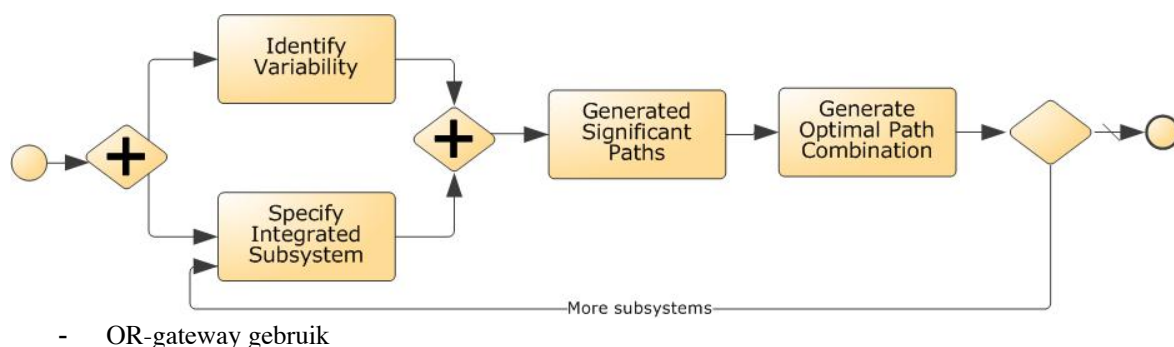
BPMN-gateways

Exclusief (XOR)	Parallel (AND)	Inclusief (OR)
<ul style="list-style-type: none"> - Exclusieve beslissing: neem één tak - Eenvoudige samenvoeging: ga verder als één tak is voltooid 	<ul style="list-style-type: none"> - Parallele splitsing: neem alle takken - Parallel samenvoegen: ga verder als alle inkomende takken voltooid zijn 	<ul style="list-style-type: none"> - Inclusief besluit: neem één of meerdere takken, afhankelijk van de voorwaarden - Inclusief samenvoegen: ga verder als alle actieve inkomende takken zijn voltooid

Voorbeeld: OR gateways “één of meer”



Hoe kunnen we dit model verbeteren?

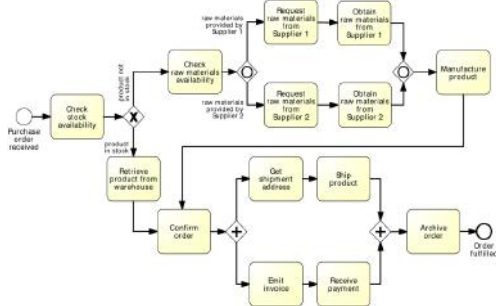


- OR-gateway gebruik

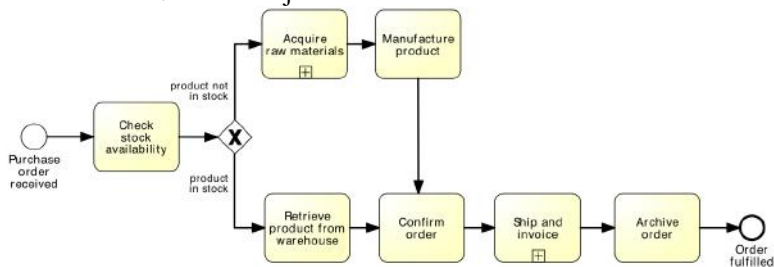
- Bij voorkeur NIET
- als het nodig is ("een of meer"), gebruik het dan op een gestructureerde manier (OR-split + OR-join).

3. Process modeling with BPMN: subprocesses

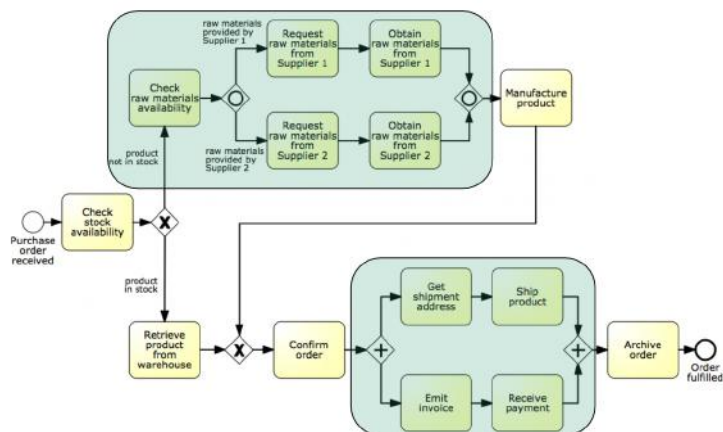
Is er iets mis met deze model ?



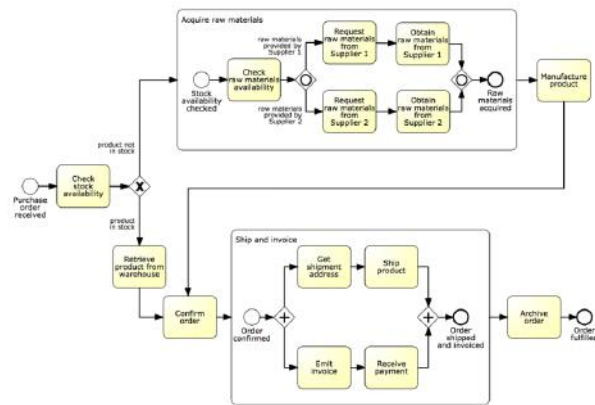
Is dit beter ? Veel duidelijker !



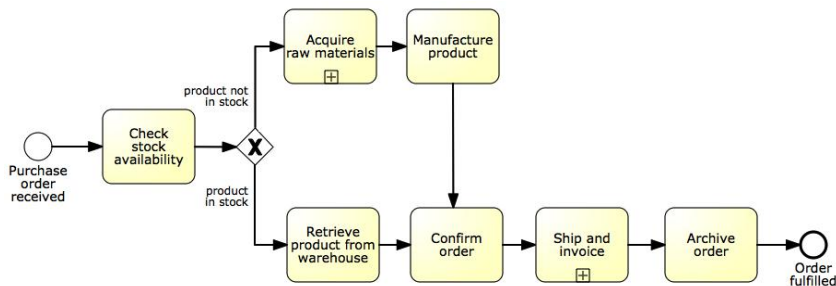
Mogelijke subprocessen identificeren



Modelleer subprocessen in afzonderlijke diagrammen



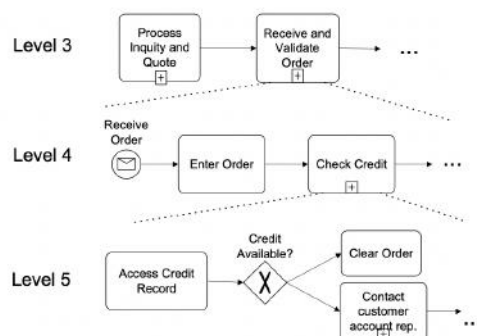
Het geherformuleerde model



Procesdecompositie

- Een activiteit in een proces kan een afzonderlijk (sub)proces "aanroepen".
- Gebruik deze functie om:
 - o grote modellen op te splitsen in kleinere, waardoor ze gemakkelijker te begrijpen en te onderhouden zijn
→ proceshiërarchieën
 - o Gemeenschappelijke fragmenten delen over meerdere processen
→ gedeelde subprocessen
 - o Identificeer delen van een proces die:
 - Herhaald
 - meerdere keren parallel worden uitgevoerd
 - geannuleerd

Proceshiërarchieën

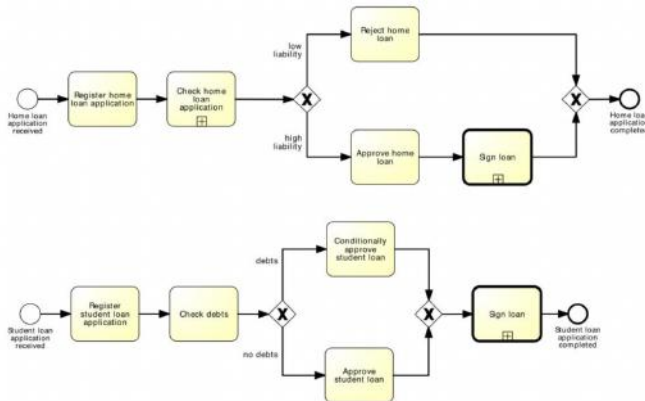


Richtlijnen voor modellering

- Goede praktijk is dat het top-level proces eenvoudig moet zijn (geen of weinig gateways) en de belangrijkste fasen van het proces moet weergeven.
 - o Elke fase wordt dan een subprocess
 - o Dit wordt soms een "waardeketen" genoemd

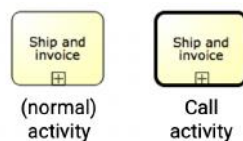
- Wanneer moeten we een procesmodel ontleden in subprocessen?
 - o Wanneer het model te groot wordt:
 - Moeilijk te begrijpen
 - Verhoogde kans op fouten
 - o Vuistregel: niet meer dan 30-50 control-flow objecten (activiteiten, gebeurtenissen, gateways)

Gedeeld subprocess: activiteiten oproepen



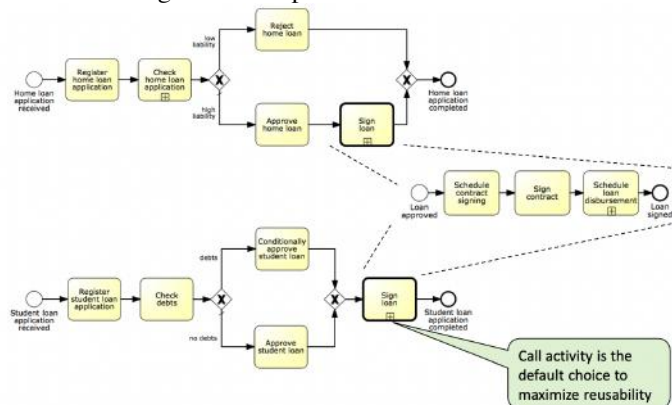
Hergebruik van processen

- Standaard wordt een subprocess "ingebed" in het bovenliggende proces (d.w.z. het wordt opgeslagen in hetzelfde bestand).
- Voor maximaal hergebruik is het mogelijk het subprocess "uit te pakken" en als afzonderlijk bestand op te slaan in het procesmodelarchief.



- Een dergelijk subprocess wordt "globaal" model genoemd, en wordt aangeroepen via een "aanroep"-activiteit (weergegeven met een dikkere rand)

Voorbeeld: hergebruik van processen

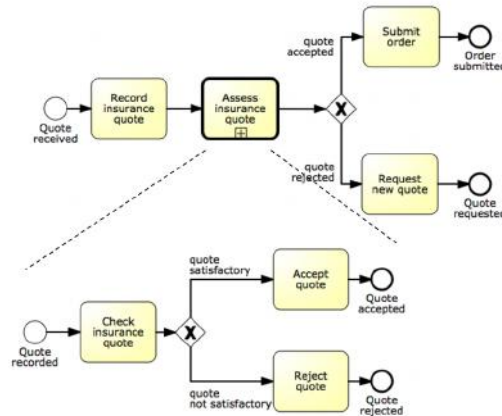


Syntactische regels voor subprocessen

- Start met ten minste één startgebeurtenis
 - o Als er meerdere zijn, zal de eerste gebeurtenis het subprocess starten
- Finish met ten minste één eindgebeurtenis
 - o Het subprocess wordt voltooid zodra alle tokens een eindgebeurtenis hebben bereikt. Kan een (X)OR-splitsing na het subprocess nodig hebben om te begrijpen welke eindgebeurtenis(en) is (zijn) bereikt.
- Sequentiestromen kunnen de grenzen van subprocessen niet overschrijden
 - o Gebruik begin/eind-gebeurtenissen

- Berichtstromen kunnen subprocessgrenzen overschrijden
 - o om berichten aan te geven die uitgaan van/in het subprocess binnenkomen

Voorbeeld: subprocess met meerdere eindgebeurtenissen



4. Process modelling with BPMN: repetition

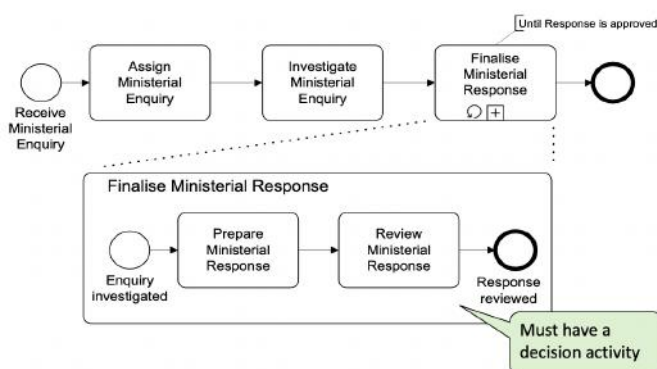
Sequentiële, blokgestructureerde herhaling: activiteitenlus

- In workflownetten worden herhalingen vastgelegd via willekeurige cycli.
- BPMN biedt ook de "Activity Loop"-constructie, waarmee de herhaling van een taak of subprocess mogelijk maakt



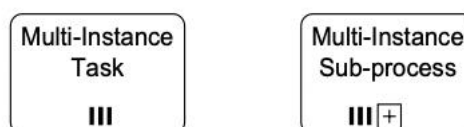
- Het belangrijkste verschil is dat de "Activity Loop" gestructureerd is, terwijl willekeurige cycli ongestructureerd kunnen zijn.
- Suggestie: gebruik "Activity Loop" als je herhaling gestructureerd is.

Voorbeeld: blokgestructureerde herhaling



Parallele herhaling: activiteit met meerdere instanties

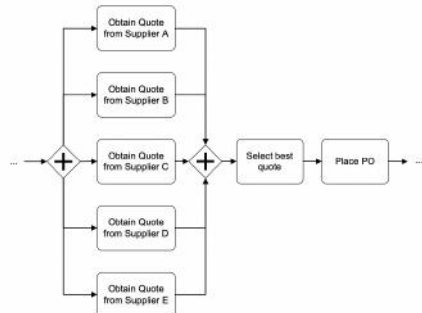
- De multi-instance activiteit biedt een mechanisme om aan te geven dat een activiteit meerdere keren tegelijk wordt uitgevoerd.



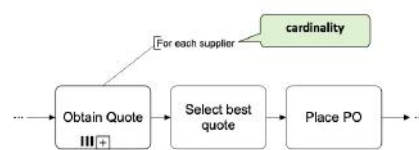
- Nuttig wanneer dezelfde activiteit moet worden uitgevoerd voor meerdere entiteiten of gegevensitems, zoals:
 - o Offertes aanvragen bij meerdere leveranciers
 - o Controleer de beschikbaarheid voor elke regel in een bestelling afzonderlijk
 - o Vragenlijsten verzenden en verzamelen voor meerdere getuigen in het kader van een verzekeringsdekking
- Kunnen ook Horizontale lijnen zijn maar dit is sequentieel ! Verschil met sequentiele pijl kennen !

Voorbeeld: activiteit met meerdere instanties

- Bij een aanbesteding moet doorgaans een offerte worden verkregen van alle voorkeursleveranciers (veronderstelling: er bestaan vijf voorkeursleveranciers). Nadat alle offertes zijn ontvangen, worden zij geëvalueerd en wordt de beste offerte geselecteerd. Vervolgens wordt een bijbehorende inkooporder geplaatst



Oplossing: met multi-instance activiteit



Ongecontroleerde herhaling: Ad-hoc subprocess

- Het ad-hoc subprocess bevat activiteiten (taken of subprocessen) die in een willekeurige volgorde en tijd worden uitgevoerd.
- Mag volgorde van subset van activiteiten bepalen door sequentiestroom. (door een pijl toevoegen)
 - o “Het moet minstens één keer gebeurd worden”
- Kan worden gebruikt in een vroege versie van een procesdiagram wanneer de volgorde van uitvoering nog onbekend is.
- Geduid met een tilde markering. (zie afbeelding)

BPM3. Advanced Process Modelling

1. Process modelling with BPMN: events

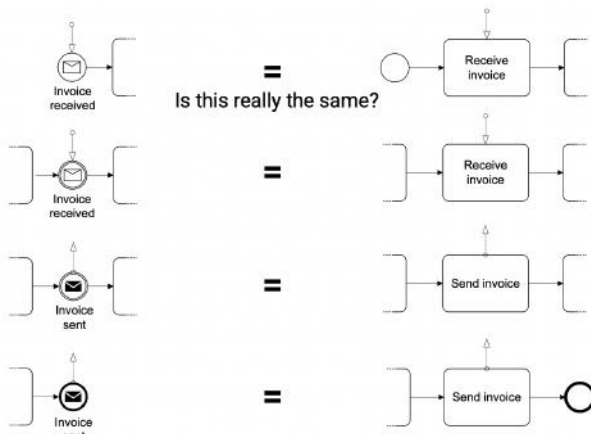
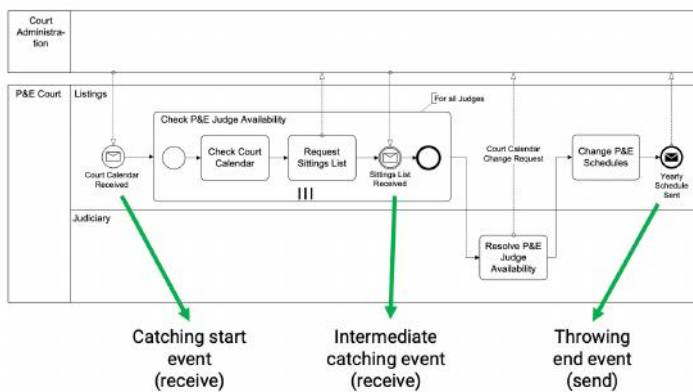
BPMN events

a) Event types (Groen aangeduide events moet je kennen)

= passieve, ogenblikkelijk momenten in de model: signalen, berichten..

- Start vs. intermediate vs. end
 - o Start: dunne cirkel
 - o Intermediate: dubbele cirkel
 - o End: dike cirkel
- Typed vs. untyped
 - o Untyped: je specificeert het type niet, dus geen tekeningen
- Throwing vs. catching
 - o Catching: witte tekeningen → signaal in het model krijgen
 - o Throwing: Zwart tekeningen → signaal buiten het model sturen
- Interrupting vs. non-interrupting
 - o Interrupting:
 - o Non-interrupting: stippenlijn (nooit daarmee beginnen behalve soms in subprocessen)

Example: message events (heel popular)



Events

	Standard	Start	Event Sub-Process Interrupting	Event Sub-Process Non-Interrupting	Catching	Intermediate	Boundary Interrupting	Boundary Non-Interrupting	Throwing	End
None: Untyped events, indicate start point, state changes or final states.	○								○	○
Message: Receiving and sending messages.	✉	✉	✉	✉	✉	✉	✉	✉	✉	✉
Timer: Cyclic timer events, points in time, time spans or timeouts.	🕒	🕒	🕒	🕒	🕒	🕒	🕒	🕒	🕒	🕒
Escalation: Escalating to an higher level of responsibility.		⬆	⬆	⬆	⬆	⬆	⬆	⬆	⬆	⬆
Conditional: Reacting to changed business conditions or integrating business rules.	📋	📋	📋	📋	📋	📋	📋	📋	📋	📋
Link: Off-page connectors. Two corresponding link events equal a sequence flow.					➡				➡	
Error: Catching or throwing named errors.	⚡	⚡	⚡	⚡	⚡	⚡	⚡	⚡	⚡	⚡
Cancel: Reacting to cancelled transactions or triggering cancellation.						✖				✖
Compensation: Handling or triggering compensation.		⏪			⏪	⏪	⏪	⏪	⏪	⏪
Signal: Signalling across different processes. A signal thrown can be caught multiple times.	⚡	⚡	⚡	⚡	⚡	⚡	⚡	⚡	⚡	⚡
Multiple: Catching one out of a set of events. Throwing all events defined.	⬢	⬢	⬢	⬢	⬢	⬢	⬢	⬢	⬢	⬢
Parallel Multiple: Catching all out of a set of parallel events.	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Terminate: Triggering the immediate termination of a process.										⬤

Vergelijking met verzend-/ontvangsttaken

Dus, wanneer gebruik je wat?

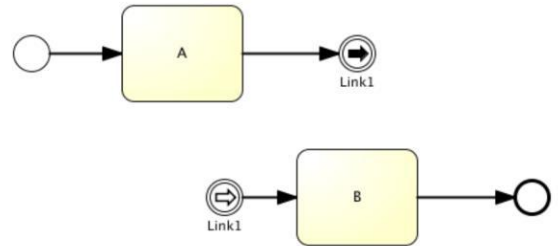
→ Gebruik berichtgebeurtenissen alleen wanneer de overeenkomstige activiteit gewoon een bericht zou verzenden of ontvangen en verder niets zou doen.

→ Als het meer dan enkele seconden duurt : TASK

→ Een event moet ogenblikkelijke momenten zijn.

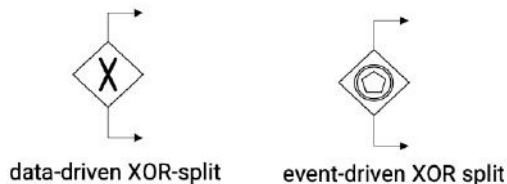
Voorbeeld: link gebeurtenissen

- Zeer eenvoudig gedrag
- Echter, vaak misbruikt!
 - o Modelleren van slangen
 - o Onjuist gebruikt om "onmogelijke" besturingsstromen te modelleren



b) Event-based decisions

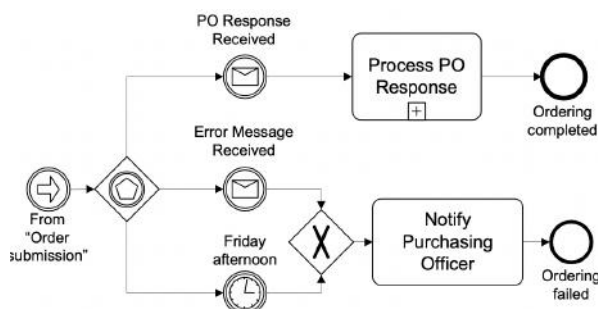
- Bij de XOR-split gateway wordt een vertakking gekozen op basis van voorwaarden die over de beschikbare gegevens worden geëvalueerd
→ De keuze kan onmiddellijk worden gemaakt nadat de token van de inkomende stroom binnenkomt.
- Soms moet de keuze worden uitgesteld tot een gebeurtenis plaatsvindt
→ De keuze is gebaseerd op een "race tussen gebeurtenissen".
- Daarom onderscheidt BPMN gegevensgestuurde en gebeurtenisgestuurde XOR-splitsingen.
- Moet gevolgd worden door een catching interrupting events !



Voorbeeld: op gebeurtenissen gebaseerde beslissing

Restaurants dienen elke donderdag bestellingen in bij leveranciers om hun voedselvoorraden aan te vullen. Het proces om een bestelling af te ronden begint wanneer het restaurant een "PO Response" of een foutmelding ontvangt. Het kan echter ook gebeuren dat er helemaal geen antwoord wordt ontvangen. Als op vrijdagmiddag geen antwoord is ontvangen of als er een foutmelding is ontvangen, moet de inkoper op de hoogte worden gebracht. Anders wordt de PO Response normaal verwerkt.

Mogelijke oplossing :



c) Boundary events

- Soms kan zich tijdens de uitvoering van een subproces een gebeurtenis voordoen die actie vereist...
- Dergelijke gebeurtenissen worden geplaatst aan de grenzen van het subproces (boundary events)
- Twee smaken:
 - o Onderbrekende boundary events
 - o Niet-onderbrekende boundary events

- Triggeren een taak/subproces **parallel** aan de normale stroom, d.w.z. zonder de normale stroom te onderbreken.
- Dubbele stippelijijn



→ Niet-onderbrekende boundary events

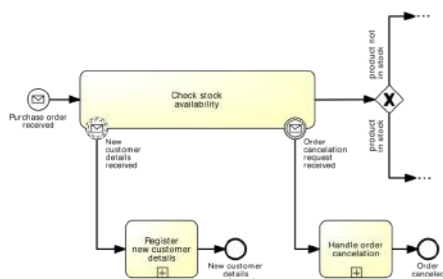
Soms moeten we een taak/subproces parallel aan de normale stroom activeren, d.w.z. zonder de normale stroom te onderbreken.

Dit kan worden bereikt door niet-onderbrekende grensgebeurtenissen te gebruiken: - -

- Dubbele stippelijijn
- Vastgemaakt aan de grens van een taak/subproces

(Als het geen stippelijijn is --> niet tegelijkertijd)

Boundary events: voorbeeld

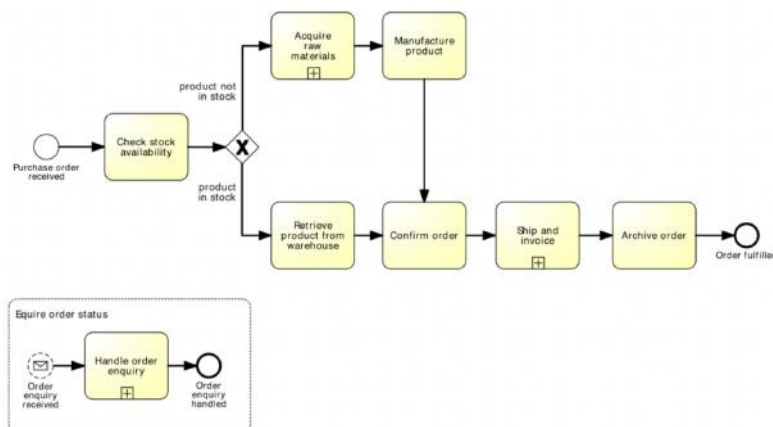


De twee boundary effect kunnen niet beginnen zolang check stock availability is begonnen

d) Eventsub-processes

- Een event-subproces is een proces gekoppeld aan een ouderproces, dat wordt geactiveerd wanneer een event plaatsvindt.
- Alternatief voor het plaatsen van een niet-onderbrekende grensgebeurtenis rond het bovenliggende proces

Event sub-process voorbeeld



Kan een stippelijijn hebben of niet !
Hier in dit geval een stippelijijn. Dus non-interrupting

2. Process modelling with BPMN: exception handling

Uitzonderingsbehandeling in BPMN

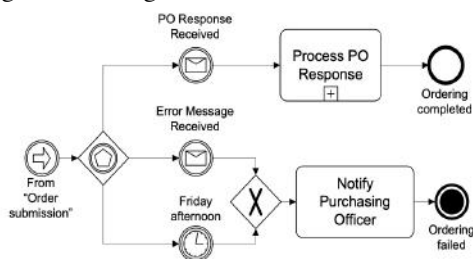
a. Beëindigingsgebeurtenis/ Terminate events

- Uitzonderingen zijn gebeurtenissen die afwijken van het "normale" verloop van een proces.
- De eenvoudigste vorm van een uitzondering is melden dat er een uitzondering is (negatief resultaat)
- Dit kan worden gedaan via de Terminate end event: deze dwingt het hele proces af te breken ("veegt" alle achtergebleven tokens weg, indien aanwezig)



Voorbeeld: beëindig gebeurtenis

- Signaleer de negatieve uitkomst...



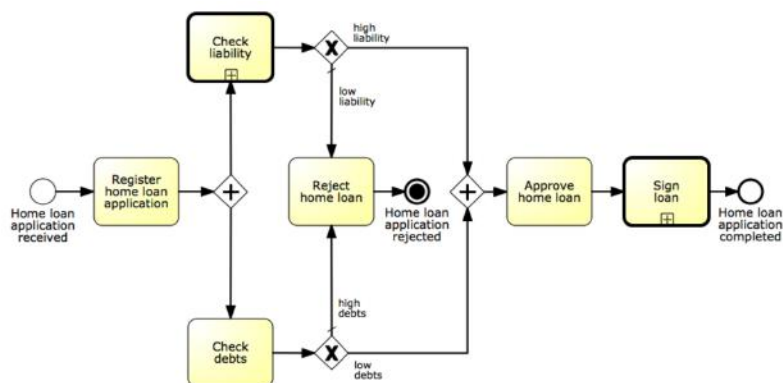
Negatieve uitkomsten met terminate event

- Is het altijd nodig om negatieve uitkomsten aan te geven met een terminate event?
→ Nee, er is geen verplichting in BPMN.
→ Dit is een best practice, meer uit pragmatisch oogpunt, om de modellen begrijpelijker te maken

Ook andere getypeerde events kunnen worden gebruikt om negatieve uitkomsten aan te geven (bv. Error)

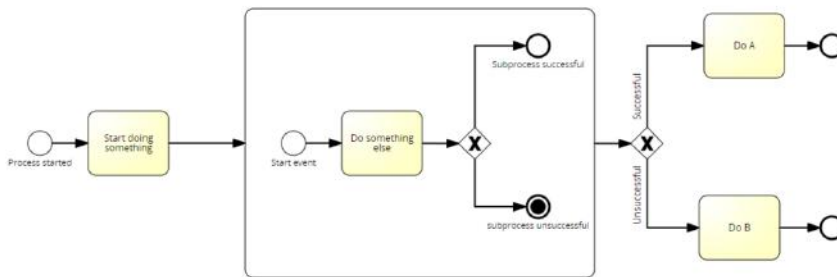
Voorbeeld 2: afsluit event

- Het proces afbreken door alle tokens te verwijderen...



(Hier zijn er twee shared-subprocess)

Het wissen van functionaliteit en hiërarchieën



Het wissen van functionaliteiten werkt enkel downwards en niet upwards.

Hier is er een terminate in de subprocess maar niet in de parentprocess.

b. Uitzonderingsafhandeling met boundary events

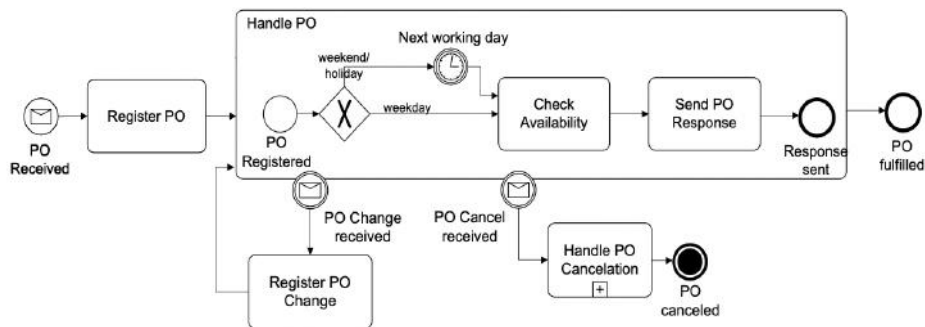
- Voor het afhandelen van uitzonderingen moet vaak een activiteit of subprocess worden gestopt en een speciale activiteit worden uitgevoerd.
- Soorten uitzonderingen voor een activiteit (taak/subproces) in BPMN:
 - o Time-out: een activiteit duurt te lang en moet worden onderbroken: afgehandeld met het Timer event
 - o Extern: er gaat iets mis buiten het proces, en de uitvoering van de huidige activiteit moet worden onderbroken. Afgehandeld met de Message event
 - o Intern: er gaat iets mis binnen een activiteit, waarvan de uitvoering dus onderbroken moet worden. Afgehandeld met de gebeurtenis Fout
- Al deze gebeurtenissen stoppen de omringende activiteit en starten een exception handling routine (ze zijn dus interrupting)

Error events / foutgebeurtenissen

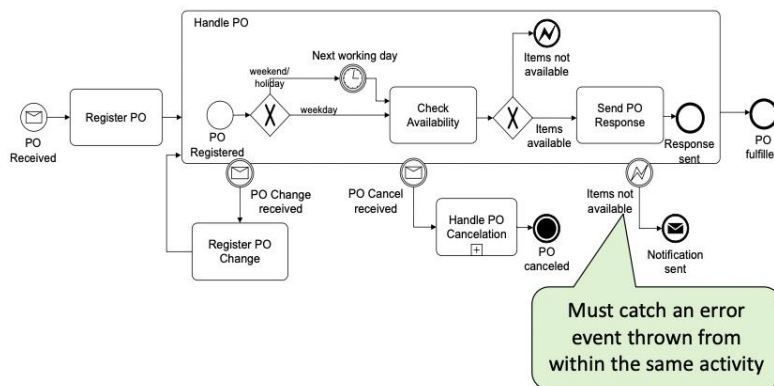
- Uitzonderingen zijn gebeurtenissen die een proces van zijn "normale" koers doen afwijken
- Afhandeling van uitzonderingen betekent vaak het stoppen van een subprocess en het uitvoeren van een speciale activiteit.
- Bereikt met behulp van twee event nodes:
 - o Een "end error event" die de uitvoering van het omsluitende subprocess stopt.
 - o Een "tussentijdse foutgebeurtenis" die aan het omsluitende subprocess is gekoppeld - dit is waar de procesuitvoering wordt voortgezet na de fout.

Voorbeeld: foutgebeurtenis

Beschouw het onderstaande voorbeeld "Verwerking PO" met de volgende uitbreiding: als een artikel niet beschikbaar is, moet elke verwerking met betrekking tot de PO worden gestopt. Daarna moet de klant worden geïnformeerd dat de PO niet verder kan worden verwerkt.

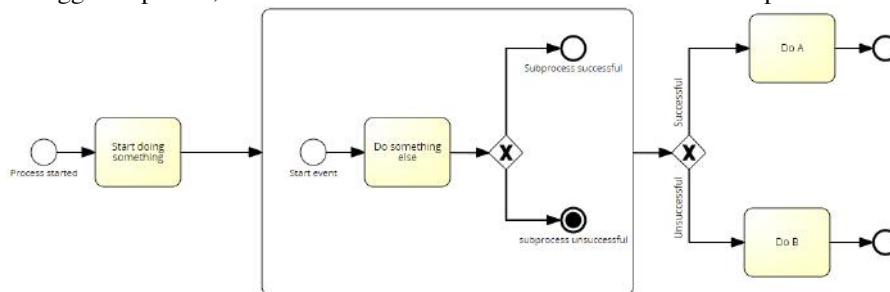


Oplossing



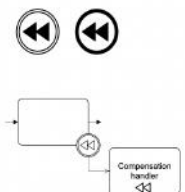
Subproces met meerdere uitkomsten

- Ofwel je gebruikt grensgebeurtenissen (zie hierboven)
- Of je gebruikt een of meer Terminate events, maar er moet een XOR-gateway worden toegevoegd in het bovenliggende proces, met de data conditions die de uitkomst van het subprocess weergeven.



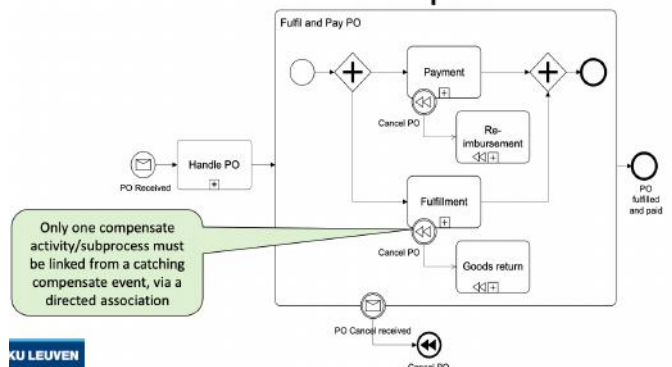
c. Compensatie afhandeling

- Kan worden gebruikt als onderdeel van een uitzonderingsafhandelingsprocedure
- Terugdraaien van voltooide procesactiviteiten
 - o Bijv: Boekingen voor vakanties zijn succesvol afgerond. Het blijkt echter dat de opgegeven creditcardgegevens onvolledig waren. Een boekingsherroeping is noodzakelijk.
- Getriggerd door het gooien van Compensate Event
- Compensation Handler (geactiveerd door een gevangen Compensate event) voert de rollback uit



Voorbeeld: compensatiegebeurtenis

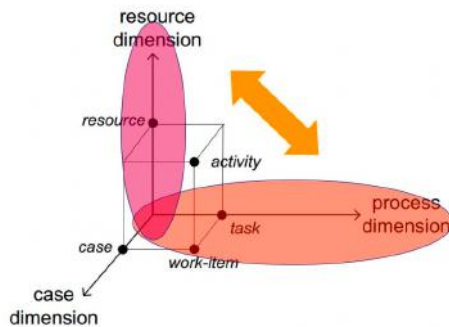
Nadat een PO is geregistreerd, gecontroleerd en een antwoord is teruggestuurd, worden parallel een subproces voor betaling en een subproces voor uitvoering gestart. Tijdens deze twee subprocessen kan een PO-annulering worden ontvangen van de klant. In dat geval worden beide subprocessen gestopt, en wordt voor elk ervan een compensatieroutine uitgevoerd (bv. terugbetaling en/of terugzending van goederen).



=> Mogelijke oplossing: compensatie-evenement

3. Resource modelling

Process dimensions



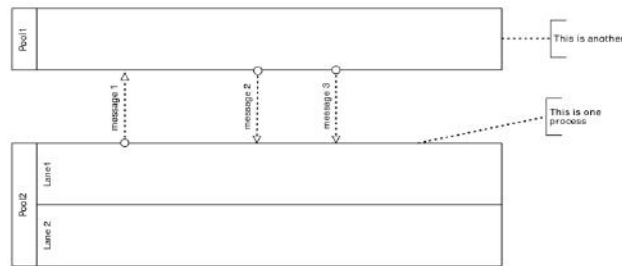
⇒ Resources zijn een heel belangrijk dimensie voor processen.

Organisatorische elementen in procesmodellen

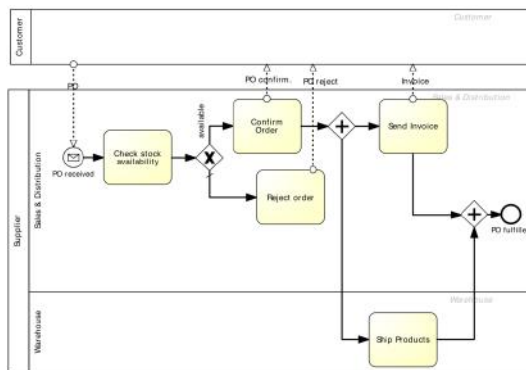
- Abstracties maken
 - o Hulpbron/ Ressource : Menselijke actor of apparatuur (bijv. printer) die nodig is om een activiteit uit te voeren
 - o Hulpmiddelenklasse / Resource class : Set van middelen met gedeelde kenmerken, bijv. bediende, manager, verzekeringsagent
- Een middelenklasse kan een:
 - o Rol (vaardigheid, competentie, kwalificatie) Classificatie gebaseerd op wat een middel kan doen of wat van een middel wordt verwacht.
 - o Groep (afdeling, team, kantoor, organisatorische eenheid) Classificatie gebaseerd op de structuur van de organisatie.

Resource-modellering in BPMN

- In BPMN worden resource-classes vastgelegd met behulp van:
 - o Pools - onafhankelijke organisatorische entiteiten
 - bijvoorbeeld - Klant, leverancier, KU Leuven, UZ Leuven,
 - o Lanes - resource classes in dezelfde organisatorische ruimte en met gemeenschappelijke systemen.
 - Verkoopafdeling, marketingafdeling
 - Medewerker, manager, ingenieur



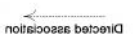
Orderbeheerproces met banen



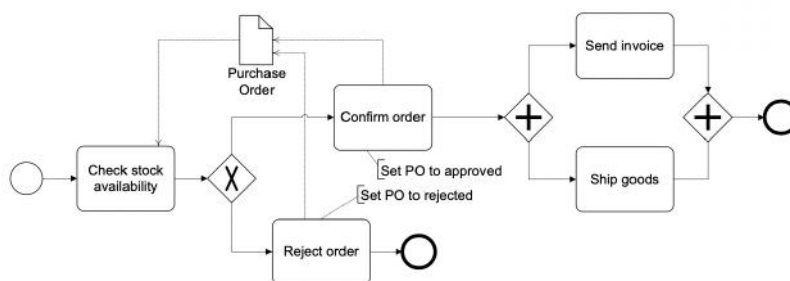
4. Data modelling

BPMN Informatie-artefacten

- Data Objects zijn een mechanisme om aan te geven hoe gegevens nodig zijn of geproduceerd worden door activiteiten.
 - o Worden afgebeeld door een rechthoek waarvan de rechterbovenhoek is omgevouwen.
 - o Representeren input en output van een procesactiviteit.
 - Data Stores zijn containers van dataobjecten die langer dan de duur van een procesinstantie moeten worden bewaard.
 - Associaties worden gebruikt om artefacten zoals dataobjecten en data stores te koppelen aan stroomobjecten (bijv. activiteiten)
- ⇒ Maakt vaak het model complex! Het brengt niet veel waarde toe, dus enkel toevoegen als het echt usefull is.



Model voor orderverwerking met artefacten



Modelleren van gegevenslementen

- Pragmatisme vereist: wees spaarzaam met het gebruik van extra data-elementen
- Het is nooit "syntactisch" fout om ze toe te voegen, maar meestal voegt hun opname niet echt waarde toe
- Vuistregel: voeg de gegevenslementen toe die ongebruikelijk of onverwacht zijn.

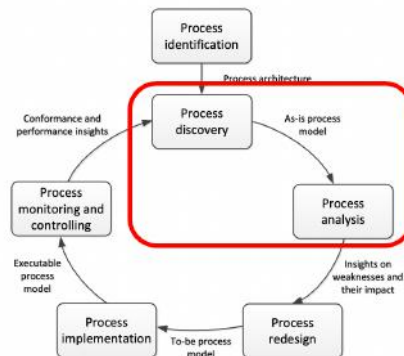
Zie slides voor extra oef !

BPM4. Process Model Verification, Validation and Certification

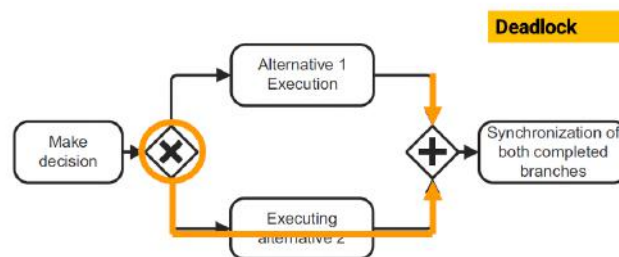
1. Process model quality assurance

BPM lifecycle

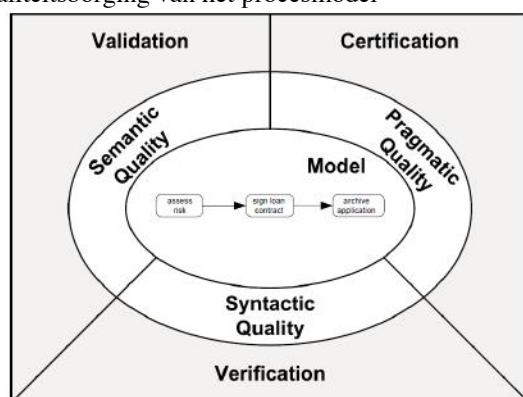
...ontwerpen, analyseren, uitvoeren en controleren van bedrijfsprocessen.



Is dit procesmodel van goede kwaliteit?



Kwaliteitsborging van het procesmodel



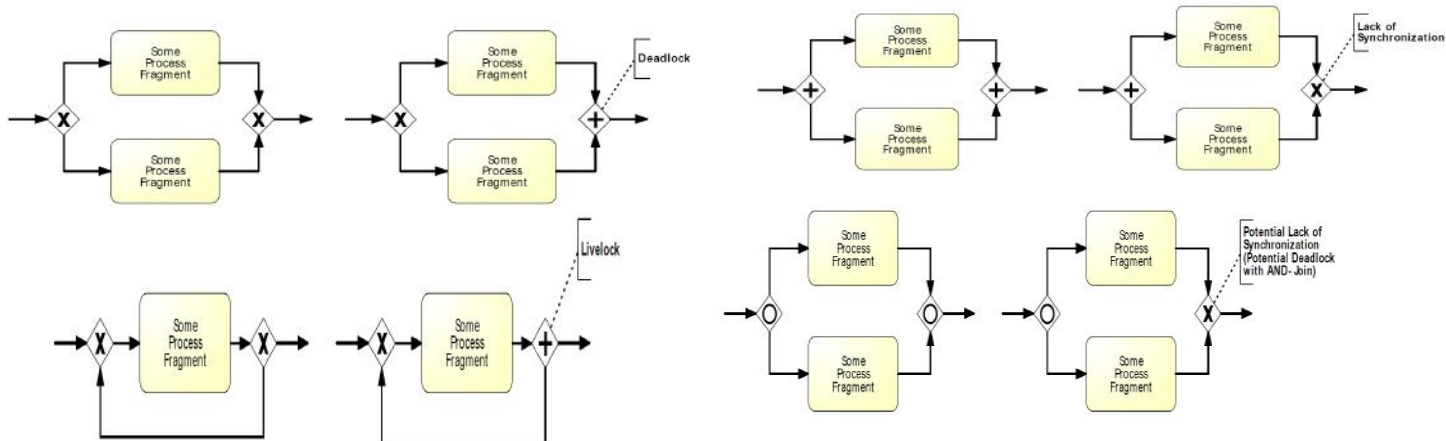
- Semantische
- Pragmatische kwaliteit
- Syntactic kwaliteit

2. Syntactic quality – Verification

Syntactic kwaliteit

- Met syntactische kwaliteit wordt bedoeld of de syntaxis, d.w.z. de regels van de modelleertaal, correct wordt gevolgd.
- Twee soorten problemen:
 - o Gedragsproblemen: hebben betrekking op de wijze waarop control-flow constructen (d.w.z. gateways) ongepast worden gecombineerd.
Bijv. impasse
 - o Structurele problemen: directe fouten tegen de structuurregels van de modelleertaal (verkeerd/onjuist gebruik van elementen)
Bv. activiteit zonder inkomende en/of uitgaande stroom

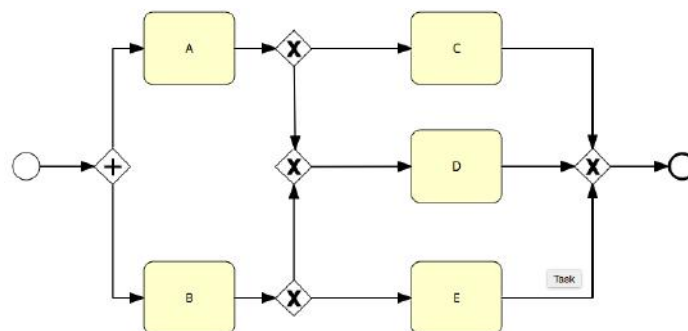
Gedragsmatige syntactische kwaliteitsproblemen



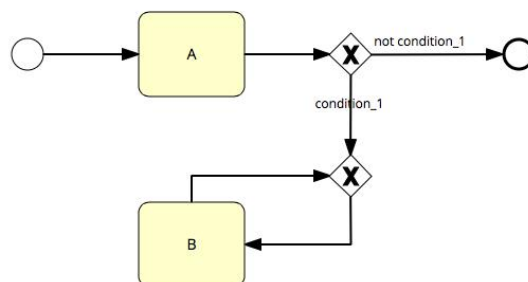
Structural correctness + behavioral correctness

Voorbeeld van gebrek aan synchronisatie

Een loper wordt overal in het model achtergelaten na het bereiken van de eindmarkering/eindgebeurtenis

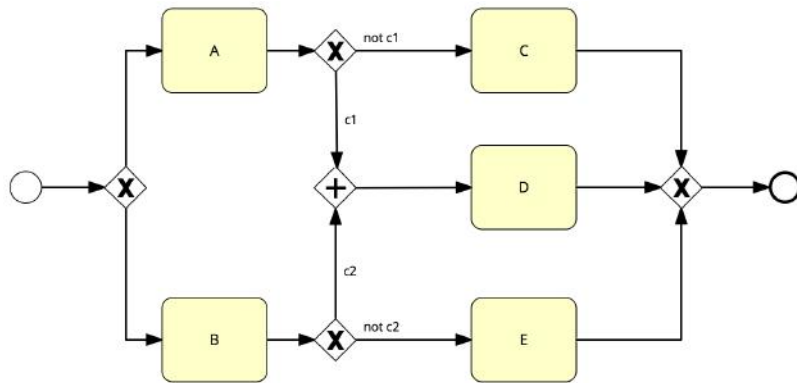


Voorbeeld livelock



Merk op dat dit model ook structureel onjuist is omdat B zich niet op een (gericht) pad naar de eindgebeurtenis bevindt.

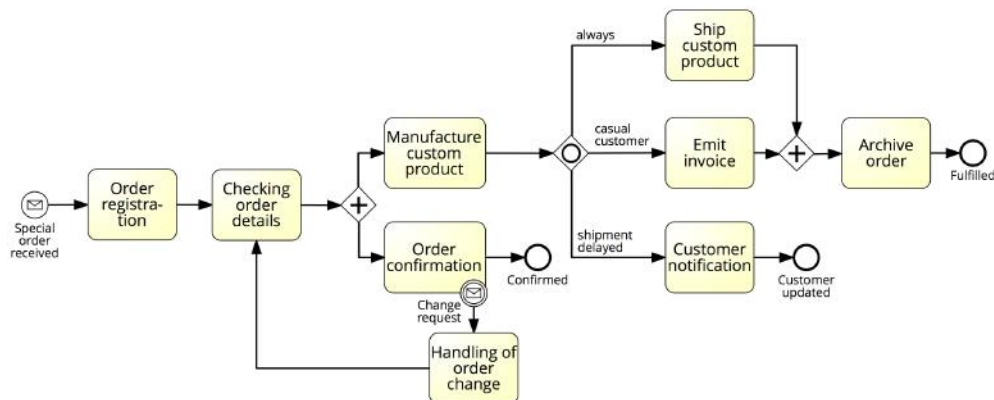
Voorbeeld Deadlock



Het is niet omdat jouw model in sommige situaties wel correct uitvoert dat het geen deadlock is. Zodra er één situatie slecht kan lopen is het een model met een deadlock

Oefening

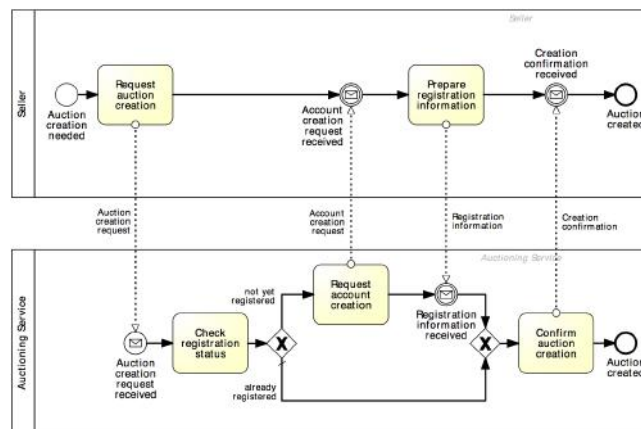
Welke gedragsregels worden geschonden in het onderstaande model?



→Lack of synchronization: bij de change request is het een and split. Je gaat het blok telkens herhalen waardoor je het bovenste stuk ook telkens gaat herhalen. Dus het bovenste stuk is niet gesynchroniseerd met het onderste stuk.

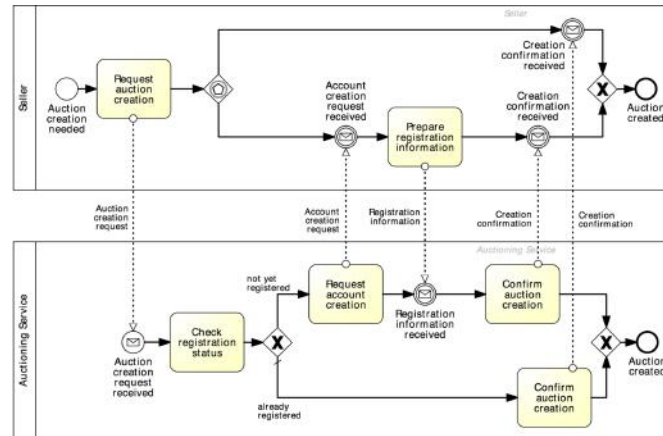
→Deadlock

Kijk uit! Niet alleen vanwege gateways

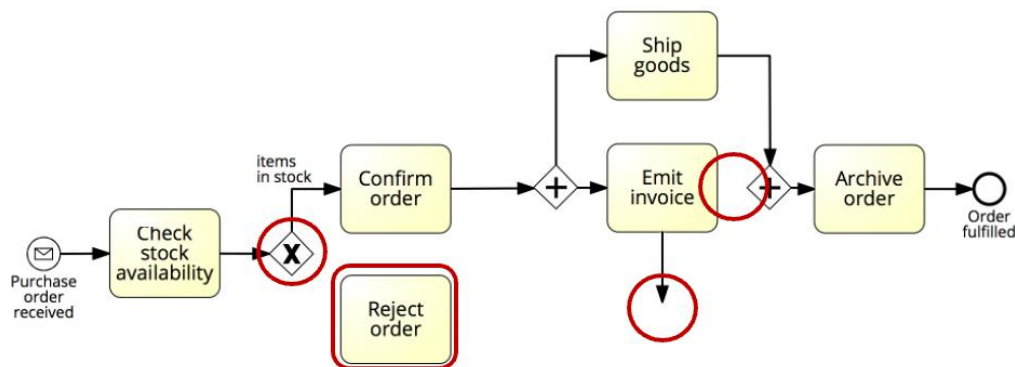


Er is een deadlock want stel: already registered dan gaat het event boven nooit getriggered worden. En ga je nooit door het bovenste process door.

Oplossing

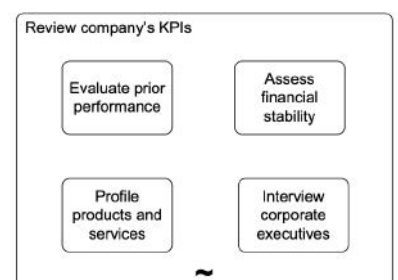


Structurele syntactische kwaliteitsproblemen

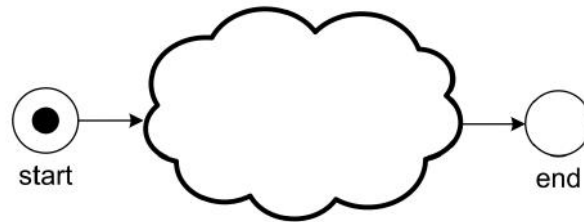


Modellering en syntaxis: een paar herinneringen

- Subprocessen:
 - o Altijd minstens één startgebeurtenis en minstens één eindgebeurtenis nodig
 - o Sequence flows kunnen de grenzen van subprocessen niet overschrijden
 - o Berichtstromen kunnen subprocesgrenzen overschrijden (wanneer je ze zou modelleren als een uitgebreid subproces)
- Ad-hoc subprocessen:
 - o Hebben geen start- of eindgebeurtenis nodig!
- Gebeurtenisgestuurd vs. gegevensgestuurd XOR
 - o Event-driven: op elke uitgaande boog moeten events volgen
 - o Datagestuurd: Voeg ALTIJD data-condities toe!
- Pools en banen
 - o Ga nooit over poolgrenzen heen met control-flows
 - o Verbind nooit elementen in banen van dezelfde pool met berichtstromen
 - o Elke pool is een apart proces
 - Met minstens één start- en eindgebeurtenis
 - Volledig verbonden!



Verificatie: Deugdelijkheid (soundness) van workflownetten

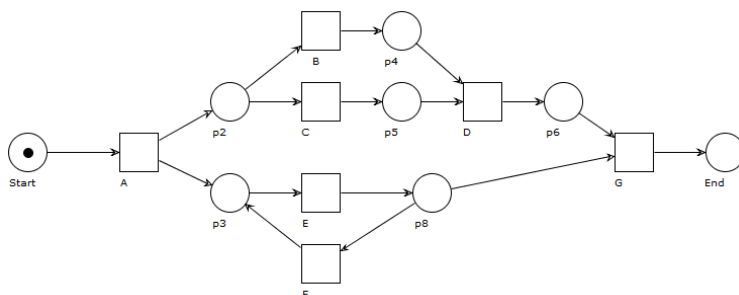


Doel: Elke zaak die via een token op zijn plaats "start" wordt opgestart, wordt succesvol afgesloten met een token op zijn plaats "end"

Drie eisen voor deugdelijkheid (soundness)

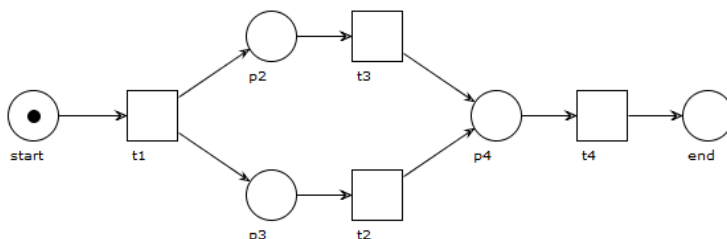
1. Optie om te voltooien
 - Voor elk geval is het altijd nog mogelijk om de toestand te bereiken die net plaats "einde" markeert
2. Juiste voltooiing
 - Als plaats "einde" gemarkeerd is (dus bereikbaar vanaf de eerste markering) zijn alle andere plaatsen leeg voor een gegeven geval.
3. Geen dode overgangen
 - Het moet mogelijk zijn een willekeurige activiteit uit te voeren door de juiste route door het Workflow net te volgen.

Voorbeeld net 1



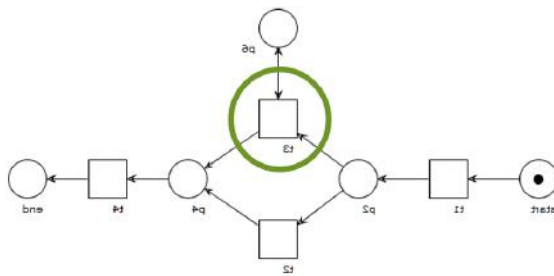
- Optie om te voltooien X
- Of B of C en d heeft 2 inputs nodig
- Juiste voltooiing
- Dode overgangen

Voorbeeld net 2



- Optie om te voltooien V
- Juiste voltooiing X
- Je kan een token op het einde hebben terwijl er nog een token ergens anders in jouw process is
- Dode overgangen

Voorbeeld net 3

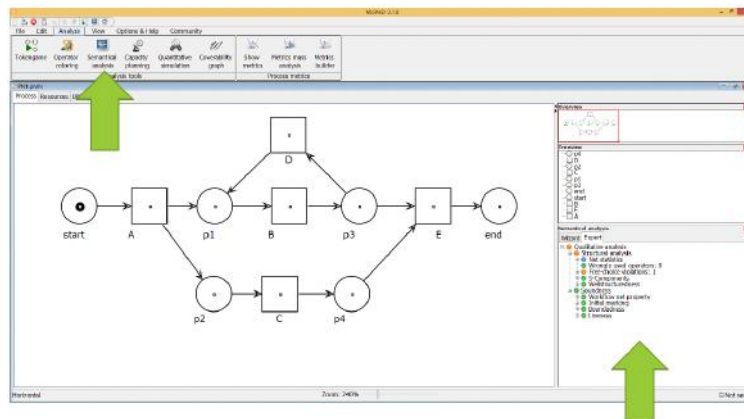


- Optie om te voltooien V
- Juiste voltooiing V
- Dode overgangen X

Deugdelijkheidscontrole/ Soundness verification

- Er zijn verschillende noties van deugdelijkheid gedefinieerd
 - o zwakkere versies, bv. zwakke deugdelijkheid, luie deugdelijkheid
 - o Sterkere versies, bv. klassieke deugdelijkheid, veralgemeende deugdelijkheid.
- De verificatie van de deugdelijkheid van andere soorten modellen wordt vaak bereikt door het model (bv. BPMN of EPC) om te zetten in Petri-netten.
- Geautomatiseerde technieken voor de verificatie van deugdelijkheid zijn zeer volwassen.
 - o Bijv: WoPeD

Soundness verification in WoPeD



3. Semantic quality – Validation

⇒ Hoe goed match mijn model met de realiteit ?

Twee aspecten

- Juistheid / corectheid = validiteit
 - o Hoe kunt u valideren? Bijv. interview of procesontdekking
- Volledigheid
 - o Ontbreken er zaken ?

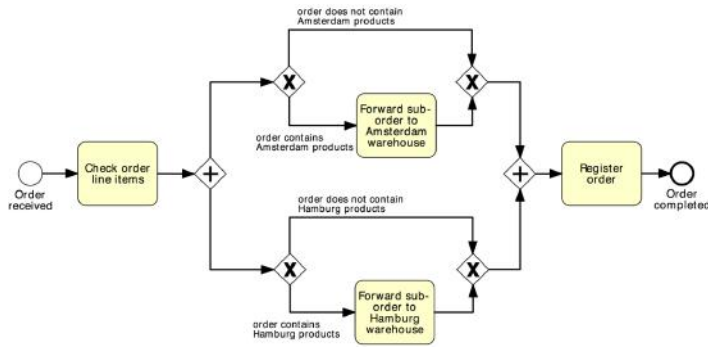
→ A.d.h.v. experts (interview, workshops, processmining...)



Voorbeeld : semantische juistheid

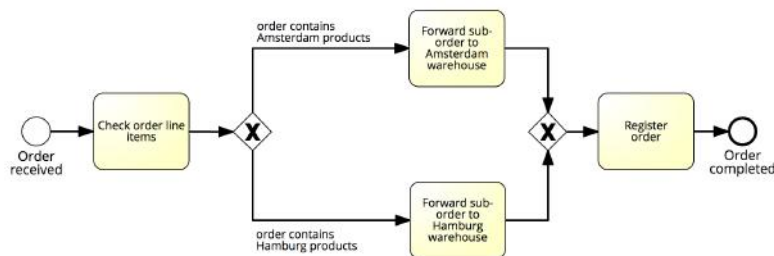
Een bedrijf heeft twee magazijnen waar verschillende producten worden opgeslagen: Amsterdam en Hamburg. Wanneer een order wordt ontvangen, wordt deze verdeeld over deze magazijnen: indien sommige van de relevante producten in Amsterdam worden bewaard, wordt daar een suborder naartoe gestuurd; evenzo, indien sommige van de relevante producten in Hamburg worden bewaard, wordt daar een suborder naartoe gestuurd. Daarna wordt de bestelling geregistreerd en het proces voltooid.

- Wat kunnen we zeggen over de semantische kwaliteit van dit model?



Het is niet mogelijk dat de producten zich noch in het magazijn in Amsterdam noch in het magazijn in Hamburg bevinden. => probleem met juistheid

- En over deze? (met verwijzing naar dezelfde beschrijving)



Bestellingen kunnen zowel producten in Amsterdam als producten in Hamburg bevatten.
⇒ Probleem met compleetheit

Validatietechnieken

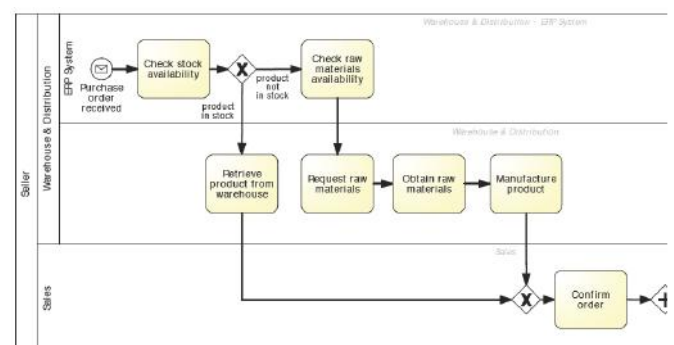
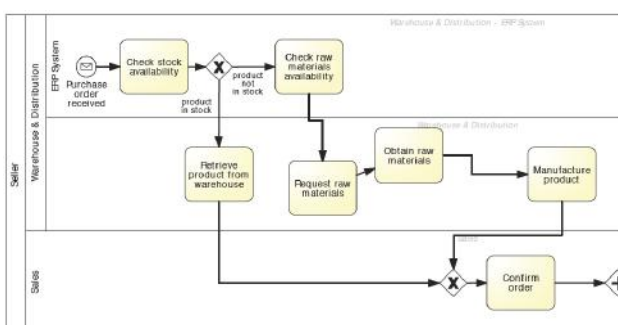
- Interviews
- Workshops
- Proces mining

4. Pragmatic quality – Certification

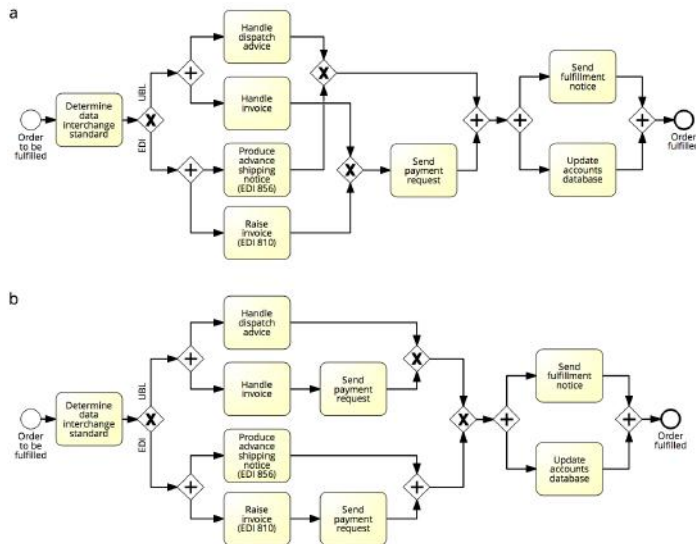
Pragmatische kwaliteit : Het gebruik van het model onderzoeken

- Dimensies
 - o Begrijpelijkheid - Gemakkelijk te lezen
 - o Onderhoudbaarheid - Gemak om veranderingen toe te passen
 - o Leren - Gemak om te begrijpen hoe het bedrijfsproces in de praktijk werkt
- Beïnvloedende factoren
 - o Omvang – groote, aantal events, tasks..
 - o Structurele complexiteit – hoeveel gateways je gebruikt
 - o Grafische lay-out - zo visueel aantrekkelijk mogelijk modelleren

Layout

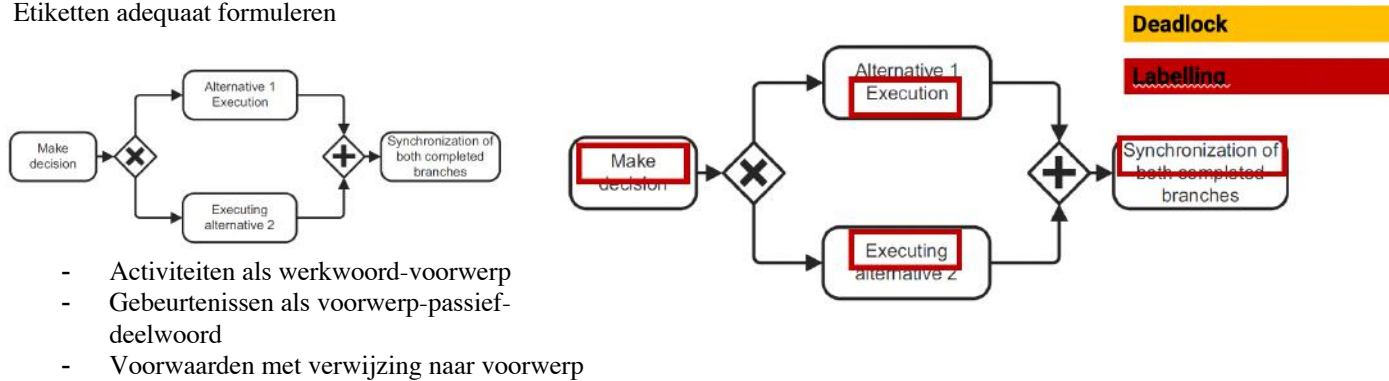


Voorbeeld: Block-structuring



Is dit procesmodel van goede kwaliteit?

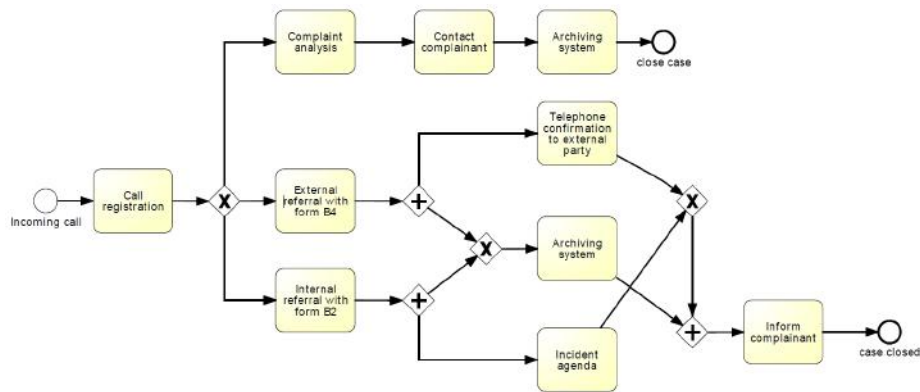
Etiketten adequaat formuleren



Zeven richtlijnen voor procesmodellering (7PMG)

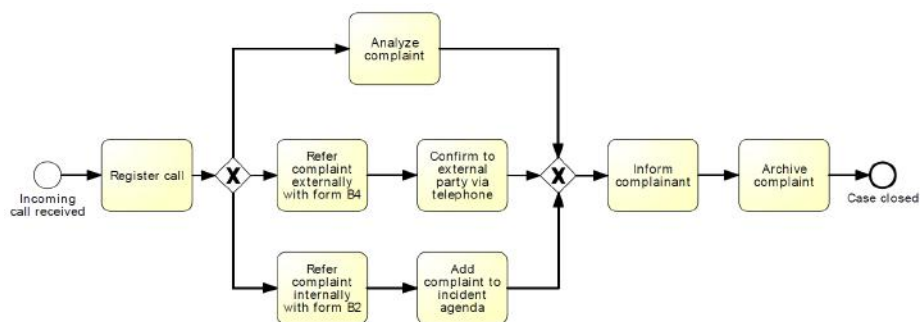
- G1: zo weinig mogelijk elementen in het model gebruiken
- G2: Minimaliseer de routeringspaden per element
- G3: Gebruik één begin- en één eindgebeurtenis
- G4: Modelleer zo gestructureerd mogelijk
- G5: Vermijd OR-routeringselementen
- G6: Gebruik werkwoord-voorwerp activiteit labels
- G7: Ontbind een model met meer dan 50 elementen

Oefening: Leg uit welke 7PMG-richtlijnen wijzen op mogelijkheden voor verbetering. Hervorm het proces op basis van uw observaties

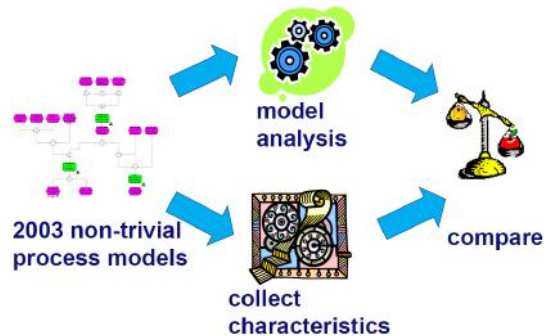


- Meer dan 1 end event
- Activity label (G6)
- Geen correcte blokstructuur (G4)

Het herwerkte proces



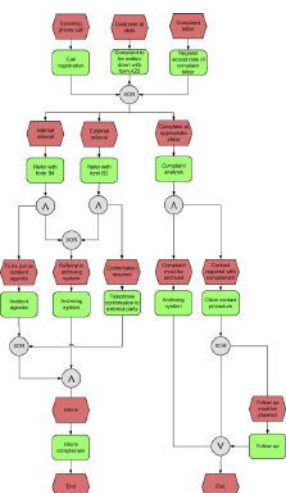
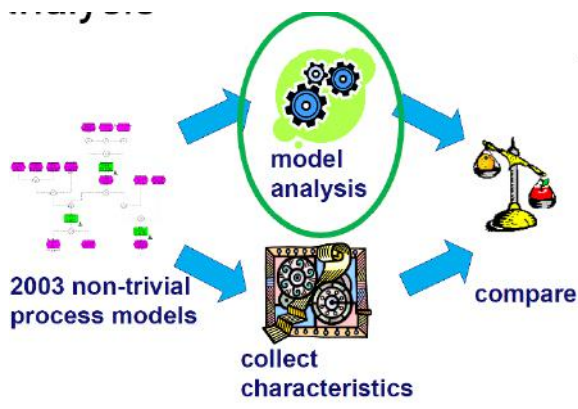
5. Case study: predicting errors in process models

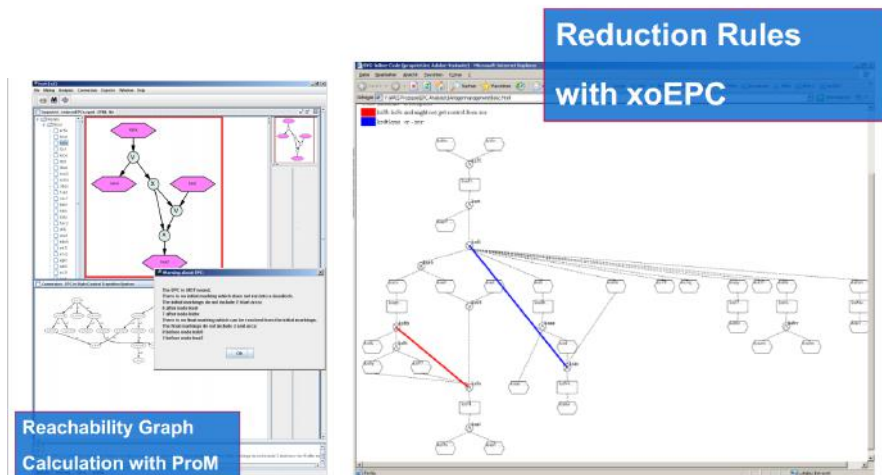


SAP-referentiemodel (anno 2000)

- Het SAP-referentiemodel bevat meer dan 600 niet-triviale procesmodellen (Event-driven Process Chains EPC's).

Model Analyse:

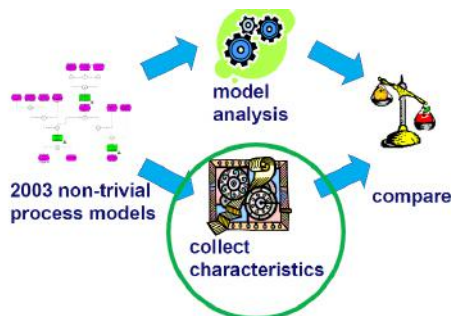




Oplossing : 10,7% bevat syntactische fouten

Parameter	Complete Sample	SAP Ref. Model	Services Model	Finance Model	Consulting Models
xoEPC errors	154	90	28	26	10
Unreduced EPCs	156	103	18	17	18
ProM error EPCs	115	75	16	7	17
EPCs with errors	215	126	37	31	21
EPCs in total	2003	604	381	935	83
Error ratio	10.7%	20.9%	9.7%	3.3%	25.3%

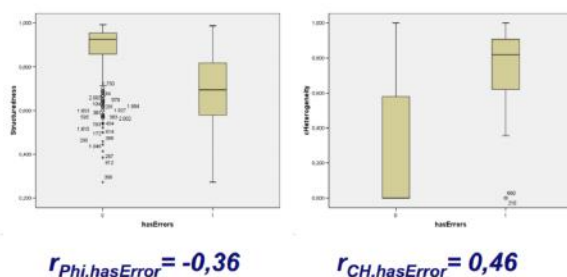
Kenmerken van het model



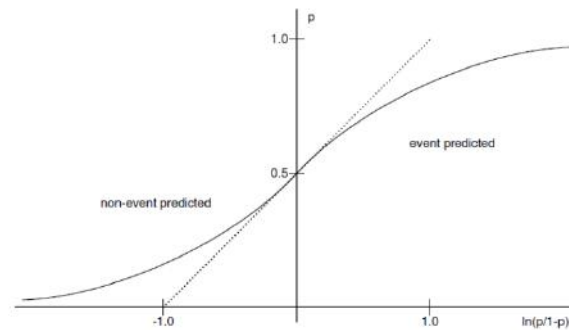
Modelgegevens:

- 28 verschillende statistieken
- Gestructureerdheid ()
- o Vergelijkt gereduceerde en niet-gereduceerde grafiek
- o Meer structuur betekent minder complexiteit
- Connector Heterogeniteit (CH)
- o Entropie over de verschillende connectortypes
- o Meer heterogeniteit betekent meer complexiteit

Correlatie van statistieken en fouten



Het voorspellen van fouten aan de hand van modelkenmerken



Voorspellingsresultaten

- $R^2 = 0,902$
 - o Enkel aan de hand van 2 variabelen !!
- 95% betrouwbaarheid

Observed			Predicted		Percentage Correct
			hasErrors		
Step 9	hasErrors	0	1724	37	97,9
		1	58	155	72,8
Overall Percentage					95,2

Conclusie

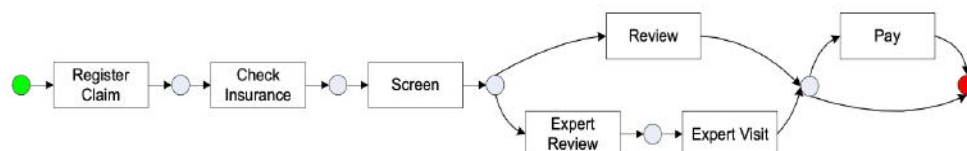
- Verificatie is zeer mature
- Modellen zijn dat niet!
- Syntactische fouten kunnen worden voorspeld op basis van pragmatische kenmerken
 - o Leidde tot richtlijnen over hoe te modelleren (7PMG)
- Hoe te verbeteren?
 - o Modellen serieus nemen, anders geen moeite doen
 - o Training verbeteren
 - o Bouw slimme editors die patronen en antipatronen ondersteunen

BPM5. Declarative Process Modelling

1. Introduction

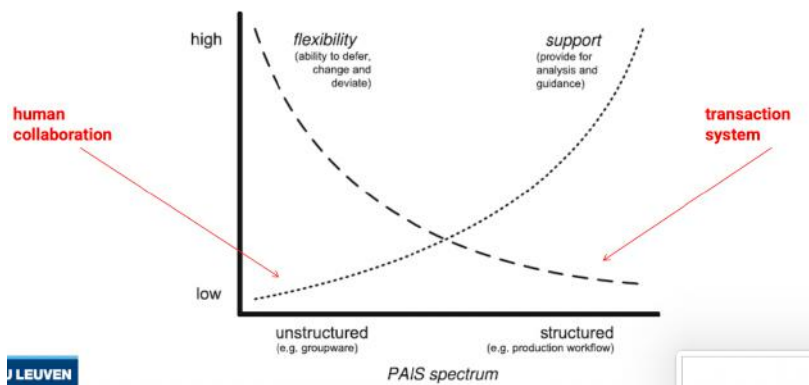
Bedrijfsprocesmodellen

Definitie: Een bedrijfsprocesmodel is een verzameling gerelateerde, gestructureerde activiteiten of taken die een specifieke dienst of product produceren (een bepaald doel dienen) voor een klant of klanten.



Doelstellingen: (1) beschrijvend, (2) prescriptief, (3) verklarend

Evenwicht tussen flexibiliteit en controle

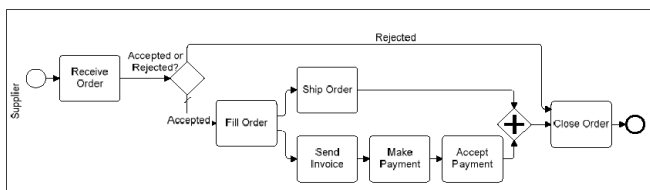


PAIS = Process Aware Information System

⇒ Model: er is flexibiliteit maar we specificeren de limieten

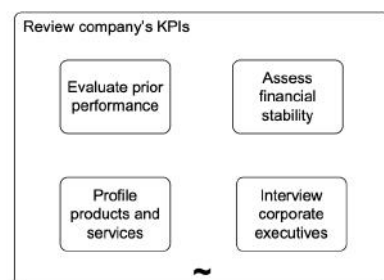
De procedurele benadering van procesmodellering

- Het procedurele of imperatieve modelleerparadigma voor bedrijfsprocessen richt zich op het definiëren van een activiteitenreeks die zal resulteren in het bereiken van het betreffende bedrijfsdoel. Deze activiteitensequenties kunnen gemakkelijk worden weergegeven in grafiektalen (BPMN, Petri Nets, ...).
- Potentiële problemen:
 - Inflexibel, alle uitvoeringspaden moeten worden voorzien
 - Overspecificatie: in praktijk zijn er wegen dat je niet kunt toepassen
 - Onderhoudbaarheid
 - Moeilijkheden bij het aantonen van naleving



Ongecontroleerde herhaling in BPMN: Ad-hoc subprocess

- Het ad-hoc subprocess bevat activiteiten (taken of subprocessen) die in een willekeurige volgorde en tijd moeten worden uitgevoerd.
- Kan volgorde van sub-set van activiteiten definiëren door sequentie flow
- Kan gebruikt worden in een vroege versie van een procesdiagram wanneer de volgorde van uitvoering nog onbekend is
- Aangegeven met een tilde markering



Procedurele versus declaratieve benadering van bedrijfsprocesmodellering

- Het declaratieve paradigma voor bedrijfsprocesmodellering richt zich op het vastleggen en definiëren van regelgevende of interne richtlijnen in beperkingen, bijv. volgorde-relaties tussen de verschillende activiteiten.
- Met een minimale specificatie van de relevante business concerns wordt enige vrijheid gelaten voor de exacte volgorde van de activiteiten. (tegengestelde van procedurele bedrijfsprocessmodellering)

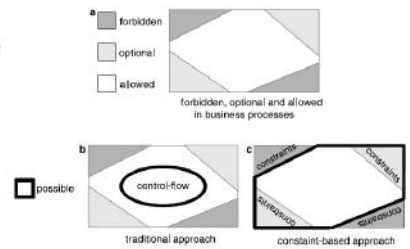
	Procedural modeling	Declarative modeling
Business concerns	implicit	explicit
Execution scenario	explicit	implicit
Execution mechanism	state-driven	goal-driven
Modality	what <i>must</i>	what <i>must, ought, can</i>
Rule enforcement	procedural (what, when, how)	declarative (what)
Communication	explicit (how)	implicit (what)

2. Declarative process modelling

Declaratief procesmodelleringsparadigma

- Het declaratieve modelleerparadigma voor bedrijfsprocessen richt zich op het vastleggen en definiëren van regelgevende of interne richtlijnen in beperkingen, bijv. volgorde-relaties tussen de verschillende activiteiten.
- Met een minimale specificatie van de relevante business concerns wordt enige vrijheid gelaten voor de exacte volgorde van de activiteiten.

Fig. 8. Mandatory constraints restrict the set of possible behaviors while optional constraints further guide the user. Since traditional approaches explicitly specify the possible behaviors, declarative languages tend to be more flexible [13].



Dynamische, doelgerichte uitvoering

- In tegenstelling tot procedurele procesmodellering, omvat declaratieve procesmodellering niet het vooraf berekenen van taakbesturingsstromen, informatiestromen en werkallocatieschema's.
- Tijdens de uitvoering van een declaratief procesmodel wordt een geschikt uitvoeringsscenario geconstrueerd (door een menselijke of machinale coördinator) dat de bedrijfsdoelstellingen van het procesmodel realiseert. Dit laatste wordt doelgerichte uitvoering (planning) genoemd.

Bias bij aannames

- Omdat procedurele procesmodellen het resultaat zijn van een impliciete voorcalculatie van taakafhankelijkheden, is het niet algemeen gegarandeerd dat procedurele procesmodellen niet een aantal extra aannames bevatten die het onderliggende bedrijfsproces te veel specificeren.
- Procedurele procesmodellen zijn vaak overgespecificeerd.

Overspecificatie

- In een declaratief model specificeren we de beperkingen:
 - o Bijv. 3 activiteiten A, B, C moeten worden uitgevoerd, maar B moet vóór C komen.
- In een procedureel model overspecificeren wij vaak door een (of enkele) bepaalde volgorde(s) te kiezen:
 - o Bijv
 - ABC
 - BCA
 - o Maar hoe zit het met:
 - BAC

Declaratieve procesmodellering talen

- Een gemeenschappelijk idee van declaratieve bedrijfsprocesmodellering is dat een proces wordt gezien als een traject in een toestandsruimte en dat declaratieve beperkingen worden gebruikt om de geldige bewegingen in die toestandsruimte te definiëren.
- De verschillen tussen declaratieve procestalen zijn deels te wijten aan verschillende percepties van toestandsruimte, overgangstypen en overgangsrestricties.
- Belangrijkste talen
 - o CMMN
 - o DCR
 - o Declare

3. Declare

Declare

- Body of process model constraints
- Komt voort uit modelcontrole, waarbij Lineaire Temporele Logica (LTL) wordt gebruikt om te controleren op eigenschappen (gebeurt A altijd vóór B?).
- Gaat verder dan het gebruik van afzonderlijke constraints en kan worden gebruikt in een model
- Modelleren: met symbolische bogen, bijv. Response(A,B): wanneer ik a heb moet b volgen



- Bevat het begrip regels, waaraan op verschillende manieren kan worden voldaan

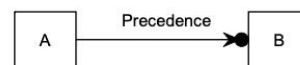
- Voldaan: de beperking is vervuld
- Tijdelijk geschonden: aan de beperking is niet voldaan, maar kan in de toekomst alsnog worden voldaan
- Permanent geschonden: de beperking is geschonden en kan niet meer worden vervuld.

Declaratieve procesmodellering met behulp van beperkingen (in Declare)

- Bijvoorbeeld een Precedence constraint:
 - Activiteit B moet, als ze voorkomt, voorafgegaan worden door (minstens één) activiteit A.
- Andere
 - Coëxistentiebeperkingen
- Geordende relaties
 - Voorrang, antwoord, opeenvolging
 - Afwisselend geordende relaties (Alternate Precedence, response, succession) -> elk op zijn beurt
 - Chain ordered relations (Chain Precedence, response, succession) -> onmiddellijk geketend

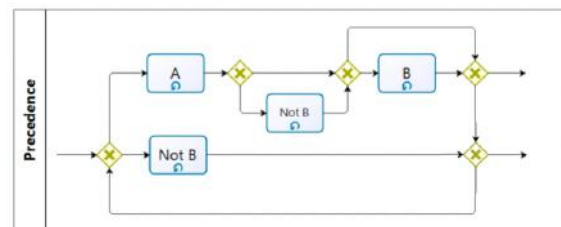


Voorrang/ precedence



- Activiteit B moet, indien zij plaatsvindt, worden voorafgegaan door (ten minste één) activiteit A.
 - Voorbeelden:

- ✓AAAB
- ✓ABBBBAA
- ✓ABAAABBAB
- ✓ABCAAB
- ✓CCCA
- ✓AAA
- ×BAA
- ×BAB



These constraints are hard to model in BPMN

- "geen B vóór een A, of nooit B". Merk op dat dit een zeer ontspannen voorrangrelatie is, omdat B niet onmiddellijk na A hoeft te worden uitgevoerd, en een andere A tussen de eerste A en B kan worden uitgevoerd.
- Bijv. voordat u een cursus kunt volgen (B), moet u bij ons langskomen (A).

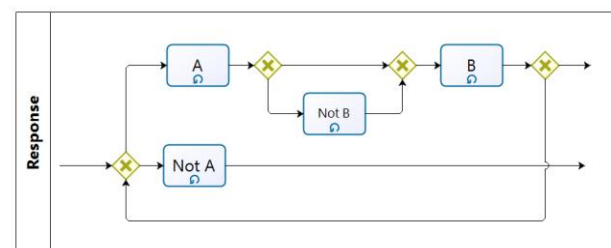
Antwoord / response

- Wanneer activiteit A wordt uitgevoerd, moet daarna activiteit B worden uitgevoerd uiteindelijk.

- Voorbeelden:

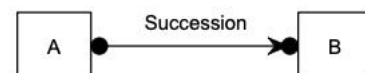
- ✓BABBB
- ✓ABAB
- ✓ABAAB
- ✓CCAAB
- ×BAABA

- Na de maaltijd (A), moet je de tafel schoonmaken (B).

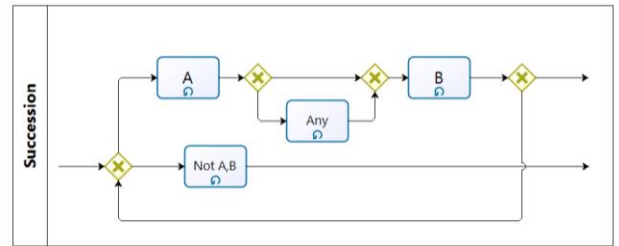


Successie / succession

- Antwoord(A,B) & voorrang(A,B)
 - Wanneer activiteit A wordt uitgevoerd, moet activiteit B uiteindelijk daarna worden uitgevoerd. Activiteit B moet, als zij plaatsvindt, worden voorafgegaan door (ten minste één) activiteit A.

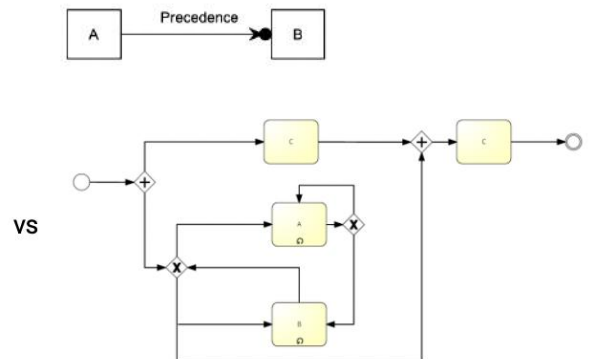
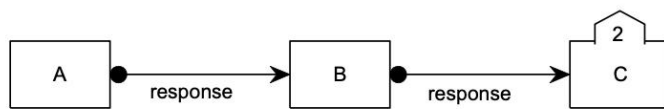


- Als u factureert (B), moet u iets geleverd hebben (A), u kunt niet leveren zonder te factureren.



Declaratieve modellen

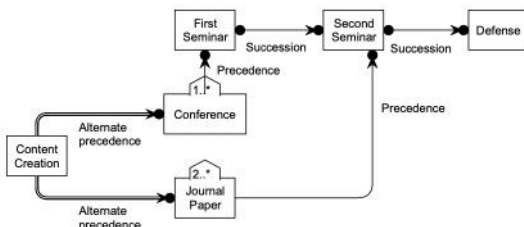
- Al deze soorten gedrag zijn moeilijk te specificeren in bijvoorbeeld BPMN-modellen of Petri-netten.
 - o Zelfs voor één beperking.
 - o Maar zeker voor een gecombineerde reeks beperkingen:



Voorbeeld: Promotietraject

- Het creëren van inhoud kan op elk moment gebeuren.
- Conferentie en paper kunnen alleen plaatsvinden na een nieuwe creatie van inhoud als ze nog niet zijn ingeschakeld.
- Eerste seminar kan alleen plaatsvinden na een conferentie, tweede seminar na een paper.
- Eerste en tweede seminar moeten uiteindelijk gevolgd worden door een verdediging.

Declaratief procesmodel – Doctoraat

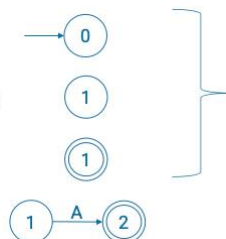


4. Declare constraints and execution

Verklaren - Uitvoering

- Voor de uitvoering worden reguliere expressies gebruikt (die kunnen worden voorgesteld als eindige toestandsmachines)
- Eindige toestandsmachines

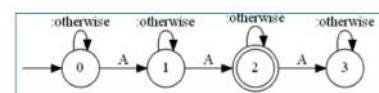
- Initial state
- Non-accepting state
- Accepting state
- Transition



Deze soorten staten komen overeen met

- Tevreden
- Tijdelijk en permanent geschonden

- ➔ Geeft een natuurlijke manier om zowel een proces en de notie van regelverificatie te behouden

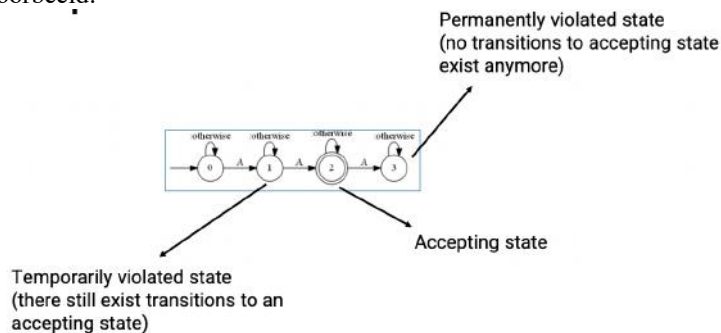


Voorbeeld:

- Precies(A,n=2): A moet precies 2 keer voorkomen.

- Begintoestand:
 - A is nog niet voorgekomen, $\#A < 2 \rightarrow$ niet-aanvaardbaar, tijdelijk geschonden (aanvaardende toestand nog steeds bereikbaar)
- A vuurt:
 - A heeft 1 keer gevuurd, $\#A < 2 \rightarrow$ niet-aanvaardbaar, tijdelijk geschonden
- A vuurt:
 - A heeft 2 keer gevuurd, $\#A = 2 \rightarrow$ accepteren
- A vuurt:
 - A heeft 3 keer gevuurd, $\#A > 2 \rightarrow$ niet aanvaarden, permanent geschonden (geen weg terug)
- Geen andere activiteit beïnvloedt de uitkomst van de beperking

Voorbeeld:

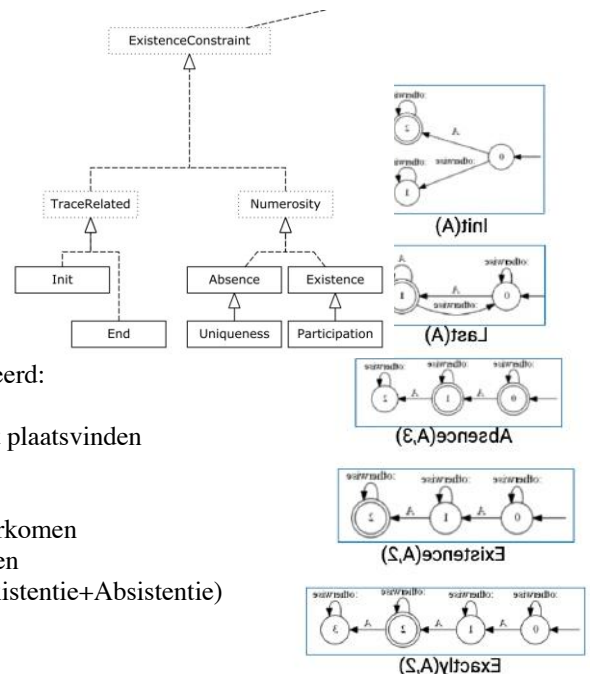


Beperkingen aangeven

- Body of constraints: gestructureerd en hiërarchisch
- Unaire en binaire constraints
- Unaire beperkingen:

Name	Graphical	Meaning
absence(n+1, a)		Activity a can be executed at most n times
existence(n, a)		Activity a must be executed at least n times
exactly(n, a)		Activity a must be executed exactly n times
init(a)		Activity a must be the first executed activity
last(a)		Activity a must be the last executed activity

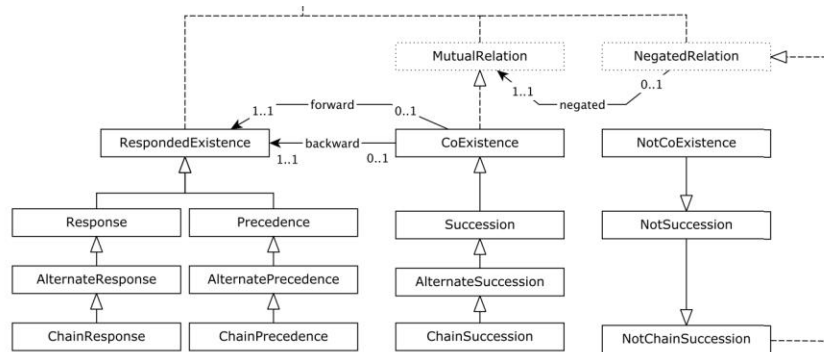
Unary Declare beperkingen



- Positiegerelateerd:

- Init(A): activiteit A moet als eerste plaatsvinden
- End/Last(A): activiteit A moet als laatste activiteit plaatsvinden
- Op numerositeit gebaseerd:
 - Afwezigheid(A,n+1): A kan maximaal n keer voorkomen
 - Existence(A,n): A moet minstens n keer voorkomen
 - Exact(A,n): A moet precies n keer voorkomen (Existentie+Absistentie)

Binaire Declaratiebeperkingen

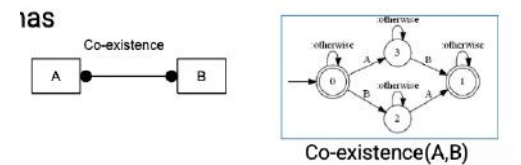
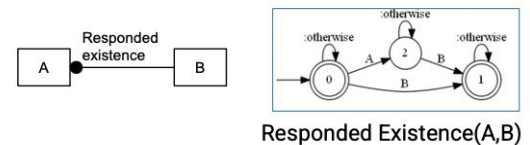


(Maar 11 daarvan kennen ! negatieve constraints moet je niet kennen)

Beperkingen aangeven – Binair

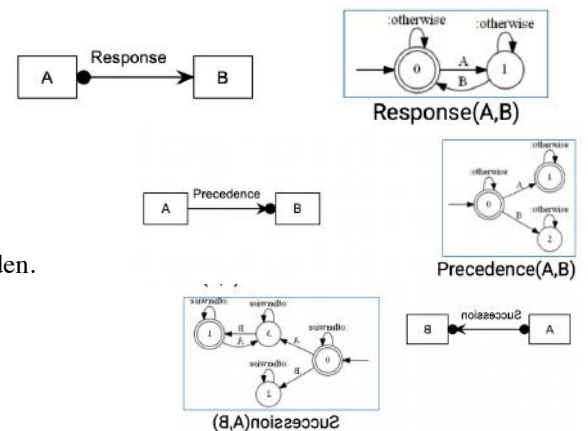
- Positie-agnostisch:
 - o Responded existence(A,B): als A voorkomt, moet B ook voorkomen
 - Bijv: ACAABA, BBA, CCCB, BB
 - Niet: ACC, AA, CAA
 - o Co-existentie(A,B): wanneer A voorkomt, moet B voorkomen en omgekeerd
 - Bijv: CCC, AABCBBA, BA, AB
 - Niet: ACAAAC, CBCC

Het aantal A's en B's is niet belangrijk.



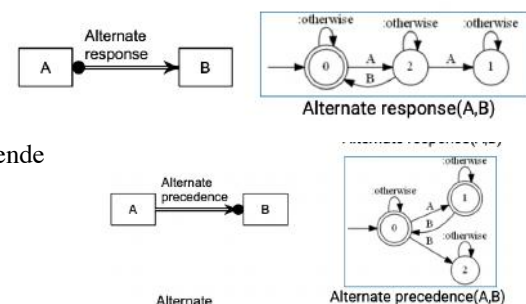
Beperkingen aangeven - Eenvoudig geordend

- Respons(A,B): wanneer A optreedt, moet B uiteindelijk ook optreden.
 - o Bijv: ACAAB, ABB, AAAB, ACCBAB, BB
 - o Niet: AABBB, A
- Voorrang(A,B): A moet gebeuren voordat B kan gebeuren
 - o Bijv: ACAABA, A, CCC, AA
 - o Niet: BACC, BB, CB, CBA
- Successie(A,B): zowel Precedentie(A,B) als Respons(A,B) gelden.
 - o Bijv: CCC, ABBB, AB, AAB, ABB, CACCAB
 - o Niet: ACC, AABBA, ACBCA, BBA, A,

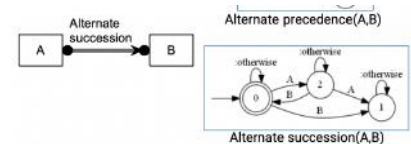


Beperkingen aangeven - Afwisselend geordend

- Alternatief antwoord(A,B): elk voorkomen van A moet worden gevolgd door een nieuw voorkomen van B.
 - o Bijv: ABAB, BBABBAB, ACCCBAB, CCB
 - o Niet: AAB, CA, A, ACAAB
- Alternatieve voorrang(A,B): B kan alleen voorkomen na het eerstvolgende voorkomen van A
 - o Bijv: AAA, CCC, AC, ABAA, ABACCBA, C
 - o Niet: ABB, CB, ABBA

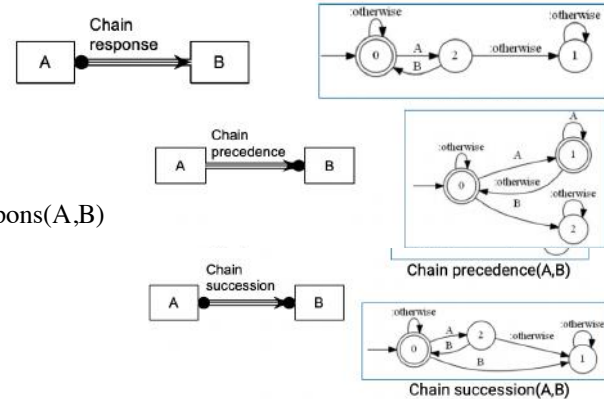


- Alternatieve opeenvolging(A,B): zowel Alternatieve voorrang(A,B) als Alternatieve reactie(A,B) gelden.
 - o Bijv: CCC, AB, ACCB, ACCBCCACCB
 - o Niet: ACC, AABBA, ACBCA, BBA, A, B



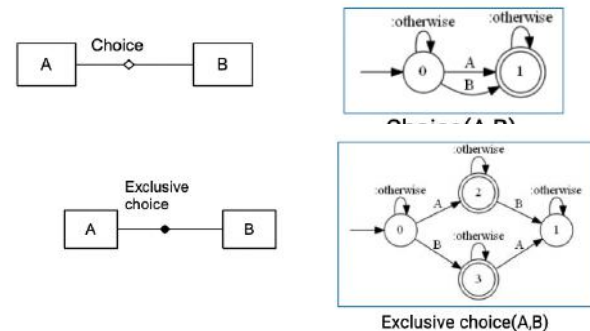
Beperkingen aangeven - Ketten geordend

- Kettingreactie(A,B): Direct na een optreden van A, kan alleen B optreden.
 - o Bijv: ABABBB, BB, CC, ABCBCCABB
 - o Niet: ACB, CA, CCAC
- Chain precedence(A,B): B kan alleen vlak na A voorkomen
 - o Bijv: AAA, CCC, AC, ABAA, C
 - o Niet: AAB, CB, B, ACAAB, ABAAACCBA
- Ketenvolging(A,B): zowel Ketenvoorrang(A,B) als Ketensuccessie(A,B) gelden
 - o Bijv: CCC, AB, CCABC
 - o Niet: ACC, AABBA, ACBCA, BA, A, B



Beperkingen aangeven – Keuze

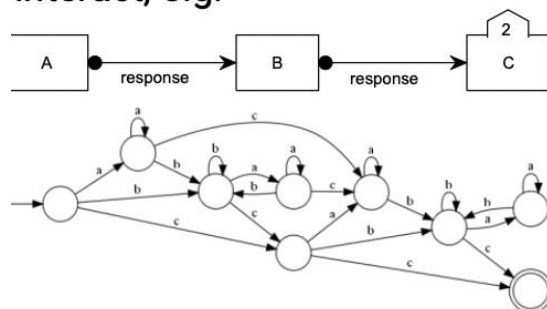
- Keuze(A,B): Of A of B moet voorkomen, of beide
 - o Bijv: ABAA, ABAAB, CCA, B, A, BC
 - o Niet: CCC
- Exclusieve keuze(A,B): A en B kunnen niet samen voorkomen, maar minstens één moet voorkomen
 - o Bijv: AAA, AC, CBBB, B A
 - o Niet: AAB, BA, ACAB, CC



Modellen declareren

- Vermenigvuldiging van alle beperkingen om een model te verkrijgen (u hoeft niet te weten hoe dit precies in zijn werk gaat)
- Beperkingen werken op elkaar in, bijv.

interact, e.g.



- ✓ CABBBAAABC
- ✓ ABCC
- ✓ CC
- ✓ ACBC
- × ABBC
- × ACC
- × BCC
- × CBCB
- × AABCBAAC

bcc wel juist

- C moet precies twee keer voorkomen - als A brandt, moet B uiteindelijk daarna voorkomen - als B brandt, moet C uiteindelijk daarna voorkomen
- Declare constraints zijn zeer geschikt voor:
 - o Lusgedrag:
 - Alternatieven hebben verschillende manieren om een lus te definiëren
 - o Globaal gedrag/afhankelijkheden op lange afstand
 - Chain constraints: beïnvloeden alle activiteiten omdat ze het hele model kunnen stoppen om één activiteit te laten plaatsvinden.
 - BPMN zou een |A|-1 boog nodig hebben om dit te modelleren
 - Iets n keer uitvoeren (tellen):

- Keuze en andere beperkingen
- BPMN zou speciale gateways vereisen, multiple instance activities (alleen sequentieel), duplicate activiteiten, ...
 - Negatieve beperkingen (extra, hier buiten beschouwing gelaten)
 - Expliciet specificeren wat niet is toegestaan

5. Conclusion

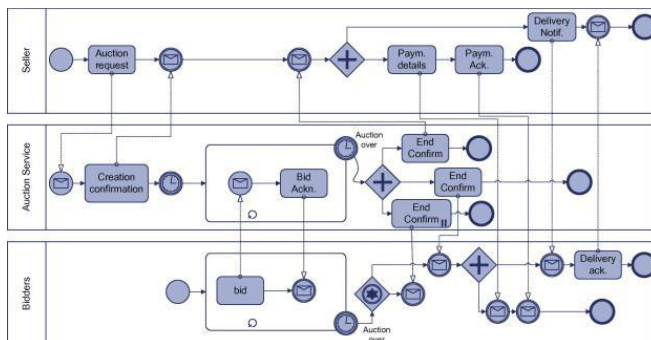
Belangrijkste conclusies

- Twee verschillende paragrafen voor het modelleren van bedrijfsprocessen
 - Procedureel vs. Declaratief
- Wanneer flexibiliteit vereist is, kan de declaratieve aanpak de voorkeur genieten
 - Case management systemen
 - Kennisintensieve processen
- Er bestaan verschillende declaratieve modelleringstalen
- Adoptie in de praktijk is traag

BPM6. Decision Modelling with DMN

1. Process modelling and decisions

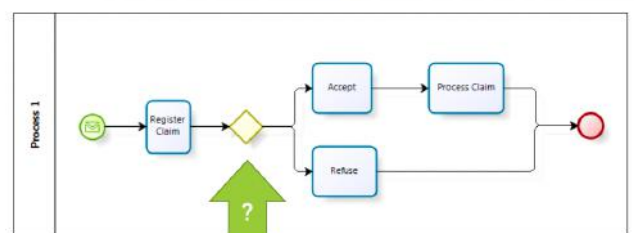
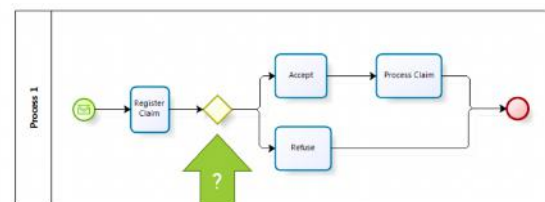
Wat moet er in een procesmodel staan?



Uitzonderingen?
 Timers?
 Gelukkig pad?
 Beslissingen? Beslissingslogica? Bedrijfsregels?
 Rollen?
 Berichten?
 Meldingen?
 Triggers? Voorwaarden?

Wat te modelleren?

- Beslissingen zijn belangrijk voor bedrijven, niet alleen processen
- Waarom zouden we alleen de processen modelleren?
 - Waar zit de beslissing?
 - Hoe wordt de beslissingslogica gemodelleerd?
 - Modelleer de beslissingsactiviteit: **Besluit acceptatie**
- Processen bevatten beslissingen
- Wanneer een claim accepteren/afwijzen? Afhankelijk van:
 - Klantgeschiedenis
 - Formaat eisen
 - Tijdsbeperkingen
 - Soort contract
 - ...



Beslissingen zijn er in vele vormen

- Strategische en tactische beslissingen
 - o Vaak op hoog niveau, lange termijn, grote impact, onregelmatig, ongestructureerd
- Operationele beslissingen
 - o Dagelijks, grote volumes, vaak gestructureerd, standaard procedures
 - Wiskundige (en analytische) modellen, of
 - Standaard operationele procedures
 - Gebaseerd op bedrijfsregels, beleid en voorschriften
 - Aanwezig in operationele bedrijfsprocessen
 - Bijv.: geschiktheid, prijs, verzekering, korting, diefstalwaardering, klantenaanbod, retentie, leveranciersselectie, huur, krediet, ...



Operationele beslissingen

- Vaak gemaakt
- Niet-triviaal
- Snel genomen
- Consequent uitgevoerd
- Hoog volume
- Meetbare bedrijfsgevolgen
- Deterministisch
- Frequente verandering (maar conform)
- Begrijpelijkheid
- Geautomatiseerd of handmatig

Beslissingen beheren

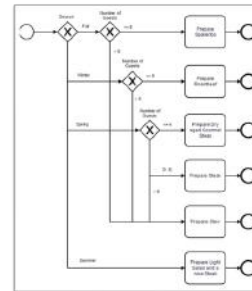
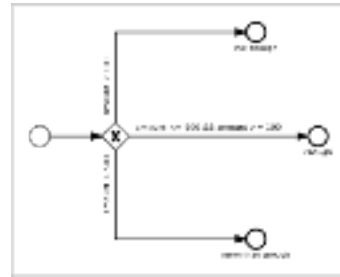
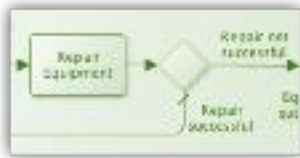
- Wat is de beslissing?
 - o geschiktheid, prijs, verzekering, diefstalwaardering, klantenaanbod, retentie, leveranciersselectie, huur, krediet, ...
- Wie, wat, hoe?
- Wie is de eigenaar van de beslissing?
- Wie neemt de beslissing elke dag?
- Wie wordt beïnvloed?
- Wat triggert de beslissing?
- Wanneer nemen we deze beslissing?
- Wat is er nodig om deze beslissing te nemen?
 - o Vereiste informatie
 - o Kennisbronnen (regelgeving, analyses, expertise)
 - o Andere beslissingen
 - o Besluitlogica

Beslissingen in processen

1. Binnen een kennisintensieve activiteit
 - Handmatig of geautomatiseerd
2. Of zoals blijkt uit een eenvoudige gateway
 - Gebaseerd op een enkele vraag (criterium) met twee of meer uitkomsten



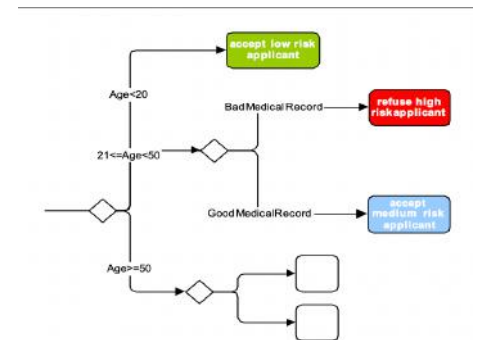
- Opmerking: de gateway zelf is niet de beslissing



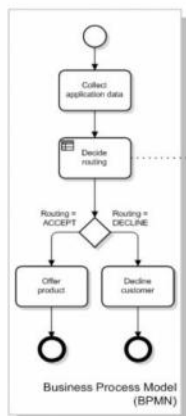
3. Of zoals blijkt uit een cascade van poorten
- Op basis van meerdere vragen (criteria).

De verkeerde manier

- Beslisbomen moeten geen procespaden zijn
 - o Codeer beslisregels niet hard in het procesmodel
 - o Het scheiden van (beslissings)regels van het proces vereenvoudigt het proces
 - o Vereenvoudig geneste beslissingspaden
 - Bepaal type aanvrager
 - Type aanvrager is afhankelijk van leeftijd (en in sommige gevallen ook van medisch dossier)



Scheid de bedrijfsregel van het proces

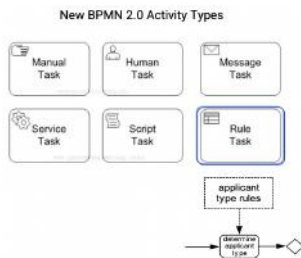


Rules to decide the routing

→ Beslissingstaak : Speciale symbool die taak aangeeft
 => beslissingstabelsymbool: taak die beslissingslogica bevat.

→ Staat vooraf een XOR-gate: die enkel de uitkomst van de beslissing gaat nemen om de controlflow impact te gaan modelleren.
 → Je kan de regels aanpassen zonder dat het een impact heeft op jouw procesmodel.

Erkend in BPMN 2.0



Voor de rule task moet je het symbool zetten !
(Voor andere tasks hoeft het niet)

Besluiten moeten worden gemodelleerd

- Beslissingsmodellen zijn geen details op lager niveau van één proces
- Beslissingsmodellen kunnen meerdere activiteiten en zelfs meerdere processen omvatten
- Scheiding van belangen

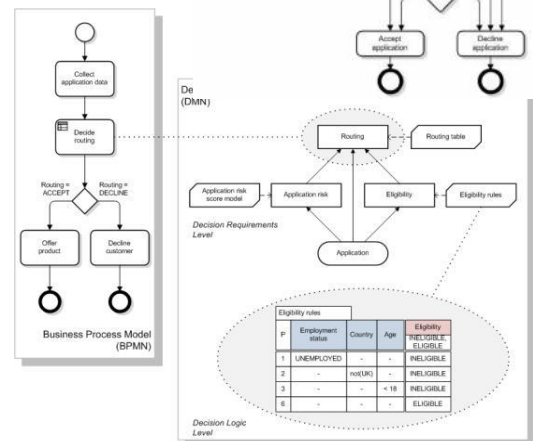
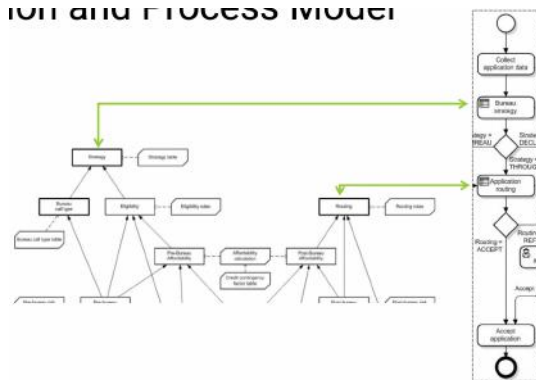
Beslissings- en procesmodel:

Decision requirement diagram : alles wat je nodig hebt om een beslissing te nemen.

→ Op de hoogste niveau de uiteindelijke beslissingen die een impact gaan hebben op controlflow.

⇒ Decision and Process Model

Decision and Process Model



De noodzaak van DMN

- Beslissing(en) (regels) moeten worden gemodelleerd
 - o Zeer complete regels modelleren bv. Fraudedetectie bij banken is op basis van 1000den regels
- DMN beslissingsmodellen zijn bedoeld voor de business, niet voor IT
- Een standaard voor processen (BPMN) is niet voldoende

→ Decision Modeling & Notation standaard (DMN)

2. Decision Model and Notation (DMN)

Beslissingsmodel en Notatie

- De DMN (Decision Modeling & Notation) standaard werd geproduceerd door OMG (versie 1.0 in 2014, versie 1.3 in 2021): <https://www.omg.org/dmn/>
- DMN is een modelleertaal en (uitvoerbare) notatie voor de precieze specificatie van zakelijke beslissingen en bedrijfsregels
- DMN is gemakkelijk leesbaar voor de verschillende soorten mensen die betrokken zijn bij het beheer van beslissingen
 - o bedrijfsanalisten die de initiële beslissingseisen en gedetailleerde beslissingsmodellen opstellen
 - o bedrijfsmensen die de regels specificeren en de toepassing ervan beheren en controleren
 - o technische ontwikkelaars die verantwoordelijk zijn voor het automatiseren van de beslissingen in processen
- Ook mensen zonder/ beperkte IT-kennis
- Uitwisselbaar tussen tools/organisaties via een XML-weergave

Beslissingen als zakelijk belang

- Steeds meer bedrijfsactiviteiten worden geautomatiseerd
- Bedrijven moeten onmiddellijk reageren, bv. bij online toepassingen
- Een mens is vaak niet meer betrokken op het moment van de transactie

→ Het is belangrijk dat systemen, processen en beslissingen correct zijn ontworpen en correct blijven na updates.

Besluitautomatisering

- DMN verenigt modellering en uitvoering
- DMN-modellen kunnen (direct) worden geautomatiseerd met behulp van een Business Rules Management System (BRMS)
 - o Beslisservice
 - Inkapseling van de beslissingslogica
 - Terbeschikkingstelling van een interface
 - Wanneer de beslisdienst wordt aangeroepen met een reeks invoergegevens, evalueert hij de gespecificeerde beslissingen en geeft hij de resultaten ervan terug

Consistente en verklaarbare beslissingen

- Geautomatiseerde systemen en processen bevatten de logica van beslissingen
- Bij geautomatiseerde beslissingen moet de beslissingslogica
 - o goed ontworpen
 - o correct
 - o consistent
 - o verklaarbaar
 - o begrijpelijk voor de business
 - o gemakkelijk te veranderen
 - o onderhouden door het bedrijf

DMN: Twee modelleringsniveaus

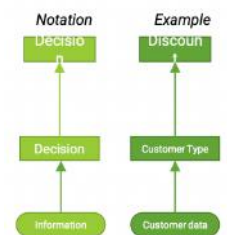
- Het niveau van de beslissingsvereisten
 - o Wat is er nodig om deze beslissing te nemen?
 - o Informatie
 - o Het resultaat van andere beslissingen
 - o Kennisbronnen (regelgeving, analyses, expertise)
- Het beslissingslogica-niveau
 - o Hoe wordt de uitkomst van de beslissing bepaald?

→ Wat is nodig om een beslissing te nemen?

- Informatie
- De uitkomst van andere beslissingen (is ook informatie)
- Kennisbronnen (regelgeving, analytics, expertise)
- Kennis (de beslissingslogica)
- Dit is een doelgerichte beschrijving van de beslissing
 - De topbeslissing is het doel

Notatie van besluitvormingsvereisten

- Een beslissing hangt af van criteria, invoergegevens, beslisregels, enz.
- Het decision requirements diagram (DRD) modelleert deze afhankelijkheden met behulp van slechts enkele symbolen.
 - Een kader vertegenwoordigt een beslissing
 - Een afgeronde rechthoek stelt de vereiste informatie voor om de beslissing te maken
 - Een volle pijl staat voor een informatiebehoefte
- Een beslissing kan ook afhangen van de uitkomst van een andere beslissing
 - Dat is ook een vereiste informatie
 - Bijv. In een regel kan staan dat als het klanttype A is, de korting 0 is, waarbij klanttype ook een beslissing is met eigen gegevens, logica, enz.



Wanneer hangt een beslissing af van een andere beslissing?

- In een autoverhuurbedrijf wordt een huur aangeboden of geweigerd afhankelijk van de leeftijdsgerechtigdheid, de geldigheidsduur van het afzetten en de beschikbaarheid van het gevraagde voertuig.
- Een voertuig is beschikbaar als de gevraagde autogroep beschikbaar is (AsRequested) of als een update groep beschikbaar is (Upgrade).
- VehicleAvailability toont de afhankelijkheid
 - Het is een vereiste in de eerste zin (indien beschikbaar)
 - En een conclusie in de tweede zin (beschikbaar indien)
- Als een informatie-item op de ene plaats een conclusie is, en op een andere plaats een voorwaarde, dan is het de bron van een afhankelijkheid.



DMN-basiscomponenten

- Een beslissing is het bepalen van een outputwaarde (de gekozen optie) uit een aantal inputwaarden, met behulp van logica die bepaalt hoe de output uit de inputs wordt bepaald.
- Deze beslissingslogica kan een of meer bedrijfskennismodellen omvatten die de bedrijfskennis inkapselen in de vorm van bedrijfsregels, analysemodellen of andere formalismen. Deze kunnen in verschillende modellen herbruikt worden
 - Data / beslissing → volle pijl
 - Knowledge → stippelijntje

Kennisbronnen

- Autoriteiten kunnen worden gedefinieerd voor beslissingen of bedrijfskennismodellen.
- Dit kunnen (bijvoorbeeld) domeinsdeskundigen zijn die verantwoordelijk zijn voor het definiëren of onderhouden van het kennismodel, of brondocumenten waaruit bedrijfskennismodellen zijn afgeleid, of sets van testgevallen waarmee de beslissingen consistent moeten zijn.
- Dit worden kennisbronnen genoemd

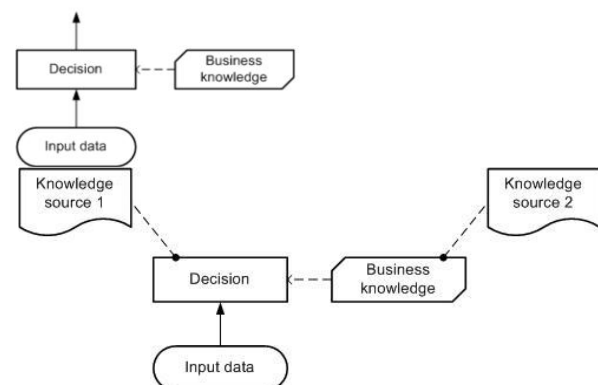
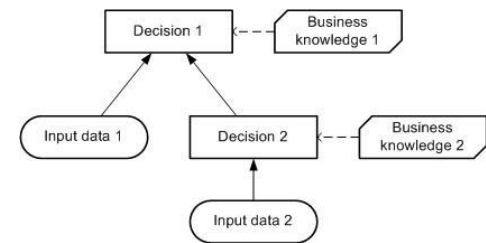


Diagram voor besluitvormingsvereisten

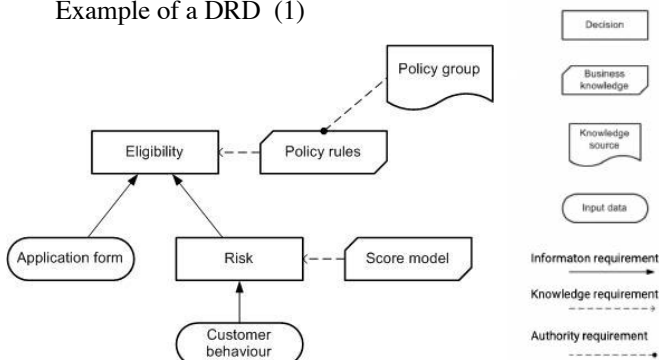
- Van een besluit wordt gezegd dat het zijn inputs "nodig" heeft om zijn output te bepalen.
- De inputs kunnen inputgegevens zijn, of de outputs van andere beslissingen
- Als de input van een beslissing beslissing1 de output van een andere beslissing beslissing2 omvat, "vereist" beslissing1 beslissing2.
- Beslissingen kunnen daarom worden verbonden in een netwerk dat een Decision Requirements Graph (DRG) wordt genoemd en dat kan worden getekend als een Decision Requirements Diagram (DRD).



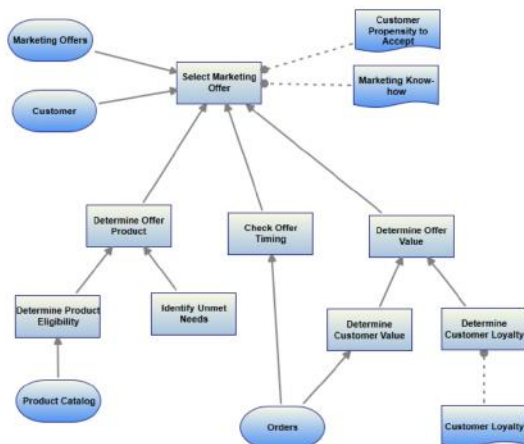
Samenvatting

Component		Description	Notation
Elements	Decision	A decision denotes the act of determining an output from a number of inputs, using decision logic which may reference one or more business knowledge models.	Decision
	Business Knowledge Model	A business knowledge model denotes a function encapsulating business knowledge, e.g., as business rules, a decision table, or an analytic model.	Business knowledge
	Input Data	An input data element denotes information used as an input by one or more decisions. When enclosed within a knowledge model, it denotes the parameters to the knowledge model.	Input data
	Knowledge Source	A knowledge source denotes an authority for a business knowledge model or decision.	Knowledge source
Requirements	Information Requirement	An information requirement denotes input data or a decision output being used as one of the inputs of a decision.	Information requirement
	Knowledge Requirement	A knowledge requirement denotes the invocation of a business knowledge model.	Knowledge requirement
	Authority Requirement	An authority requirement denotes the dependence of a DRD element on another DRD element that acts as a source of guidance or knowledge.	Authority requirement

Example of a DRD (1)

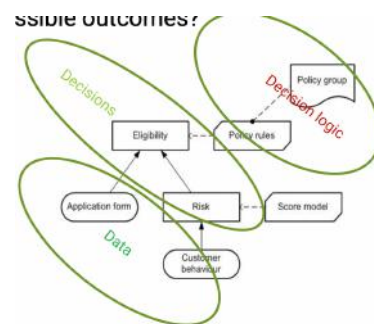


Example of DRD (2)



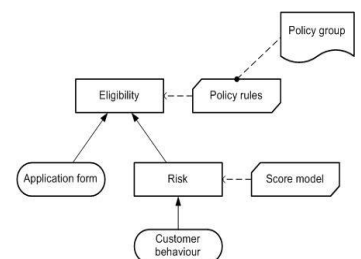
Wat te doen, of hoe te doen? - Wat moet er besloten worden?

- Mogelijke resultaten?
- Beslissingen vereisen:
 - o Invoergegevens
 - Transacties
 - Master data
 - Externe gegevens
- Besluitlogica
 - o Regels, kennis
 - o Beleid
 - o Analytics
- Resultaat van andere beslissingen
 - o Herbruikbaarheid



Voordelen van besluitvormingseisen modellering

- Waarom is het zo belangrijk om de beslissingsstructuur te scheiden van de beslissingslogica?
- Het Decision Requirements Diagram (DRD) richt zich op het grote geheel, niet op de details van de beslisregels.
 - o Het maakt het mogelijk de beslissing op te splitsen in subbeslissingen
 - o Het vermijdt de "grote emmer van regels"
 - o Het maakt het mogelijk de regels bij te werken zonder de structuur te veranderen
 - o Het maakt hergebruik van subbeslissingen mogelijk
 - o Het is declaratief van aard
 - Als een beslissing bijvoorbeeld twee subbeslissingen vereist, legt de DRD geen uitvoeringsvolgorde op.

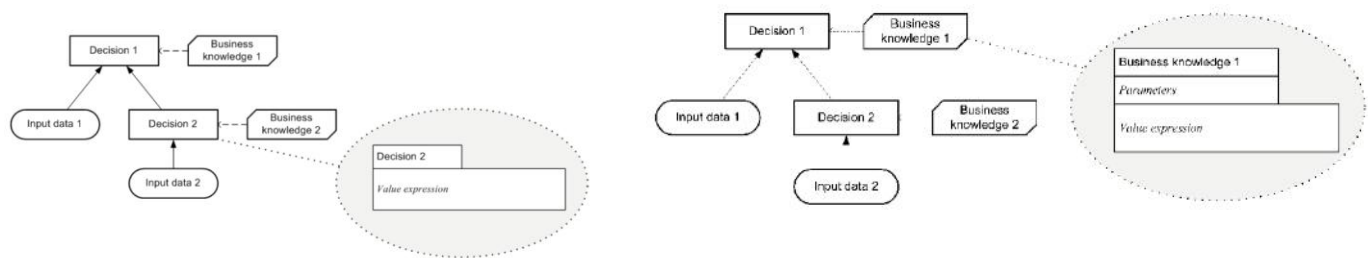


4. DMN decision logic

Besluitlogica

- Met beslissingslogica kunnen de beslissingscomponenten van een DRG gedetailleerder worden gespecificeerd, zodat een volledige reeks bedrijfsregels en berekeningen wordt vastgelegd en (indien gewenst) de besluitvorming volledig kan worden geautomatiseerd.
- Op het niveau van de beslissingslogica wordt elke beslissing in een DRG gedefinieerd met behulp van een **waarde-expressie/ value expression** die aangeeft hoe de output van de beslissing wordt bepaald aan de hand van de inputs.
- Op dat niveau wordt de beslissing beschouwd als de evaluatie van de uitdrukking. De waarde-uitdrukking kan worden genoteerd met behulp van een boxed expression

Beslissingslogica als uitdrukkingen in een kader



Drie basistypen uitdrukkingen/ types van expression

- Beslissingstabellen: logica gebaseerd op regels
- Letterlijke uitdrukkingen: formules zoals rekenkunde
- Invocaties: gebruik van beslissingslogica van elders (bv. BKM)

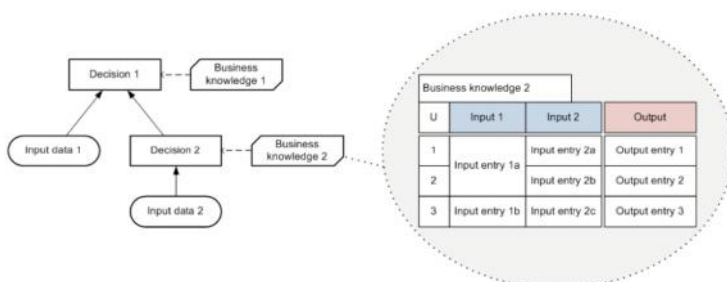
Hoe beslislogica schrijven?

- Natuurlijke taal
 - o Onduidelijk, dubbelzinnig
- Logica
 - o Krachtig, ondubbelzinnig, maar niet voor zakenmensen
- Gestructureerde Engelse regels
 - o Subset van natuurlijk Engels
 - o Afweging tussen gebruiksgemak en kracht
- Beslisbomen, tabellen, grafieken, diagrammen
 - o Verschillende weergaven voor verschillende doeleinden: acquisitie, V&V, besluitvorming, afhankelijkheden => standaardmanier !
- Object Constraint Language
 - o Deel van UML
 - o Nuttig voor pre- en postcondities impactanalyse

Hoe beslissingslogica schrijven?

- Een beslissings- of bedrijfskennismodel kan elke beslissingslogica bevatten die kan worden weergegeven als een functie.
- Hierdoor kunnen veel bestaande normen voor het modelleren van beslissingslogica (bv. voor bedrijfsregels en analytische modellen) in DMN worden geïmporteerd.
- Een belangrijk formaat van beslissingen/zakelijke kennismodellen, dat specifiek wordt ondersteund in DMN, is de Beslissingstabel

Besluitenlogica als een beslissingstabel

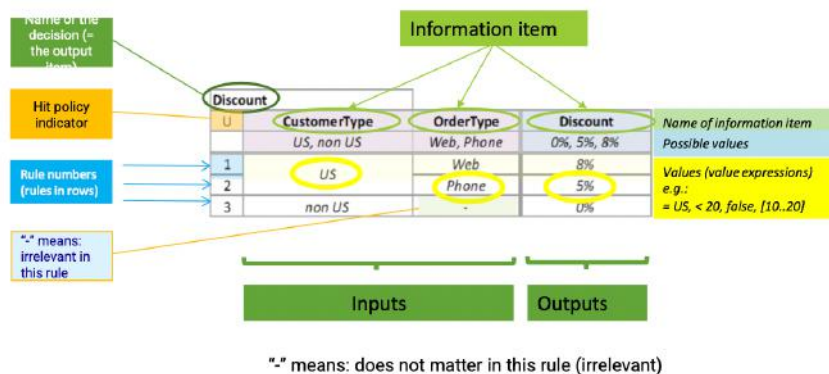


Achter twee mogelijke elementen kan je een beslissingstabel zetten in een businessproces : achter de beslissing zelf of achter een business knowledge element.

Vb. decision 2 is een evocation.

5. Decision tables

Elementen van een beslissingstabel in DMN



- Naam van de beslissing
- Rijen: elke rij is een individuele beslissing
- Inputs en outputs: inputs aan de linker kan van de lijn – outputs aan de rechter kan van de lijn
- Information item: verwacht je als input te hebben in jouw DRD voor jouw beslissing
- Naam van de information item
- Value expression (in dit geval gebruik ik strings (=tekstvelden DUS geen numerieke variabele) als input en outputs)
- Hit policy indicator: op welke manier kan je tot een uitkomst/beslissing komen in de beslissingstabel?
 - o In dit geval unique hit decision table (= elke combinatie van inputs tot 1 unieke uitkomst zal komen, bij elke inputs wordt er 1 regel geactiveerd)

De beslissingstabel is niet nieuw => heel oud voorbeeld

To Quit or not to Quit

If I don't get a raise of at least 10 percent, I will find a job somewhere else. But if I get promoted, then I will expect my own office or I'll quit, unless the work is going to be more interesting; in which case, I'll stay with just a 10 percent raise.

Quit or not quit -QUI-

Promotion ?	Y	N	
Own office ?	Y	N	-
Raise of ≥ 10 % ?	-	Y	N Y N
More interesting work ?	-	Y	N - -
Quit	-	-	x x - x
Stay	x	x	- - x -

Waarom beslissingstabellen?

- Problemen met lijsten van regels:
 - o De volgorde van informatie-items is niet in alle regels hetzelfde
 - o De naam van informatie-items wordt in veel regels herhaald
 - o Verbindingen kunnen zijn en, of, met haakjes, enz.
 - o Regels zijn complex, moeilijk te begrijpen en moeilijk te valideren.
- Beslissingstabellen lossen deze problemen op
 - o De volgorde van informatie-items zal in alle regels hetzelfde zijn
 - o De naam van informatie-items wordt slechts eenmaal geschreven
 - o Verbindingen zijn alleen AND
 - o De verzameling regels zal gemakkelijk te valideren zijn

Hit policies

- Single hit (1 regel met resultaat(en))
 - o Standaard:
 - Als regels elkaar niet overlappen: unieke treffer
 - o Herken anderen:

- Als regels elkaar overlappen, dan moet 1 regel worden geselecteerd: any hit, first hit, priority hit
- Multiple hit (geeft een lijst van regels terug)

Hit policies:

- Als een regel nodig is:
 - Uniek (als er maar één is voor elke set ingangen) [standaard]
 - Of selecteer de gewenste: Any, First, Priority
- Als alle toepasselijke regels vereist zijn:
 - Resultaten in een uitvoerlijst (bewerkingen kunnen worden toegepast, bijv. som(partialscores))
 - Lijstvolgorde is Geen volgorde (willekeurig), Regelvolgorde, Uitvoervolgorde

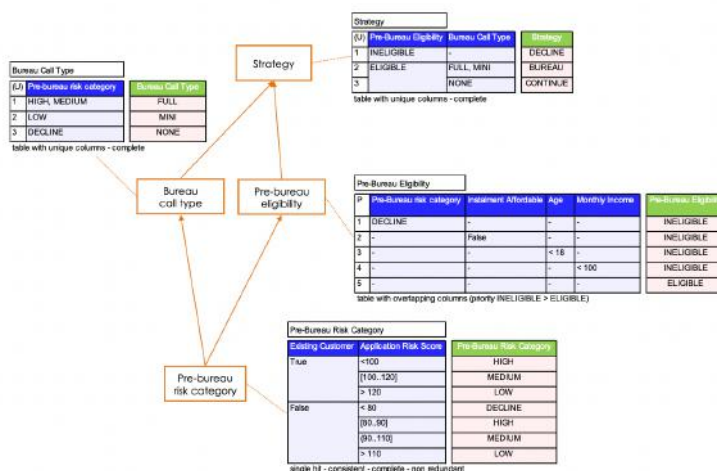
Applicant Risk Rating			
U	Applicant Age	Medical History	Applicant Risk Rating
1		good	Medium
2	> 60	bad	High
3	[25..60]	-	Medium
4		good	Low
5	< 25	bad	Medium
Inputs			Outputs

DMN Tafelsoorten (Single hit)

- DMN onderscheidt verschillende soorten tabellen, aangegeven door de eerste letter:
 - **Unieke hit**-tabellen: elk invoergeval is opgenomen in slechts één regel. Er is geen overlapping tussen regels. Goed
 - **Any hit** tables: elk invoergeval mag in meer dan één regel voorkomen, maar de uitkomsten zijn gelijk. Regels mogen elkaar overlappen. Lelijk
 - **Priority hit** tables: meerdere regels kunnen overeenkomen, met verschillende uitkomstwaarden. Dit beleid geeft de matchende regel met de hoogste prioriteit voor de uitkomstwaarde (bv. hoogste korting). De prioriteitsvolgorde wordt bepaald door de volgorde van de vermelde outputwaarden in de kop van de outputkolom.
 - Eerste hit tabellen / **First hit** tables : meerdere (overlappende) regels kunnen overeenkomen, met verschillende uitkomst waarden. De eerste treffer per regelvolgorde wordt geretourneerd (en de evaluatie kan stoppen). Dit is een gebruik, omdat het inconsistenties oplost door de eerste treffer te forceren. Het is belangrijk dit type tabel te onderscheiden van andere, omdat de betekenis Bad afhangt van de volgorde van de regels. Vanwege deze volgorde is de tabel moeilijk handmatig te valideren en moet daarom voorzichtig worden gebruikt. SLECHT

→ In de meeste gevallen kiezen voor eerst een unieke hit en als dat niet mogelijk is een priority hit!

Voorbeeld van beslissingslogica



Unieke hitbeslissingstabellen

Applicant Risk Rating			
U	Applicant Age	Medical History	Applicant Risk Rating
1	> 60	good	Medium
2		bad	High
3	[25..60]	-	Medium
4		good	Low
5	< 25	bad	Medium

Applicant Risk Rating					
Applicant Age	< 25		[25..60]	> 60	
Medical History	good	bad	-	good	bad
Applicant Risk Rating	Low	Medium	Medium	Medium	High
U	1	2	3	4	5

Applicant Risk Rating		Medical History	
		good	bad
Applicant Age	< 25	Low	Medium
	[25..60]	Medium	Medium
	> 60	Medium	High

Applicant Risk Rating					
Applicant Age	< 25		[25..60]	> 60	
Medical History	good	bad	-	good	bad
Low	X	-	-	-	-
Medium		X	X	X	
High					X
U	1	2	3	4	5

Waarom verschillende soorten tabellen?

- Twee uitgangspunten
 - o Hoe eenvoudiger hoe beter, en minder regels is eenvoudiger
 - o De regels moeten de onderliggende logica zo begrijpelijk mogelijk maken mogelijk
- Vergelijking
 - o U-tabellen vereisen vaak meer regels, maar zijn minder foutgevoelig
 - o A-tabellen kunnen de onderliggende logica (relatie invoer tot uitvoer) gemakkelijk te begrijpen
 - o P-tabellen kunnen handig zijn als de input van de beslissing veel opgesomde waarden heeft, maar ze vertroebelen soms de logica.

Validatie van beslissingstabellen

- Wat maakt een goede beslissingstabel?
 - o Volledigheid
 - Er mag geen combinatie van toegestane invoerwaarden zijn die niet overeenkomt met ten minste één regel.
 - o Consistentie
 - In een U-tabel mogen regels elkaar niet overlappen.
 - In een A-tabel moeten overlappende regels dezelfde outputwaarde hebben.
 - In een P-tabel mag geen enkele regel worden "gemaskeerd" (verschijnen onder een regel die altijd raak zou zijn)
 - o Vermijd subsumptie
 - Beslissingstabellen mogen geen twee of meer regels bevatten die gecombineerd kunnen worden
 - Beslissingstabel samentrekking toepassen
- Moet je zelf allemaal kunnen nakijken

6. Expression language for decision logic (FEEL)

Uitdrukken van de beslissingslogica

- FEEL ("Friendly Enough Expression Language") implementeert de vereiste mechanismen
 - o Ingebouwde types, functies en operatoren
 - o Elke beslissing in een DRD kan worden gedefinieerd met behulp van een formele uitdrukking die specificeert hoe de output van de beslissing wordt bepaald aan de hand van de input.
 - o Volledige beslissingsmodellen kunnen worden gedefinieerd
 - o Formele expressies kunnen ook worden ingekapseld als functies.
 - o Ondersteunt abstractie, samenstelling en schaalbaarheid.
- S-FEEL ("Simple FEEL") is een basissubset van FEEL, ontworpen om aan de essentiële eisen van op beslissingstabellen gebaseerde DMN-modellen te voldoen.
- DMN kan worden uitgebreid met andere expressie- en typedefinitietalen

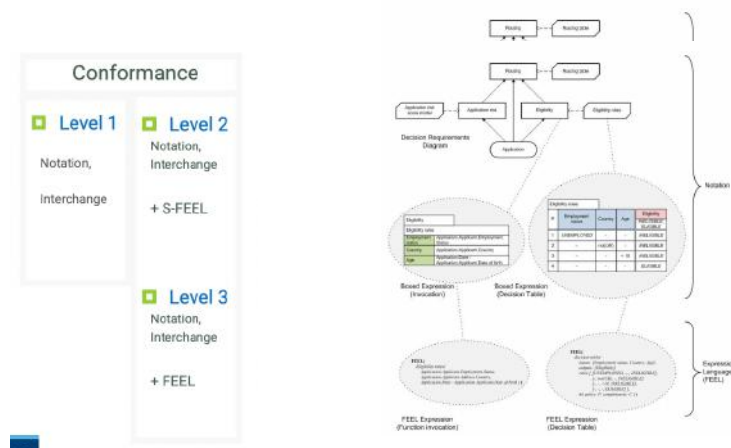
FEEL

- Vergelijkbaar met functies in Excel
 - o Gewoon een uitdrukkingstaal die waarden berekent
- Bruikbaar voor zakenmensen

→ Om DMN beslissingsmodellen uitvoerbaar te maken, is FEEL een belangrijke vereiste.

- o DMN-modellen zijn niet slechts zakelijke eisen die aan programmeurs worden doorgegeven.
- o U kunt het model direct uitvoeren in een modelleertool
- o U kunt het implementeren in een cloudservice

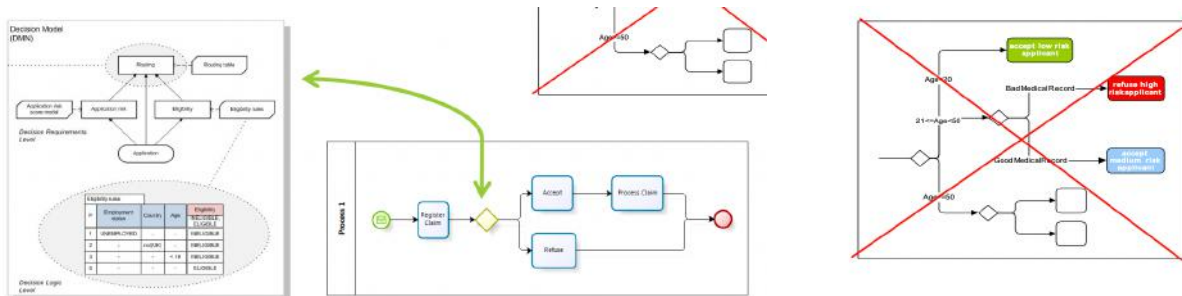
Conformiteit



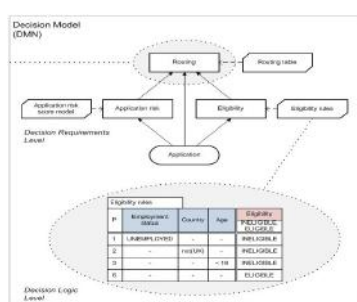
7. Conclusion

Problemen die DMN oplost

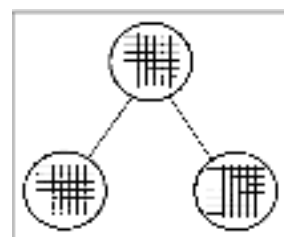
- Scheiden van beslissingen en processen
 - o Gebruik van een standaard modelleernotatie



- Scheiding van beslissingsstructuur en beslissingslogica
 - o Maakt het mogelijk beslisrelaties te modelleren, zelfs als niet alle logica in tabellen wordt uitgedrukt



- Methodologie voor besluitvormingsmodellering
 - o Behoud de inzichten uit het verleden en voorkom verwarring
 - o Goede beslissingstabellen modellen zijn een beproefde techniek voor het weergeven van beslissingsregels
 - o Consistentie, volledigheid en correctheid door ontwerp
- Typen beslissingstabellen
 - o Herkennen, en ondubbelzinnig uitwisselen
 - o DMN staat meerdere tabeltypes toe (goed voor uitwisseling)
 - Unieke hit, Elke hit (zoals in sommige methodologieën), Eerste hit (zoals in sommige tools), Meervoudige hit, ...
 - Goede beslissingstabelmodellen zijn veel strikter
 - Beslissingstabelmethodologie is een methodologie, DMN niet
- DMN bevat niets nieuws over beslissingstabellen, maar DMN maakt het mogelijk andere beslissingstabelformaten te standaardiseren en te herkennen en vermijdt dubbelzinnigheid over het concept.



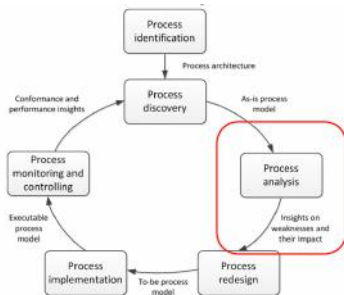
Toepassingsgebieden & hulpmiddelen

- Beslissingen, beslissingen, beslissingen
 - o Beslissingen in procesmodellering
 - o Procesvariabiliteit
- "Eenvoudige" operationele beslissingen
 - o Toelaatbaarheid, krediet, churn, sociale zekerheid, financiën, verzekeringen, gezondheidszorg, veiligheid, definities, ...
 - o Regelgeving, wetgeving
- Opsommen en evalueren van alternatieven, beperkingen
 - o Combinatorische optimalisaties
 - o Planning en toewijzing van middelen

BPM7. Qualitative Process Analysis

1. Introduction

Business Process Analysis



=> hoe zit het met de gezondheid van mijn proces ?

Technieken voor procesanalyse

- Kwalitatieve analyse (subjectief)
 - o Analyse van toegevoegde waarde en afval
 - o Issuedocumentatie
 - Issueregister
 - Analyse van de Onderliggende Oorzaak
 - Paretoanalyse - PICK-diagrammen
- Kwantitatieve analyse (cijfermatig)
 - o Kwantitatieve stroomanalyse
 - o Wachtrijtheorie
 - o Proces simulatie

2. Value-added analysis & waste analysis

Analyse van de toegevoegde waarde

- Decorticeer het proces in stappen
 - o Stappen die aan een taak voorafgaan
 - o De taak zelf, eventueel onderverdeeld in kleinere stappen
 - o Stappen die worden uitgevoerd na een taak, ter voorbereiding op de volgende taak
- Classificeer elke stap in:
 - o Toegevoegde waarde (VA/ Value adding)
 - o Bedrijfswaarde toevoegend (BVA/ Business value adding)
 - o Niet-waarde toevoegend (NVA/ non value adding)

Activiteiten die waarde toevoegen

- Produceren waarde of voldoening voor de klant
- Criteria
 - o Is de klant bereid voor deze stap te betalen?
 - o Is de klant het ermee eens dat deze stap noodzakelijk is om zijn doelen te bereiken?
 - o Als de stap wordt verwijderd, zou de klant dan vinden dat het eindproduct of de dienst minder waardevol is?
- Voorbeelden
 - o Order-to-cash proces: Bevestig leveringsdatum, Lever producten
 - o Toelating tot de universiteit: Aanvraag beoordelen, Toelatingsresultaat meedelen

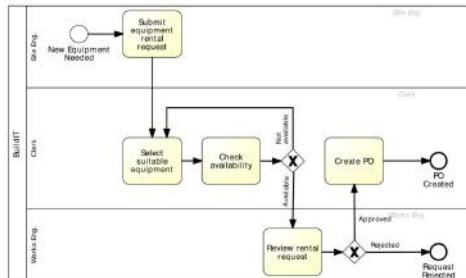
Activiteiten die waarde toevoegen aan het bedrijf

- Noodzakelijk of nuttig voor de bedrijfsvoering
- Criteria
 - o Is deze stap nodig om inkomsten te verzamelen, het bedrijf te verbeteren of te laten groeien?
 - o Zou het bedrijf er op lange termijn (mogelijk) onder lijden als deze stap werd verwijderd?
 - o Vermindert deze stap het risico van bedrijfsverliezen?
 - o Is deze stap vereist om te voldoen aan wettelijke vereisten?
- Voorbeeld
 - o Order-to-cash proces: Inkooporder controleren, kredietwaardigheid van klant controleren, factuur uitreiken, betaling innen, feedback van klant verzamelen
 - o Toelatingsproces universiteit: Volledigheid van aanvraag controleren, Geldigheid van diploma's controleren, Geldigheid van resultaten taaltest controleren

Activiteiten die geen waarde toevoegen

- Al het andere dan VA en BVA. Activiteiten waarvoor de klant niet bereid is te betalen.
- Omvat
 - o Overdrachten, contextwisselingen
 - o Wachttijden, vertragingen
 - o Herwerken of verhelpen van defecten
- Voorbeelden
 - o Order-to-cash proces: PO doorsturen naar magazijn, bevestiging opnieuw versturen, afgekeurde producten ontvangen
 - o Toelatingsproces universiteit: Aanvragen doorsturen naar commissie, Toelatingsresultaten van commissie ontvangen

Voorbeeld (verhuur van apparatuur, voorbeeld 1.1 Procure-to-pay bij BuildIT)



Vervolg Voorbeeld - Verhuur van apparatuur

Step	Performer	Classification
Fill request	Site engineer	VA
Send request to clerk	Site engineer	NVA
Open and read request	Clerk	NVA
Select suitable equipment	Clerk	VA
Check equipment availability	Clerk	VA
Record recommended equipment & supplier	Clerk	BVA
Forward request to works engineer	Clerk	NVA
Open and examine request	Works engineer	BVA
Communicate issues	Works engineer	BVA
Forward request back to clerk	Works engineer	NVA
Produce PO	Clerk	BVA
Send PO to supplier	Clerk	VA

Afval elimineren (Muda)

- “Het enige wat we doen is kijken naar de tijdslijn, vanaf het moment dat de klant ons een order geeft tot het moment dat we het geld ophalen.
 - En we verkorten de tijdslijn door de niet waarde toevoegende verspillingen te verminderen.”
Taiichi Ohno, Toyota
- ⇒ Identificeren van afval/waste !

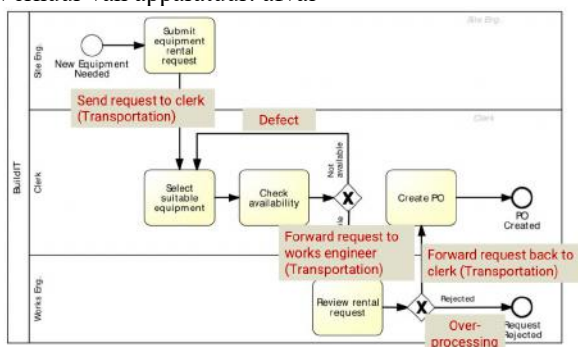
7+1 Afvalbronnen

1. Onnodig vervoer (verzenden, ontvangen)
2. Beweging (drop-off, pick-up, ga naar)
⇒ Move
3. Inventarisatie (groot werk in uitvoering)
4. Wachten (wachtijd tussen taken)
⇒ Hold
5. Over-Processing (uitvoeren wat nog niet nodig is of misschien niet nodig is)
6. Over-Productie (onnodige zaken)
7. Defecten (herwerken van defecten)
⇒ Over do
8. Onderbenutting van middelen (verspilling van intellect)

D.O.W.N.T.I.M.E => verschillende soorten waste (niet kennen)



Verhuur van apparatuur: afval



3. Issue documentation

⇒ High level problemen identificeren

1) Issueregister

- Doel: het categoriseren van geïdentificeerde problemen als onderdeel van de as-is procesmodellering
- Gewoonlijk een tabel met de volgende kolommen (eventueel andere):
 - o Issuenummer
 - o Naam
 - o beschrijving/uitleg
 - o impact: kwalitatief vs. Kwantitatief
 - o mogelijke oplossing

⇒ Schatting

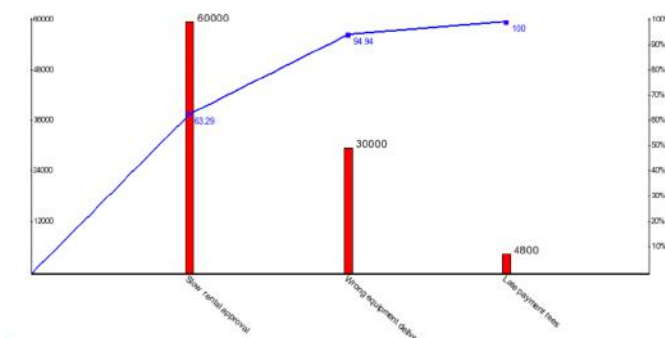
Issue register (Equipment rental example)

Name	Explanation	Assumptions	Qualitative Impact	Quantitative Impact
Equipment kept longer than needed	Site engineers keep the equipment longer than needed by means of deadline extensions	BuildIT rents 3000 pieces of equipment p.a. In 10% of cases, site engineers keep the equipment two days longer than needed. On average, rented equipment costs 100 per day		$0.1 \times 3000 \times 2 \times 100 = 60,000$ p.a.
Rejected equipment	Site engineers reject delivered equipment due to non-conformance to their specifications	BuildIT rents 3000 pieces of equipment p.a. Each time an equipment is rejected due to an internal mistake, BuildIT is billed the cost of one day of rental, that is 100. 5% of them are rejected due to an internal mistake	Disruption to schedules. Employee stress and frustration	$3000 \times 0.05 \times 100 = 15,000$ p.a.
Late payment fees	BuildIT pays late payment fees because invoices are not paid by the due date	BuildIT rents 3000 pieces of equipment p.a. Each equipment is rented on average for 4 days at a rate of 100 per day. Each rental leads to one invoice. About 10% of invoices are paid late. Penalty for late payment is 2%.		$0.1 \times 3000 \times 4 \times 100 \times 0.02 = 2400$ p.a.

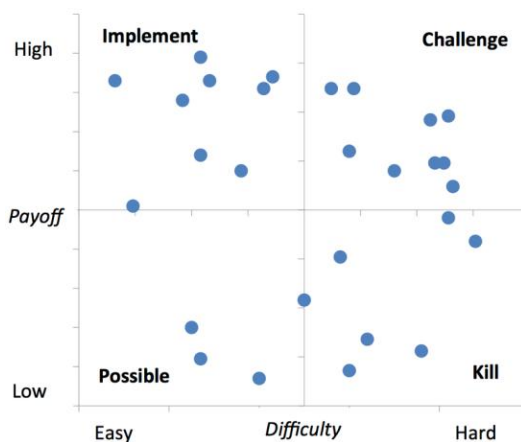
2) Paretodigram

- Nuttig om een verzameling problemen of factoren achter een probleem te prioriteren
- Staafdiagram waarbij de hoogte van de staaf de impact van elk vraagstuk aangeeft
- Staven gesorteerd op impact
- Bovenliggende curve waarbij de y-as de cumulatieve procentuele impact weergeeft

Voorbeeld: excessive rental expenses

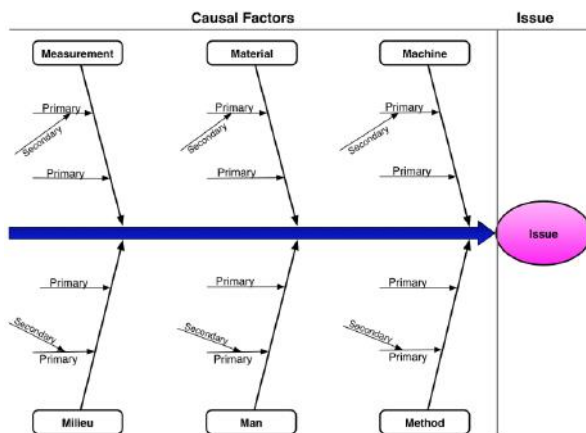


3) PICK Chart – Two-Dimensional Prioritization



4. Root cause analysis

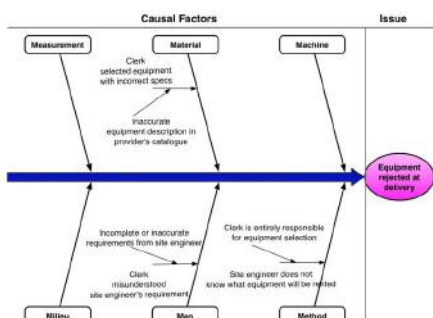
Cause-effect (fishbone) diagrams



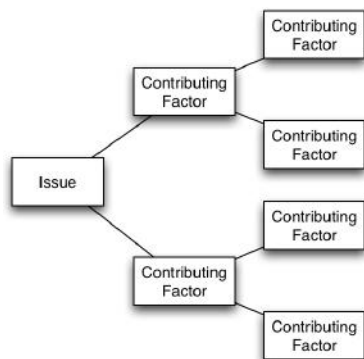
Categorieën van oorzaken: Zes Ms (uitleg hoeft je niet van buiten kennen!)

- Machine: factoren die voortvloeien uit de gebruikte technologie
 - o Gebrek aan geschikte functionaliteit in de ondersteunende softwaretoepassingen
 - o Slecht ontwerp van de gebruikersinterface (UI)
 - o Gebrek aan integratie tussen systemen
- Methode: factoren die voortvloeien uit de manier waarop het proces is ontworpen, begrepen of uitgevoerd
 - o Onduidelijke toewijzing van verantwoordelijkheden
 - o Onduidelijke instructies
 - o Onvoldoende opleiding
 - o Gebrek aan tijdige communicatie
- Materiaal: factoren die voortvloeien uit inputmaterialen of -gegevens
 - o Ontbrekende, onjuiste of verouderde gegevens
- Man: factoren die voortkomen uit verkeerde beoordelingen of onjuiste uitvoering van stappen die te wijten zijn aan:
 - o Gebrek aan opleiding en duidelijke instructies
 - o Gebrek aan motivatie
 - o Te hoge eisen aan procesmedewerkers
- Meten: factoren die voortkomen uit het vertrouwen op:
 - o Onnauwkeurige schattingen
 - o Misrekeningen
- Milieu: factoren buiten de scope van het proces
 - o Vertragingen veroorzaakt door niet reagerende externe actoren
 - o Plotselinge toename van de werklust door bijzondere omstandigheden

Oorzaak-gevolg diagram (afgekeurde apparatuur)



Why-why diagram



Waarom-waarom diagram (verhuur van materieel)

- Werf ingenieurs houden materieel langer aan, waarom?
- ... ○ De ingenieur vreest dat het materiaal later niet beschikbaar zal zijn, waarom?
- tijd tussen aanvraag en levering te lang, waarom?
- te veel tijd besteed aan het vinden van geschikt materieel en het goedkeuren van de aanvraag, waarom?
- ... ○ tijd besteed aan het contacteren van mogelijk meerdere leveranciers na elkaar, waarom?

⇒ 5 why's diep

5. Conclusion

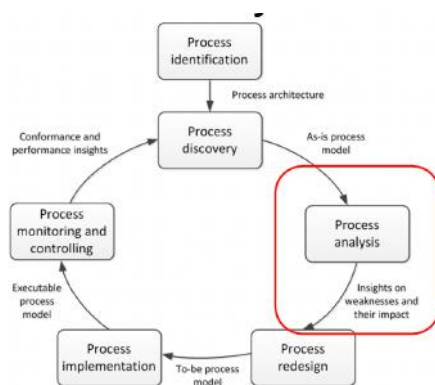
Kwalitatieve procesanalyse omvat:

- Segregeer stappen die waarde toevoegen, die zakelijke waarde toevoegen en stappen die geen waarde toevoegen (analyse van de toegevoegde waarde)
- Identificeren van verspilling (verspillingsanalyse)
- Problemen verzamelen en systematisch organiseren, hun impact beoordelen (probleemregister, Pareto grafiek, PICK grafiek)
- Analyseer hoofdoorzaken van issues (visgraatdiagram, waarom-waarom diagram)

BPM8. Quantitative Process Analysis

1. Introduction

Business Process Analysis



Technieken voor procesanalyse

- Kwalitatieve analyse
 - Analyse van de toegevoegde waarde en afvalanalyse
 - Issuedocumentatie
 - Issueregister
 - Oorzaakanalyse
 - Paretoanalyse - PICK-diagrammen
- **Kwantitatieve analyse** (Cijfermatig analyse)

- Kwantitatieve stroomanalyse
- Wachtrijtheorie
- Proces simulatie

Procesprestatiemetingen

- Kosten:
 - Kosten per uitvoering
 - Gebruik van hulpbronnen
 - Afval
- Tijd
 - Cycle tijd
 - Wachtijd / tijd besteed aan taken die geen waarde toevoegen
- Kwaliteit
 - Wachtijd/tijd besteed aan taken die geen waarde toevoegen
 - Gemiste belofte

2. Flow analysis

1) Analyse van de cyclustijd/ cycle time

- Cyclustijd: Verschil tussen de begin- en eindtijd van een opdracht.
- Cyclustijdanalyse: de taak van het berekenen van de gemiddelde cyclustijd voor een volledig proces of procesfragment
 - Veronderstelt dat de gemiddelde activiteitstijden voor alle betrokken activiteiten beschikbaar zijn (activiteitstijd = wachttijd + verwerkingstijd)

→ In het eenvoudigste geval bestaat een proces uit een opeenvolging van activiteiten op een sequentieel pad

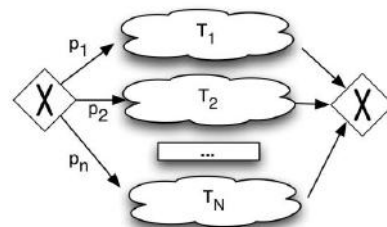
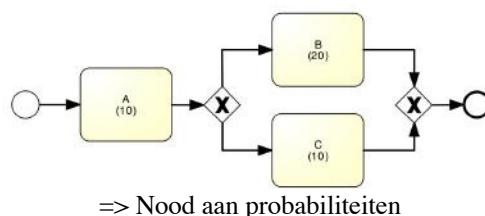
- De gemiddelde cyclustijd is de som van de gemiddelde activiteitstijden

... maar in het algemeen moeten we rekening kunnen houden met

- Alternatieve paden (XOR-splitsingen)
- Parallelle paden (AND-splitsingen)
- Herwerken (cycli)

Alternatieve paden

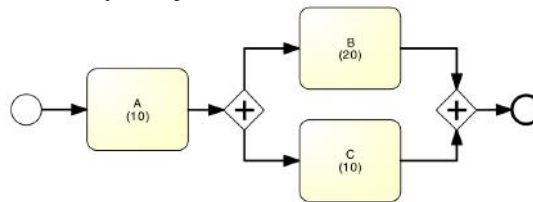
Voorbeeld - Wat is de gemiddelde cyclustijd?



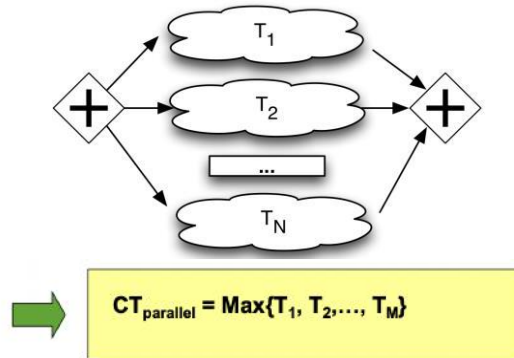
$$CT = p_1 T_1 + p_2 T_2 + \dots + p_n T_n = \sum_{i=1}^n p_i T_i$$

Parallele paden

Voorbeeld - Wat is de gemiddelde cyclustijd?

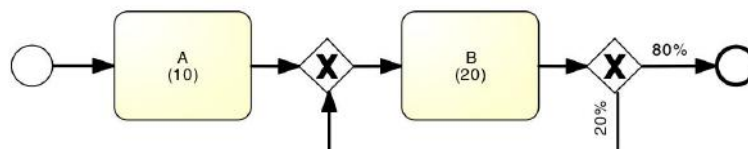


Indien twee activiteiten met betrekking tot dezelfde taak parallel worden uitgevoerd, is de bijdrage aan de cyclustijd voor de taak het maximum van de twee activiteitentijden.

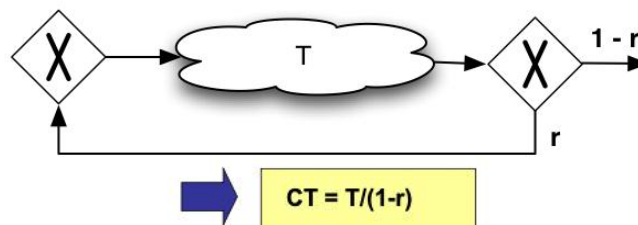


Herwerken - Voorbeeld

Wat is de gemiddelde cyclustijd?

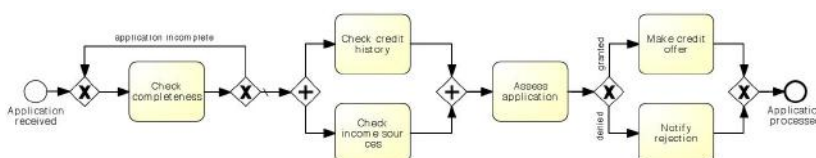


Veel processen omvatten controle- of inspectiepunten waar het werk, als het niet aan bepaalde normen voldoet, wordt teruggestuurd voor herbewerking.



Oefening

Bereken de cycle time



Activity	Cycle time
Check completeness	2 hours
Check credit history	2 hours
Check income sources	5 hours
Assess application	7 hours
Make credit offer	3 hours
Notify rejection	2 hours

Wat is de gemiddelde cyclustijd van dit proces?

- A. 16,2 uur
- B. 16,5 uur
- C. 16,8 uur
- D. 17,1 uur**
- E. 17,3 uur

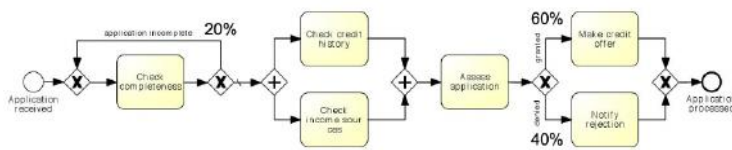
2) Efficiëntie van de cyclustijd

Gemeten als het percentage van de totale cyclustijd dat aan waardetoevoegende activiteiten wordt besteed.

$$\text{Cycle Time Efficiency} = \frac{\text{Theoretical Cycle Time}}{\text{CT}}$$

- CT = cyclustijd zoals hierboven gedefinieerd
- Theoretische cyclustijd (TCT) is de cyclustijd als we alleen waardetoevoegende activiteiten tellen en wachttijd of overdrachtstijd uitsluiten.
 - o Alleen verwerkingstijden tellen

Voorbeeld: bereken het efficiëntie tijd



Activity	Cycle time	Processing time
Check completeness	2 hours	20 mins
Check credit history	2 hours	10 mins
Check income sources	5 hours	30 mins
Assess application	7 hours	50 mins
Make credit offer	3 hours	15 mins
Notify rejection	2 hours	5 mins

Wat is de efficiëntie van de cyclustijd? (theoretical cycle time = 116 min)

- A. 11,3%
- B. 21,3%
- C. 31,3%
- D. 41,3%

3) Werk in uitvoering/ Work in Progres: Wet van Little / Little's Law

$$\text{WIP} = \lambda \cdot \text{CT}$$

- WIP = (gemiddeld) onderhanden werk
 - o Aantal lopende zaken (gestart maar nog niet afgerond)
 - o Bijv. # actieve en niet ingevulde orders in een order-to-cash proces
- = aankomstnelheid (aantal nieuwe zaken per tijdseenheid)
- CT = cyclustijd

⇒ Heel vaak een nuttigere KPI dan een gewone tijdsduur

Oefening

Een fastfoodrestaurant ontvangt gemiddeld 1200 klanten per dag (tussen 10:00 en 22:00 uur). Tijdens de piekuren (12:00-15:00 en 18:00-21:00) ontvangt het restaurant in totaal ongeveer 900 klanten en zijn er op een bepaald moment (gemiddeld) 90 klanten in het restaurant te vinden. Buiten de piekuren ontvangt het restaurant in totaal 300 klanten en zijn er op een bepaald tijdstip (gemiddeld) 30 klanten in het restaurant aanwezig.

- Wat is de gemiddelde tijd die een klant in het restaurant doorbrengt tijdens de piekuren?
- Wat is de gemiddelde tijd die een klant in het restaurant doorbrengt buiten de piekuren?
- Het restaurant wil een marketingcampagne lanceren om meer klanten aan te trekken. De capaciteit van het restaurant is echter beperkt en wordt tijdens piekuren te vol. Wat kan het restaurant doen om dit probleem aan te pakken zonder te investeren in uitbreiding van het gebouw?

Wat is de gemiddelde tijd die een klant in het restaurant doorbrengt tijdens piekuren/niet-piekuren?

- A. 28 minuten
- B. 32 minuten
- C. 36 minuten**
- D. 40 minuten
- E. 44 minuten

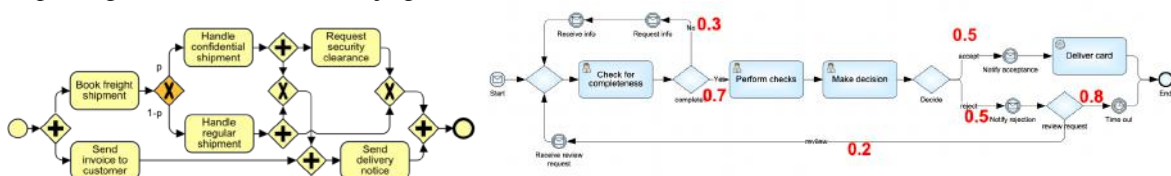
4) Stromingsanalyse: Andere invalshoeken en beperkingen

De voorgaande techniek voor cyclustijdanalyse is slechts één voorbeeld van wat met stromingsanalysetechnieken kan worden gedaan

Andere toepassingen:

- Berekenen van kosten-per-proces-instantie
 - o And wordt dan + ipv van max
- berekenen van foutenpercentages op procesniveau
- schatten van capaciteitsbehoeften

Beperking 1: Niet alle modellen zijn gestructureerd



⇒ Mathematische uitwerking onmogelijk

Beperking 2: Vaste belasting + vaste middelen capaciteit

- Cyclustijdanalyse houdt geen rekening met wachttijden als gevolg van conflicterende middelen
- Analyse en simulatie van wachtrijen pakken deze beperkingen aan en zijn breder toepasbaar.

3. Queuing analysis

Waarom is stroomanalyse niet voldoende?

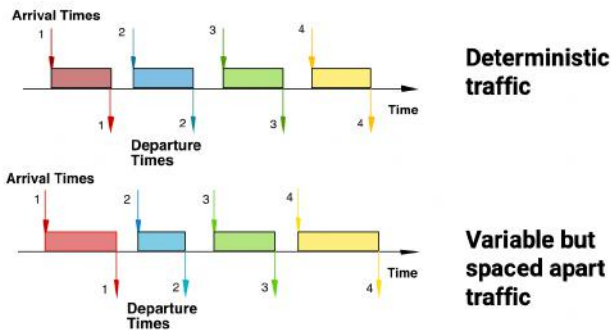
- Bij de stroomanalyse wordt geen rekening gehouden met wachttijden als gevolg van conflicterende middelen
- Analyse en simulatie van wachtrijen pakken deze beperkingen aan en zijn breder toepasbaar.

Waarom is wachtrijanalyse belangrijk?

- Capaciteitsproblemen zijn heel gewoon in de industrie en een van de belangrijkste drijfveren voor het herontwerpen van processen.
 - o Noodzaak om de kosten van verhoogde capaciteit af te wegen tegen de voordelen van verhoogde productiviteit en service.
- Wachtrij- en wachttijdanalyse is bijzonder belangrijk in dienstverleningssystemen
 - o Grote kosten van wachten en omzetverlies door wachten

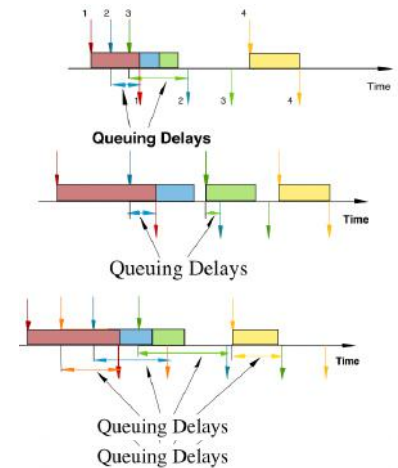
Vertraging wordt veroorzaakt door baaninterferentie

→ Als aankomsten regelmatig zijn of voldoende uit elkaar liggen, treedt er geen vertraging op.

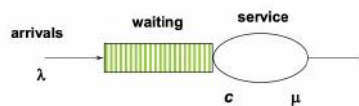


Oorzaken van interferentie tussen opdrachten

- Intervallende intervallen tussen aankomsten
 - o Periode waar je heel veel binnen krijgt en dan periode waar je niks binnen krijgt
- Variatie in taakgrootte
- Hoge bezettingsgraad
 - o Een systeem in de buurt van 100% bezetting is onstabiel



Wachtrijtheorie: basisbegrippen

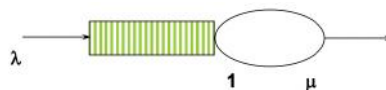


- λ (gemiddelde aankomstfrequentie) = gemiddeld aantal aankomsten per tijdseenheid
- μ (mean service rate) = gemiddeld aantal opdrachten dat door 1 server uitgevoerd kan worden per time unit
- c = aantal servers

Gegeven λ , μ en c , kunnen we berekenen:

- Bezettingsgraad: ρ
- W_q = gemiddelde tijd in de wachtrij
- W = gemiddelde tijd in het systeem (d.w.z. cyclustijd)
- L_q = gemiddeld aantal in de wachtrij (d.w.z. lengte van de wachtrij)
- L = gemiddeld aantal in het systeem (d.w.z. onderhanden werk)

Meest basale systeem: M/M/1 wachtrij



Assumpties:

- De tijd tussen aankomsten en de dienstdtijd volgen een negatieve exponentiële verdeling
- 1 server ($c=1$)
- FIFO

$$\rho = \frac{\text{Capacity Demand}}{\text{Available Capacity}} = \frac{\lambda}{\mu}$$

$$L = \rho / (1 - \rho)$$

$$W = L / \lambda = 1 / (\mu - \lambda)$$

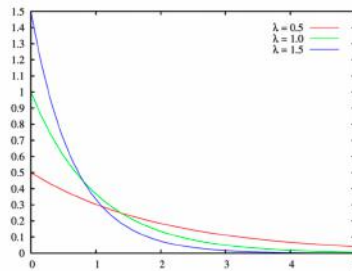
$$L_q = \rho^2 / (1 - \rho) = L \cdot \rho$$

$$W_q = L_q / \lambda = \lambda / (\mu(\mu - \lambda))$$

Het Poisson-proces

- Veel voorkomende aankomstaanne in veel wachtrij- en simulatiemodellen
- De tijden tussen aankomsten zijn onafhankelijk, identiek verdeeld en exponentieel
 - o $P(\text{aankomst} < t) = 1 - e^{-\lambda t}$
- Belangrijkste eigenschap: Het feit dat een bepaalde gebeurtenis nog niet heeft plaatsgevonden zegt niets over hoe lang het nog duurt voordat die gebeurtenis plaatsvindt (geheugenloosheid)
 - o bijv. $P(X > 40 | X >= 30) = P(X > 10)$

Negatief exponentiële verdeling



M/M/c wachtrij

- Nu staan er c servers parallel, dus de verwachte capaciteit per tijdseenheid is dan $c * \mu$

$$\rho = \frac{\text{Capacity Demand}}{\text{Available Capacity}} = \frac{\lambda}{c * \mu}$$

⇒ Hulpmiddelen : voor M/M/c-systemen is de exacte berekening van L_q nogal complex...

$$L_q = \sum_{n=c}^{\infty} (n-c) P_n = \dots = \frac{(\lambda/\mu)^c \rho}{c!(1-\rho)^2} P_0$$

$$P_0 = \left(\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!} \cdot \frac{1}{1-(\lambda/(c\mu))} \right)^{-1}$$

Overweeg het gebruik van een instrument, bijv.

- <http://apps.business.ualberta.ca/aingolfsson/qtp/>
- <http://www.stat.auckland.ac.nz/~stats255/qsim/qsim.html>

Voorbeeld - Ziekenhuis ER

- Situatie
 - o Patiënten arriveren volgens een Poisson proces met intensiteit λ (⇔ de tijd tussen aankomsten is $\exp(\lambda)$ verdeeld.
 - o De diensttijd (de onderzoeks- en behandelingstijd van een patiënt door een arts) volgt een exponentiële verdeling met gemiddelde $1/\mu$ (= $\exp(\mu)$ verdeeld)
 - o De ER kan worden gemodelleerd als een M/M/c-systeem waarbij c=het aantal artsen
- Gegevens
 - o $\lambda = 2$ patiënten per uur
 - o $\mu = 3$ patiënten per uur
 - o
- Vraag
 - o Moet de capaciteit worden verhoogd van 1 naar 2 artsen?

Analyse van wachtrijen - Ziekenhuis scenario

- Interpretatie
 - o In de wachtrij staan = in de wachtkamer staan
 - o In het systeem zitten = op de spoedafdeling zitten (wachten of in behandeling zijn)
- Is het gerechtvaardigd om een tweede dokter aan te nemen ?

Characteristic	One doctor (c=1)	Two Doctors (c=2)
ρ	2/3	1/3
L_q	4/3 patients	1/12 patients
L	2 patients	3/4 patients
W_q	2/3 h = 40 minutes	1/24 h = 2.5 minutes
W	1 h	3/8 h = 22.5 minutes

Beperkingen van basis wachtrijmodellen

- Kan worden gebruikt voor het analyseren van wachttijden (en dus cyclustijden), maar niet voor kosten- of kwaliteitsmetingen.
- Geschikt voor het analyseren van één enkele activiteit tegelijk, uitgevoerd door één enkele resource pool. Niet geschikt voor het analyseren van end-to-end processen bestaande uit meerdere activiteiten uitgevoerd door meerdere resource pools.
- Over het algemeen niet toepasbaar wanneer het systeem parallelle activiteiten omvat
- Gaat uit van "steady-state" (alleen geldig voor "lange termijn" analyse)

4. Simulation

Processimulatie

- Processimulatie is een veelzijdigere kwantitatieve analysemethode (ook populairder)
- Processimulatie = laat een groot aantal (hypothetische) procesinstanties lopen, verzamel gegevens (kosten, duur, gebruik van middelen) en bereken statistieken op basis van de output.

Stappen in de evaluatie van een proces met simulatie

- o Modelleer het proces (bijv. BPMN)
- o Verbeter het procesmodel met simulatie-info (meest complexe stap) → simulatiemodel
 - Op basis van aannames of beter op basis van gegevens (logs)
- o De simulatie uitvoeren
- o Analyseer de output van de simulatie
 - Procesduur- en kostenstatistieken en histogrammen
 - Wachttijden (per activiteit)
 - Gebruik van middelen (per middel)
- o Herhalen voor alternatieve scenario's

Elementen van een simulatiemodel (stap 2)

- Het procesmodel met inbegrip van:
 - o Gebeurtenissen, activiteiten, control-flow relaties (flows, gateways)
 - o Hulpmiddelenklassen/ resource classes (d.w.z. banen)
- Toewijzing van middelen
 - o Mapping van activiteiten naar middelenklassen
- Verwerkingstijden
 - o Per activiteit of per activiteit-middelenpaar
- Kosten
 - o Per activiteit en/of per activiteit/middelenpaar
- Aankomstsnelheid van procesinstanties
- Voorwaardelijke vertakkingskansen (XOR-poorten)

Moeilijkheden van de simulatie

- Talrijke keuzes die moeten worden gemaakt
- Vereenvoudigende veronderstellingen noodzakelijk
 - o Bv. menselijke deelnemers ≠ robots

- Betrouwbaarheid hangt grotendeels af van de nauwkeurigheid van de invoer
 - o Gebruik van reële gegevens is vaak nuttig
- Meerdere validatielussen nodig met belanghebbenden van het proces om de geldigheid te verifiëren
- Gevoelighedsanalyse (met een klein verandering in inputs kan je iets helemaal verschillend krijgen)

5. Conclusion

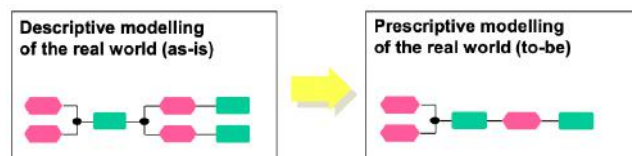
- Processen kunnen worden bestudeerd vanuit een kwantitatief perspectief
 - o Stromingsanalyse
 - o Wachtrijtheorie
 - o Simulatie
- Al deze methoden hebben voor- en nadelen - Er zijn meer veelzijdige simulatietechnieken en instrumenten beschikbaar
- Discrete Gebeurtenissimulatie
 - o Arena
 - o CPN-tools

BPM9. Process Redesign

1. Introduction

Herontwerp van processen

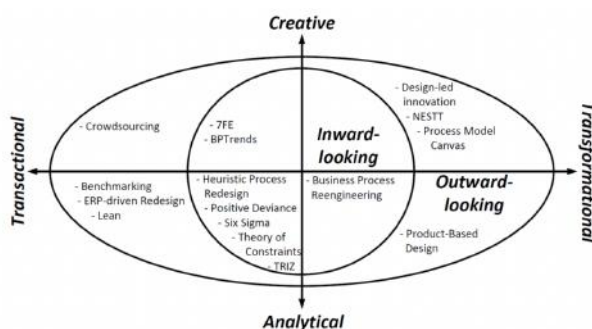
- Doel: Mogelijkheden identificeren om het ontwerp van een proces: "as is" → "to be"



→ We willen een prescriptief model

- Geen wondermiddel: vereist creativiteit
- Een grote verscheidenheid aan methoden, technieken en instrumenten

Het spectrum van methoden voor procesherontwerp



Creatief vs. Analytical

Transactioneel vs. Transformationeel

Inward-looking (gebruik van inzichten binnen het proces) vs. Outward-looking (inzichten buiten het proces gebruiken)

Benaderingen voor procesherontwerp

- Verkennend herontwerp/ exploratief redesign (transformationeel)
 - o Stelt de fundamentele veronderstellingen en principes van de bestaande processtructuur ter discussie
 - o Streeft naar doorbraakinnovatie
 - o Je mag wijzigen wat je wil wijzigen
 - o Voorbeeld: Herontwerpen van bedrijfsprocessen (BPR)
- Exploitatief herontwerp (transactioneel)

- Stelt de huidige processtructuur niet ter discussie
- Streeft ernaar problemen te identificeren en ze stapsgewijs, stap voor stap, op te lossen.
- Voorbeeld: Heuristisch herontwerp

2. Business Process Reengineering

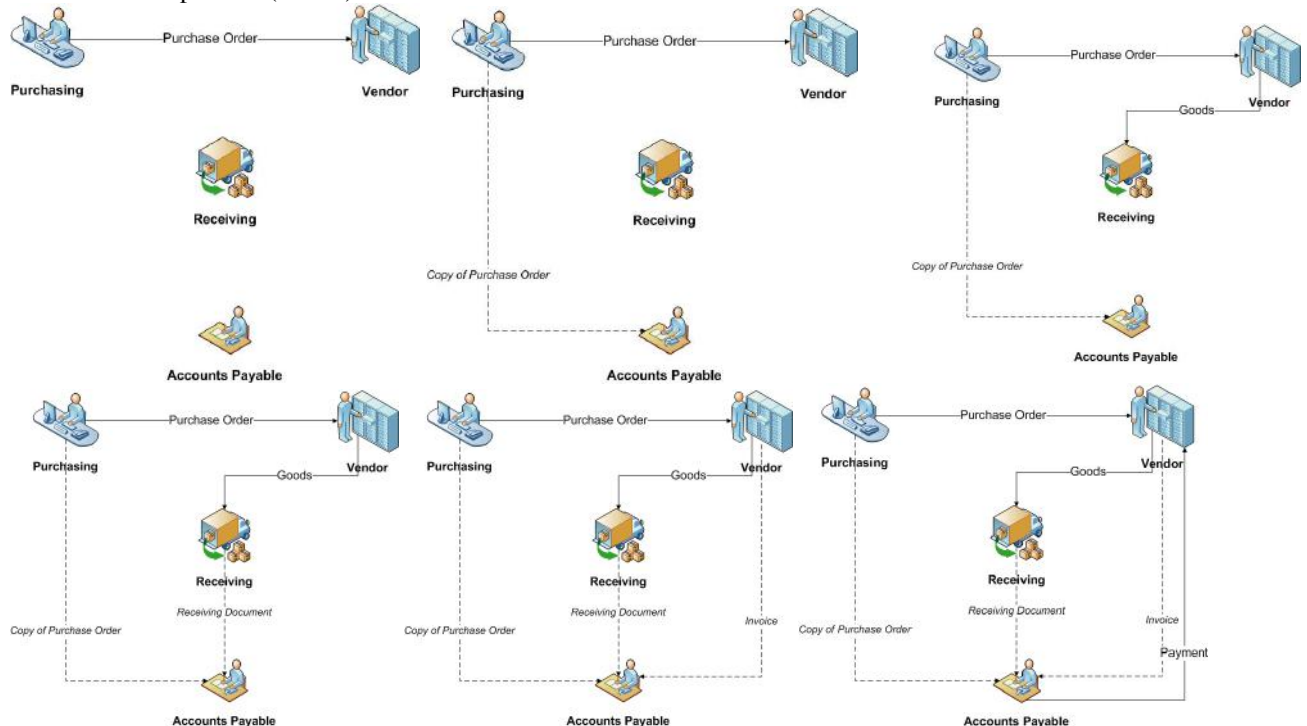
Herontwerpen van bedrijfsprocessen (BPR)

- Transformatief: Stelt de fundamentele veronderstellingen van het "as is"-proces ter discussie
- Analytisch: Gebaseerd op een reeks principes die
 - Resultaatgerichte processen bevorderen
 - Integratie van informatievergaring, werk en beslissingen
- Inward-looking

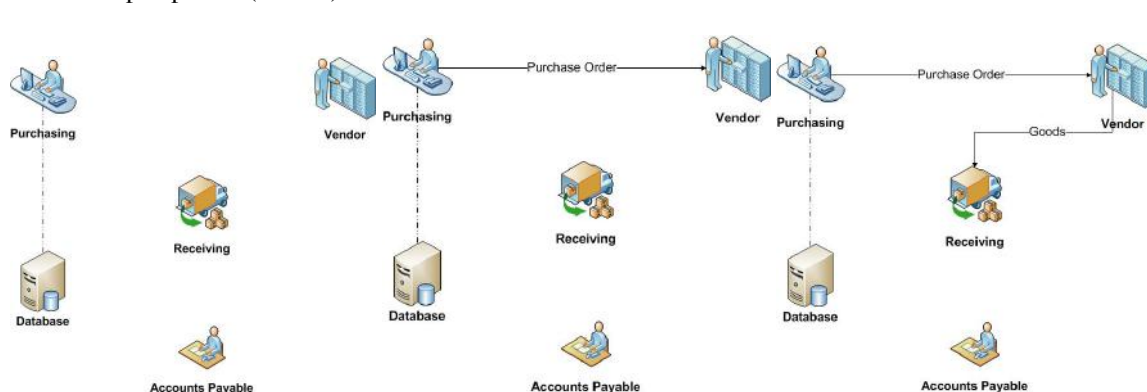
De Ford-casus

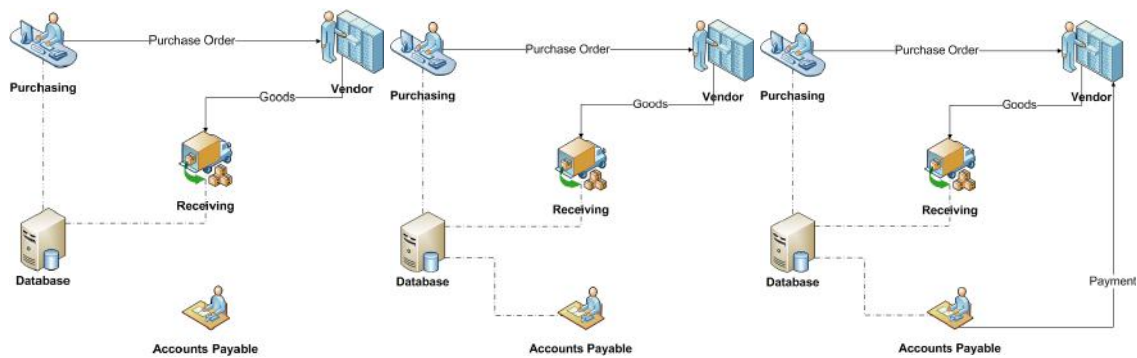
- Ford moest zijn inkoopproces herzien om:
 - Het goedkoper te doen (kosten besparen)
 - Het sneller te doen (doorlooptijden verkorten)
 - Het beter te doen (minder fouten)
- De crediteurenadministratie in Noord-Amerika alleen al had > 500 mensen in dienst en de doorlooptijden voor het verwerken van PO's en facturen lagen in de orde van weken.

Hoe werkte het proces? ("as is")



Herontworpen proces ("to be")





Resultaat...

- 75% minder personeel
- Eenvoudiger materiaalcontrole
- Meer accurate financiële informatie
- Snellere inkoopaanvragen
- Minder achterstallige betalingen
- Lessen:
 - o Waarom iets automatiseren wat helemaal niet nodig is?
 - o Automatiseer dingen die gedaan moeten worden.
- « Automatiseer niet, vernietig ! » (Hammer, 1990)

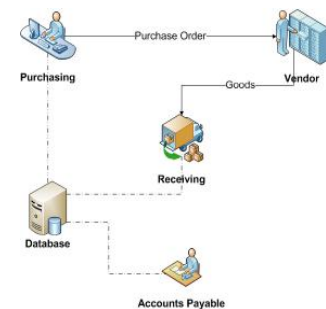
Enkele beginselen/principes van BPR

1. Leg informatie eenmalig en bij de bron vast
2. Informatieverwerkingswerk onderbrengen in het echte werk dat de informatie produceert
3. Laat degenen die de output van het proces gebruiken het proces sturen
4. Plaats het beslissingspunt waar het werk wordt uitgevoerd, en bouw controle in het proces in.
5. Behandel geografisch verspreide middelen alsof ze gecentraliseerd zijn.

Principe 1

"Informatie eenmalig en bij de bron vastleggen"

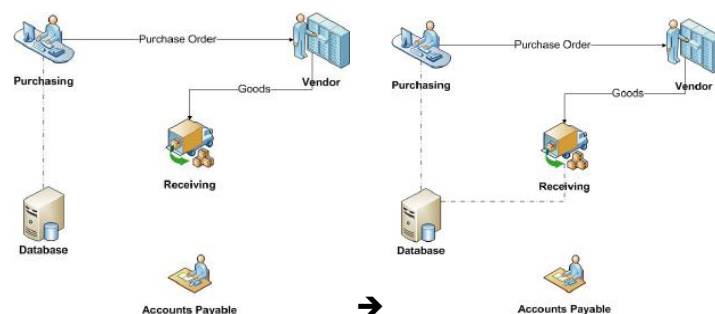
- Gedeelde gegevensopslag
 - o Alle proceswerkers hebben toegang tot dezelfde gegevens
 - o Stuur gegevens niet rond, maar deel ze!
- Zelfbediening
 - o Klanten leggen zelf gegevens vast
 - o Klanten voeren zelf taken uit (bv. documenten verzamelen)



Principe 2

"Informatieverwerkingswerk onderbrengen in het echte werk"

- Geëvalueerde ontvangstafwikkeling: bij ontvangst van de producten de vervulling van de PO vastleggen, die tot betaling leidt.



Principe 3

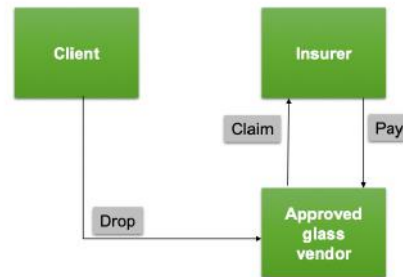
"Laat degenen die de output van het proces gebruiken het proces sturen".

- Doorleveranciers beheerde inventaris
- Op scannen gebaseerde handel
- Duw werk naar de actor die de stimulans heeft om het te doen.

Voorbeeld: problematische claimprocedure



Herontworpen claimproces

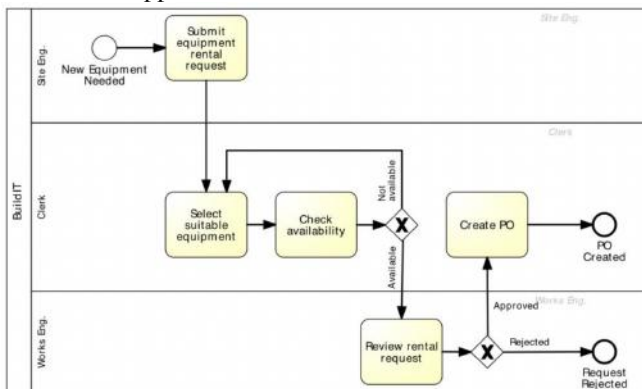


Principe 4

"Plaats het beslissingspunt waar het werk wordt uitgevoerd en bouw controle in het proces in".

- Geef de procesmedewerkers zeggenschap
 - o Geen aparte dienst om te controleren
- Voorzie de procesmedewerkers van informatie die nodig is om zelf beslissingen te nemen
- Vervang heen-en-weer-overdrachten tussen arbeiders en managers (transportverspilling) door goed ontworpen controles.

Verhuur van apparatuur



→Op zelfbediening gebaseerd herontwerp

- Principles 1 & 2
 - o Wanneer materiaal nodig is, raadpleegt de werfingenieur de catalogus van de leveranciers, selecteert het materiaal en activeert de PO.
- Principe 3
 - o Leverancier slaat vaak gebruikt materieel op de bouwplaats op, monteurs scannen om het in gebruik te nemen
- Principe 4
 - o Werfingenieur krijgt de bevoegdheid om het materieel te huren; werfingenieur voert statistische controles uit

Principe 5

"Behandel geografisch verspreide bronnen alsof ze gecentraliseerd zijn."

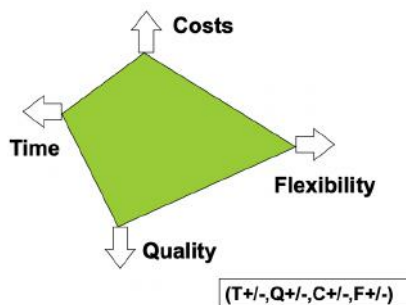
- Als dezelfde mensen dezelfde functie uitvoeren op verschillende locaties, integreer en deel hun werk dan waar mogelijk
- Grotere middelenpools → minder wachttijden, zelfs bij relatief hoog middelengebruik

3. Heuristic process redesign

Heuristisch procesherontwerp

- **Transactioneel:** verandert het "huidige" proces stapsgewijs
- **Naar binnen gericht/ inward-looking:** werkt binnen de reikwijdte en context van het "as is"-proces
- **Analytisch:** gebaseerd op herontwerp heuristieken die afwegingen maken tussen:
 - o Kosten
 - o Tijd
 - o Kwaliteit
 - o Flexibiliteit

The Devil's Quadrangle



Je wil de driehoek zo groot mogelijk maken, maar als je aan een hoek trekt dan gaat een van de andere hoeken naar binnen springen.

T, Q, C, F

- Effect op tijd, kwaliteit, kosten en flexibiliteit
 - o Zoals blijkt uit een onderzoek onder BPR praktijkmensen
 - o Studie van Mansar en Reijers (2005)
- Betekenis:
 - o T+: positief effect op tijd → verminderde cyclustijd
 - o Q+: verbeterde kwaliteit → minder fouten, storingen
 - o C+: positief effect op kosten → daling van proceskosten
 - o F+: verbeterde flexibiliteit → beter in staat om te reageren op veranderingen of onverwachte omstandigheden

$(T+/-, Q+/-, C+/-, F+/-)$

Herontwerpcriterium 1: Tijd

- De doorlooptijd bestaat uit:
 - o servicetijd (inclusief opzetten)
 - o transporttijd (kan vaak worden teruggebracht tot 0)
 - o wachttijd/wachtrij
 - delen van middelen (beperkte capaciteit)
 - externe communicatie (triggertijd)
- Er zijn verschillende manieren om de doorvoer/wachttijd te evalueren:
 - o Gemiddelde
 - o Variantie
 - o Serviceniveau
 - o vermogen om vervaldata te halen

Herontwerp criterium 2: Kwaliteit

- Extern: tevredenheid van de klant
 - o Product: product voldoet aan specificatie/verwachting.
 - o Proces: de manier waarop het product wordt geleverd (serviceniveau)
- Intern: werkomstandigheden
 - o Uitdagend
 - o Wisselend
 - o Beheersing

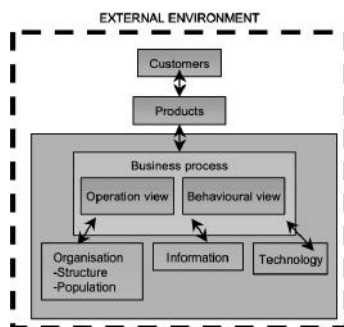
Herontwerpcriterium 3: Kosten

- Soort kosten
 - o vast of variabel,
 - o menselijk, systeem (hardware/software) of extern;
 - o verwerking, beheer of ondersteuning.
- Let op de afweging tussen menselijke en systeemgerelateerde kosten.

Herontwerpcriterium 4: Flexibiliteit

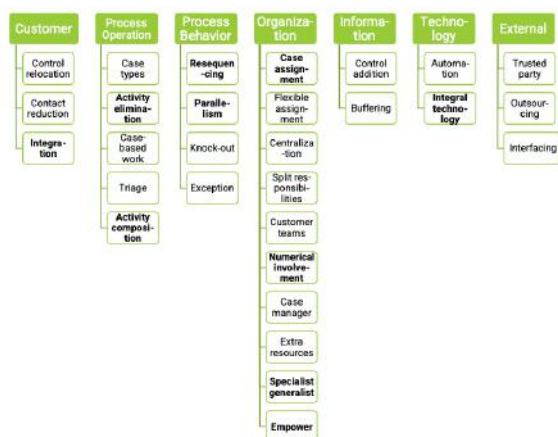
- Het vermogen om op veranderingen te reageren
- Flexibiliteit van
 - o middelen (vermogen om veel taken/nieuwe taken uit te voeren. – specialisten/ generalisten)
 - o proces (vermogen om verschillende gevallen en veranderende werklast te behandelen)
 - o beheer (vermogen om regels/toewijzing te veranderen)
 - o organisatie (vermogen om de structuur te veranderen en in te spelen op wensen van de markt en zakenpartners)

Heuristisch kader voor herontwerp (hoef je niet in detail te kennen)



- Mengt elementen van WCA, CIMOSA en MOBILE kaders.
- Is gevalideerd aan de hand van twee case studies en een vragenlijst onder meer dan 100 Nederlandse en Britse BPR experts.
- Classificeert verzamelde best practices

In het kader geplaatste heuristieken (niet uit de hoofd kennen)



→ voor degenene in vet moet je een redenering kunnen geven in termen van de devils quadrangle

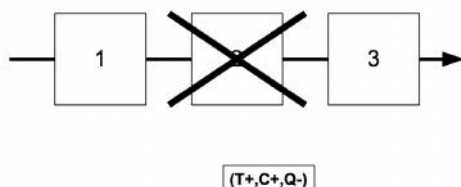
Heuristiek gedefinieerd (1) Moet je niet allemaal uit jouw hoofd kennen

Framework elements	Best practice name	Definition
Customers	Control relocation	Move controls towards the customer
	Contact reduction	Reduce the number of contacts with customers and third parties
	Integration	Consider the integration with a business process of the customer or a supplier
Products	None	
Operation view	Order types	Determine whether tasks are related to the same type of order and, if necessary, distinguish new business processes
	Task elimination	Eliminate unnecessary tasks from a business
	Order-based work	Consider removing batch-processing and periodic activities from a business process
	Triage	'Consider the division of a general task into two or more alternative tasks' or 'consider the integration of two or more alternative tasks into one general task'
	Task composition	Combine small tasks into composite tasks and divide large tasks into workable smaller tasks
Behavioural view	Resequencing	Move tasks to more appropriate places
	Knock-out	Order knockout decisions in a decreasing order of effort and in an increasing order of termination probability
	Parallelism	Consider whether tasks may be executed in parallel
	Exception	Design business processes for typical orders and isolate exceptional orders from normal flow
External environment	Trusted party	Instead of determining information oneself, use results of a trusted party
	Outsourcing	Consider outsourcing a business process in whole or parts of it
	Interfacing	Consider a standardised interface with customers and partners
Organisation: structure	Order assignment	Let workers perform as many steps as possible for single orders
	Flexible assignment	Assign resources in such a way that maximal flexibility is preserved for the near future
	Centralisation	Treat geographically dispersed resources as if they are centralised
	Split responsibilities	Avoid assignment of task responsibilities to people from different functional units
	Customer teams	Consider assigning teams out of different departmental workers that will take care of the complete handling of specific sorts of orders
	Numerical involvement	Minimise the number of departments, groups and persons involved in a business process
	Case manager	Appoint one person as responsible for the handling of each type of order, the case manager
Organisation: population	Extra resources	If capacity is not sufficient, consider increasing the number of resources
	Specialist-generalist	Consider to make resources more specialised or more generalist
	Empower	Give workers most of the decision-making authority and reduce middle management
Information	Control addition	Check the completeness and correctness of incoming materials and check the output before it is sent to customers
	Buffering	Instead of requesting information from an external source, buffer it by subscribing to updates
Technology	Task automation	Consider automating tasks
	Integral technology	Try to elevate physical constraints in a business process by applying new technology

Meest populaire herontwerpheuristieken*

1) Taak eliminatie

= Onnodige taken uit een workflow verwijderen



→ Die T, C en Q is wat je verwacht maar is niet noodzakelijk zo

2) Integrale technologie

- Fysieke beperkingen in een bedrijfsproces opheffen door nieuwe technologie toe te passen
 - o Nieuwe technologieën kunnen positieve effecten hebben op de vierhoeksdimensie van elke duivel (zeker op lange termijn)
 - o Aanschaf, ontwikkeling, implementatie en opleiding kunnen duur zijn
 - o Werknemers kunnen terughoudend zijn ten opzichte van nieuwe technologie, wat kan leiden tot kwaliteitsvermindering.

(T+,C-) (geen impact op flexibiliteit)

3) Taaksamenstelling

- Kleine taken combineren tot samengestelde taken en grote taken opdelen in werkbaarere kleinere delen

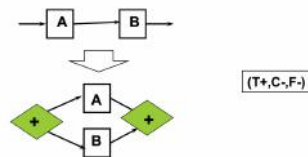


- Kleine taken combineren tot een grotere
 - o opstellingsvermindering, geen versnippering, meer inzet
 - o meer inzet, één persoon moet voor beide delen gekwalificeerd zijn

(T+,C+,F-)

4) Parallellisme

- Meer parallellisme leidt tot betere prestaties: kortere wachttijden en betere benutting van de capaciteit
- IT-infrastructuren die het delen van gegevens en werk mogelijk maken, maken parallellisme mogelijk.
- De kosten kunnen toenemen wanneer er knock-outs zijn
- Het beheer kan complexer worden (F-)



(T+,C-,F-)

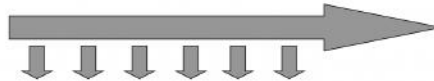
5) Specialist-generalist

- Overwegen om de vaardigheden van de middelen te verdiepen of te verbreden
- Meer specialisten:
 - o Snellere verwerking van activiteiten
 - o Hogere kwaliteit
- Meer generalisten:
 - o Meer flexibiliteit
 - o Betere benutting (snellere verwerking van zaken)

(T+,Q+,F-)

(T+,Q-,F+)

6) Resequencing

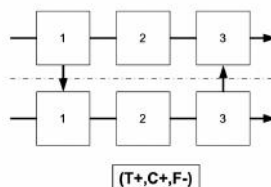


- Sorteer taken op basis van kosten/ffect
- Eerst "knock-out controles" uitvoeren - problemen vroegtijdig opsporen
 - o Knock-out heuristiek is een specifieke vorm van resequencing
- Stel dure taken uit tot het einde
- Met andere woorden: rangschik de taken op basis van de verhouding "kosten/ffect".

(T+,C+)

7) Integratie

- Overweeg de integratie met een workflow van een klant of een leverancier - Meer geïntegreerde workflows verbeteren de efficiëntie (zowel tijd als kosten)
- Flexibiliteit kan afnemen door wederzijdse afhankelijkheid



(T+,C+,F-)

8) Versterken

- Geef werknemers de meeste beslissingsbevoegdheid, in plaats van te vertrouwen op het middenkader
 - o Soepeler operaties met lagere doorlooptijden (geen escalatie nodig)
 - o Indien veel slechte beslissingen, meer herwerk, waardoor de kosten stijgen

(T+,Q-,F+)

9) Numerieke betrokkenheid

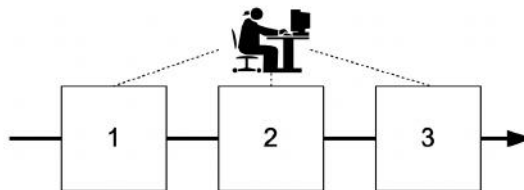
- Minimaliseer het aantal afdelingen, groepen en personen die betrokken zijn bij een bedrijfsproces
 - o Minder coördinatieproblemen, meer tijd voor verwerking
 - o Minder middelen beschikbaar, flexibiliteit neemt af

(T+,F-)

10) Case opdracht

- Laat werknemers zoveel mogelijk stappen uitvoeren voor afzonderlijke gevallen
 - o Verkort de insteltijd
 - o Kwaliteit verbeteren
- Flexibiliteit van de werkverdeling neemt aanzienlijk af
 - o Toename van de wachttijd (geen algemeen effect op de tijdsdimensie)

(Q+,F-)



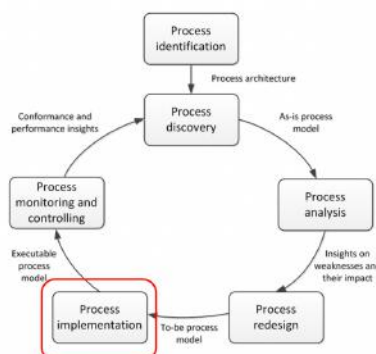
4. Conclusion

- Herontwerp is moeilijk, nooit definitief, maar cruciaal voor veel organisaties
- Geen wondermiddel!
- Herontwerpen betekent inherent "compromissen sluiten".

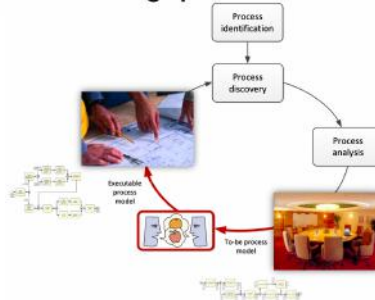
BPM10. Process Automation

1. Introduction

Waar zijn we ?



De bekende kloof...



Het resultaat: twee kanten van het verhaal

- Conceptuele "to-be" procesmodellen
 - o worden gemaakt door domeindeskundigen
 - o bieden een basis voor communicatie tussen relevante belanghebbenden
 - o moeten begrijpelijk zijn
 - o moeten intuïtief zijn en mogen ruimte laten voor interpretatie
 - o bevatten uitsluitend relevante procesinformatie
- Uitvoerbare procesmodellen
 - o worden gemaakt door IT-deskundigen
 - o dienen als input voor een procesuitvoeringssysteem – BPMS
 - o moeten machinaal leesbaar zijn
 - o moet ondubbelzinnig zijn en mag geen onzekerheden bevatten
 - o moeten verdere details bevatten die alleen relevant zijn voor de uitvoering

⇒ Tussenstap: “to be” processmodel

2. Bridging the gap: A five step approach

De kloof overbruggen: één taak per keer

1) Bepaal de grenzen van de automatisering

Principe: niet alle processen kunnen worden geautomatiseerd.

→ Begin met het identificeren van het type taak:



In BPMN: taakmarkeringen specificeren



2) Herzie de handmatige taken

Principe: als het niet door het BPMS kan worden gezien, bestaat het niet.

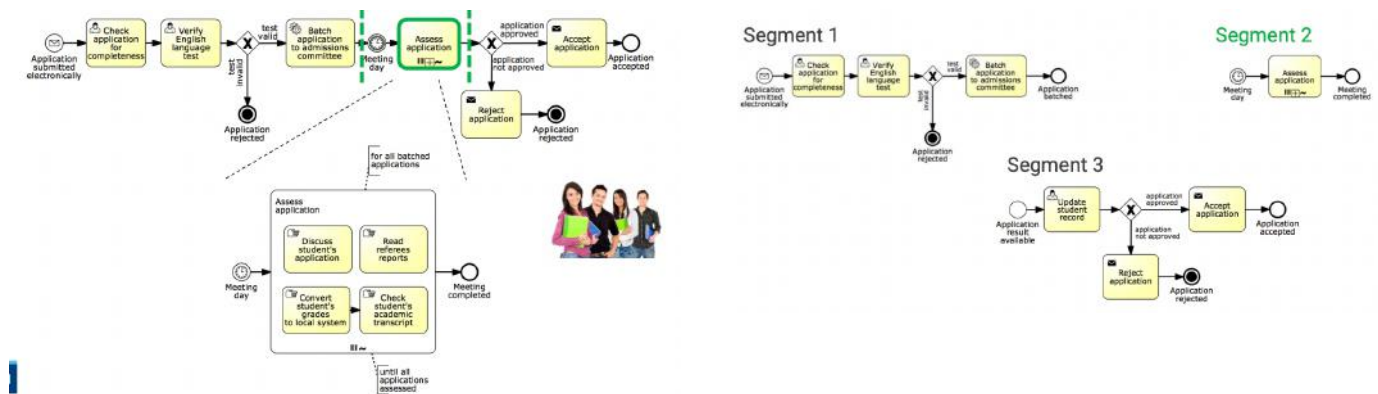
→ Zoek manieren om handmatige taken via IT te ondersteunen:

- via gebruikerstaak/ usertask
 - o Het manuele taak blijft een manuele taak maar wordt zichtbaar voor het systeem
- via geautomatiseerde taak

→ Isoleer ze en automatiseer de rest



Alternatief: handmatige taken isoleren



BPMN-elementen die niet relevant zijn voor de uitvoering

- Fysieke gegevensobjecten
- Berichten met fysieke gegevensobjecten
- Gegevensopslag (zowel fysiek als elektronisch)
- Pools & lanes
- Tekst annotaties

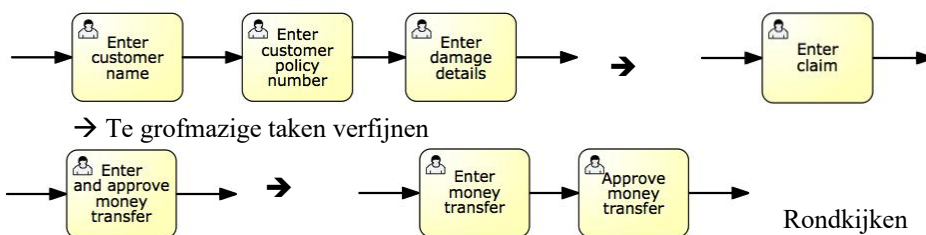
→ Verwijderen of verwaarlozen, afhankelijk van BPMS

3) Voltooi het procesmodel

- Principe: uitzonderingen zijn de regel.
→ Uitzonderingshandlers toevoegen
- Principe: geen gegevens = geen beslissingen, geen taakoverdracht.
→ Specificeer alle elektronische bedrijfsobjecten

4) Pas de taakgranulariteit aan

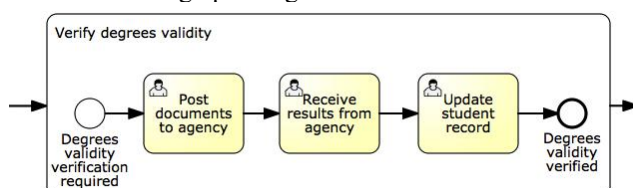
- Principe: BPMS'en voegen waarde toe als zij overdrachten van werk tussen hulpmiddelen coördineren.
→ Voeg twee opeenvolgende taken samen die aan hetzelfde middel zijn toegewezen



→ Te grofmazige taken verfijnen

Rondkijken
- Kandidaat-taken voor aggregatie hoeven niet noodzakelijk opeenvolgend te zijn wegens een suboptimale volgorde van taken in het conceptuele model.

Een uitzondering op de regel



5) Specificeer uitvoeringseigenschappen (execution properties)

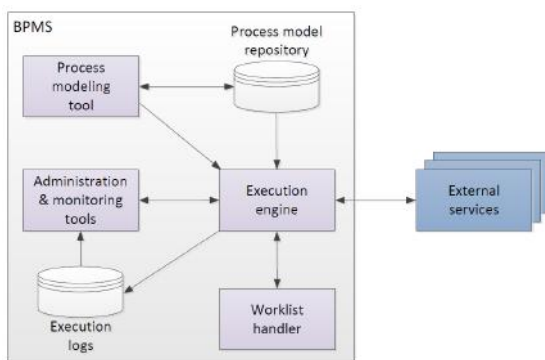
- Procesvariabelen, berichten, signalen, fouten
- Taak- en gebeurtenisvariabelen en hun toewijzingen aan procesvariabelen
- Details van de dienst
- Codefragmenten
- Regels voor deelnemerstoewijzing en gebruikersinterfacestructuur
- Taak, gebeurtenis en sequentie flow expressies
- BPMS-specifiek: werkwachtrijen, formulieren, connectoren...

Spiekbriefje

1. Controle flow → specificeer de regels voor de toewijzing van deelnemers, de details van de dienst...
2. Datastroom. → specificeer sequence flow expressies.
3. Resources. → specificeer datatypes en datamappings

3. Business Process Management Systems (BPMS)

BPMS - Algemene Architectuur



Hulpmiddel voor procesmodellering

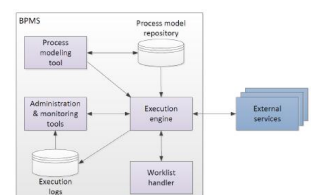
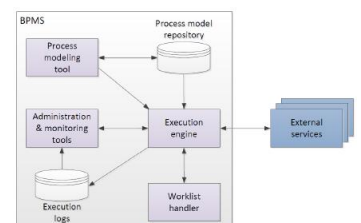
- Uitvoerbare procesmodellen maken en wijzigen (door uitvoeringseigenschappen op te geven)
- Om automatiseringsoplossingen op te slaan en op te halen uit een procesmodelopslagplaats
- Kan importeren uit hulpmiddelen voor conceptuele procesmodellering

Uitvoeringsprogramma

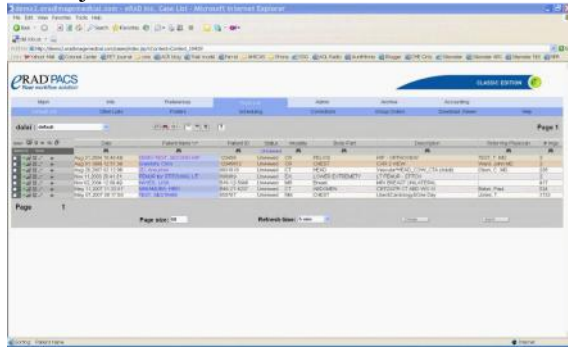
- Implementeert uitvoerbare procesmodellen (ook wel "cases" genoemd)
- Orkestreert de distributie van werkitems naar procesdeelnemers en softwarediensten om een bedrijfsproces van begin tot eind uit te voeren
- Logt uitvoeringsgegevens

Werklijst Handler

- Zie het als een "inbox"
- Biedt werkitems aan deelnemers aan en laat deelnemers toe zich aan deze werkitems te binden
- Behandelt de werkwachtrijen en werkprioriteiten van deelnemers
- Kan sociale netwerkmogelijkheden bieden



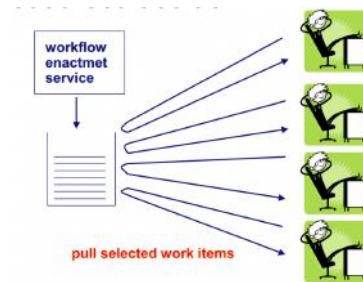
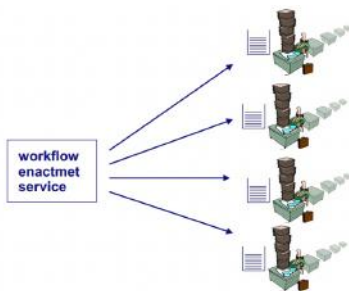
Werklijst



Werkverdeling: push vs. pull



Push: selecteer een "slachtoffer" (bepaald door systeem) Pull: de middelen beslissen (ressources beslissen)

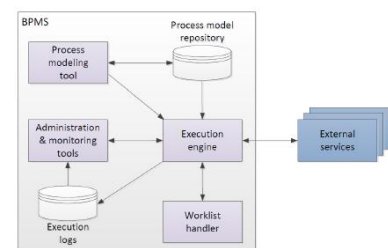


Push vs. Pull

- Pull controle
 - o Bepaald door mensen, d.w.z. "race" tussen middelen
 - o Selecteren om te starten en/of selecteren om werk toe te wijzen
- Push controle
 - o (Gewogen) round robin
 - o Kortste wachtrij
 - o Selecteer de meest geschikte (gespecialiseerde) bron

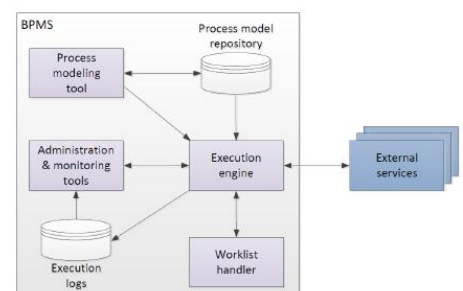
Beheer- en bewakingstools

- Automatiseringsoplossingen beheren
- Toegang tot systeemonderdelen configureren
- Om de beschikbaarheid van deelnemers en de prestaties van procesgevallen te bewaken



Externe diensten

- Stel een service-interface beschikbaar waarmee de engine kan communiceren.
- De engine voorziet de aangeroepen service van de nodige gegevens die het nodig heeft om de activiteit voor een specifiek geval uit te voeren.
- Voorbeelden: rules engine, e-mail of Twitter notificatie, DB connector, CRM connector...



BPMS-landschap

Big vendors	Other closed-source	Commercial open-source	Community open-source
<ul style="list-style-type: none"> • IBM BPM • Oracle BPMS • Microsoft Power Automate • SAP NetWeaver BPM • Software AG webMethods • Pagaystems PegaRULES 	<ul style="list-style-type: none"> • Appian BPMS • BizAgi BPM Suite • Bosch inubit Suite • OpenText BPM • Perceptive BPM ONE • Progress Savvion • TIBCO ActiveMatrix BPM 	<ul style="list-style-type: none"> • Bonita Open Solution • Camunda Fox • Intalio BPM • JBoss jBPM 	<ul style="list-style-type: none"> • Shark • YAWL

BPMS-classificatie volgens BPMN-ondersteuning

1. **Pure BPMN:** (her)ontworpen vanaf de grond om de specs naar de letter te volgen.
 - IBM BPM, Appian BPMS, Camunda Fox.
2. **Aangepaste BPMN:** gebruiken een BPMN skin maar vertrouwen op interne representatie - dateren van voor BPMN.
 - Bonita Open Solution, BizAgi BPM Suite
3. **Niet BPMN:** eigen taal en semantiek
 - Bosch inubit Suite, BPMOne, YAWL

BPM11. Process Mining – Process Discovery

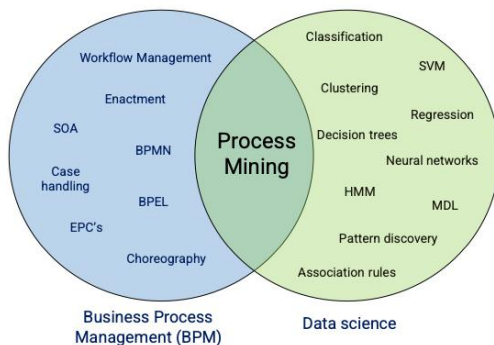
1. Introduction

Gegevens blijven groeien:

→ Voor beslissingen wordt steeds meer van data gebruikt gemaakt

Maar niet alleen gegevens, ook processen zijn van belang !

⇒ Process Discovery : op het snijvlak van BPM (processes) en data science



⇒ Naar data kijken met een process perspectief

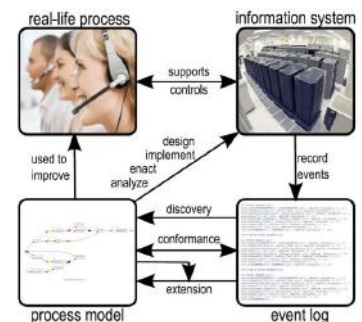
De drie hoofdtypen van process mining: ontdekking, conformiteit en uitbreiding

→ Meeste processen zijn ondersteund door een informatiesysteem.

Assumptie om process mining te gebruiken: processen zijn ondersteund door een informatiesysteem.

Als we dus geen event data hebben kunnen we niet gebruik maken van process mining tools : Discovery, conformance en extensions technieken.

- Discovery neemt event log als input en een process model als output
- Conformance neemt: beide event log en processmodel worden als input genomen waarbij we een contrast analyse doen.



- ⇒ Doel: de process verbeteren

→ om process mining te starten

[illegible]

- 1 Case ID
- 2 Activity name
- 3 Timestamp

The diagram illustrates the Business Process Management (BPM) cycle. It features three main stages in blue boxes: **Process design**, **Process implementation**, and **Process execution**. These stages are connected by yellow arrows labeled **GAF 1** and **GAF 2**. A large green arrow labeled **Process Mining** points from the **Process execution** stage back to the **Process design** stage, indicating a feedback loop. Above the stages are three images: **Bedrijfs-analysen** (Business analysis) showing a meeting, **IT-systeem** (IT system) showing a server room, and **Event log** showing a table of log entries.

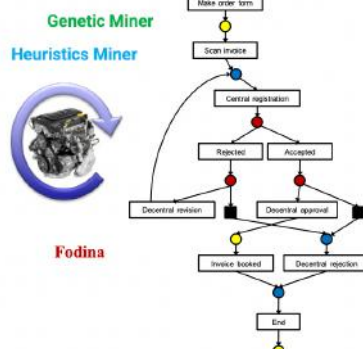
Time	Activity	Duration	Start Time	End Time
10:00	Start	0:00	10:00	10:00
10:05	Check email	0:05	10:05	10:10
10:10	Open system	0:05	10:10	10:15
10:15	Log in	0:05	10:15	10:20
10:20	Start process	0:05	10:20	10:25
10:25	Process step 1	0:05	10:25	10:30
10:30	Process step 2	0:05	10:30	10:35
10:35	Process step 3	0:05	10:35	10:40
10:40	Process step 4	0:05	10:40	10:45
10:45	Process step 5	0:05	10:45	10:50
10:50	Process step 6	0:05	10:50	10:55
10:55	Process step 7	0:05	10:55	11:00
11:00	Process step 8	0:05	11:00	11:05
11:05	Process step 9	0:05	11:05	11:10
11:10	Process step 10	0:05	11:10	11:15
11:15	Process step 11	0:05	11:15	11:20
11:20	Process step 12	0:05	11:20	11:25
11:25	Process step 13	0:05	11:25	11:30
11:30	Process step 14	0:05	11:30	11:35
11:35	Process step 15	0:05	11:35	11:40
11:40	Process step 16	0:05	11:40	11:45
11:45	Process step 17	0:05	11:45	11:50
11:50	Process step 18	0:05	11:50	11:55
11:55	Process step 19	0:05	11:55	12:00
12:00	Process step 20	0:05	12:00	12:05
12:05	Process step 21	0:05	12:05	12:10
12:10	Process step 22	0:05	12:10	12:15
12:15	Process step 23	0:05	12:15	12:20
12:20	Process step 24	0:05	12:20	12:25
12:25	Process step 25	0:05	12:25	12:30
12:30	Process step 26	0:05	12:30	12:35
12:35	Process step 27	0:05	12:35	12:40
12:40	Process step 28	0:05	12:40	12:45
12:45	Process step 29	0:05	12:45	12:50
12:50	Process step 30	0:05	12:50	12:55
12:55	Process step 31	0:05	12:55	13:00
13:00	Process step 32	0:05	13:00	13:05
13:05	Process step 33	0:05	13:05	13:10
13:10	Process step 34	0:05	13:10	13:15
13:15	Process step 35	0:05	13:15	13:20
13:20	Process step 36	0:05	13:20	13:25
13:25	Process step 37	0:05	13:25	13:30
13:30	Process step 38	0:05	13:30	13:35
13:35	Process step 39	0:05	13:35	13:40
13:40	Process step 40	0:05	13:40	13:45
13:45	Process step 41	0:05	13:45	13:50
13:50	Process step 42	0:05	13:50	13:55
13:55	Process step 43	0:05	13:55	14:00
14:00	Process step 44	0:05	14:00	14:05
14:05	Process step 45	0:05	14:05	14:10
14:10	Process step 46	0:05	14:10	14:15
14:15	Process step 47	0:05	14:15	14:20
14:20	Process step 48	0:05	14:20	14:25
14:25	Process step 49	0:05	14:25	14:30
14:30	Process step 50	0:05	14:30	14:35
14:35	Process step 51	0:05	14:35	14:40
14:40	Process step 52	0:05	14:40	14:45
14:45	Process step 53	0:05	14:45	14:50
14:50	Process step 54	0:05	14:50	14:55
14:55	Process step 55	0:05	14:55	15:00
15:00	Process step 56	0:05	15:00	15:05
15:05	Process step 57	0:05	15:05	

2^{de} gap: hoe process is implemented en hoe het is executed. Er zijn mensen die afwijken van procedures of er zijn beperkingen die ervoor zorgen dat je het werk niet kan verrichten zoals het gepland was. (Het is niet noodzakelijk negatief)

⇒ Het ontdekken van deze gaps is de doel van processmining
want het laat toe om inefficiënties vast te stellen en deze dan te
verbeteren

Doel: van event log naar een processmodel (petri-net)

Case ID	Activity Name	Original	Timestamp	Extra Data
001	Make order form	Harry	20-07-2019 14:00:00	...
002	Make order form	John	20-07-2019 18:00:00	...
001	Scan invoice	John	10-08-2019 08:52:31	...
001	General registration	John	11-08-2019 10:00:00	...
002	Scan invoice	John	11-08-2019 08:15:22	...
002	General registration	John	11-08-2019 08:56:01	...
001	Accepted	Stephen	13-08-2019 08:00:00	...
002	Accepted	Stephen	13-08-2019 08:21:12	...
001	Disapproval rejection	Myron	14-08-2019 14:15:14	...
001	End	Sylvia	14-08-2019 14:15:15	...
002	Disapproval approval	John	16-08-2019 19:02:58	...
002	Invoice Issued	Sylvia	16-08-2019 16:22:59	...
001	End	Sylvia	16-08-2019 16:23:00	...
002	Make order form	Harry	19-08-2019 07:52:41	...
004	Make order form	Myron	19-08-2019 19:21:39	...

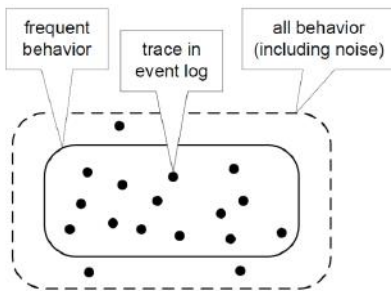


A central yellow box labeled "process discovery" is connected by four grey arrows to four surrounding terms: "fitness" (top-left), "simplicity" (top-right), "precision" (bottom-right), and "generalization" (bottom-left). Each term is accompanied by a quote: "able to replay event log" for fitness, "Occam's razor" for simplicity, "not underfitting the log" for precision, and "not overfitting the log" for generalization.

Als je bijvoorbeeld heel gedetailleerd werkt zal de fitness minder zijn want dan gaat het moeilijker zijn om alle events log te replay.

Abstract voorbeeld

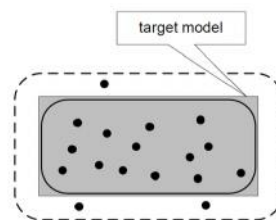
- Waargenomen gedrag van een stabiel proces over een oneindig lange periode



Elke punt stelt een variance voor.

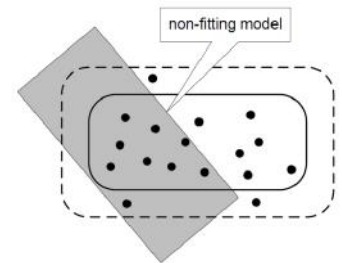
- ➔ In volledige vierkant frequente gedrag
- ➔ Buiten volledige vierkant: ook minder frequente gedrag die bijvoorbeeld door noise beïnvloed zijn.

Doelmodel



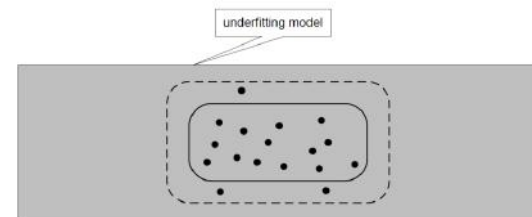
Geschiktheid

- Het ontdekte procesmodel moet het in het gebeurtenissenlogboek waargenomen gedrag mogelijk maken



Precisie

- Het ontdekte procesmodel mag geen gedrag toestaan dat volledig losstaat van wat in het gebeurtenissenlogboek is waargenomen.
- Het moet underfitting vermijden : het neemt de noise en gaat nog veel verder dan de noise (areas waar we zelf geen observaties hebben)

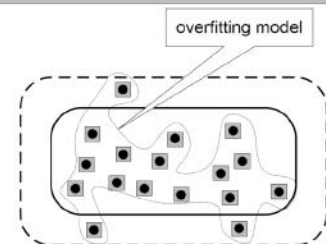


Generalisatie

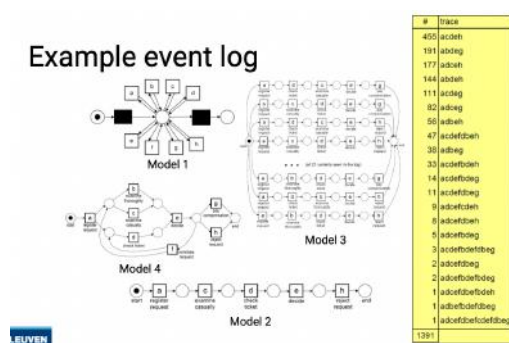
- Het ontdekte procesmodel moet het voorbeeldgedrag uit het gebeurtenissenlogboek generaliseren.
- Het moet overpassing vermijden

Eenvoud

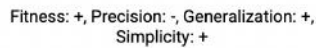
- Het ontdekte model moet zo eenvoudig mogelijk zijn
- Het scheermes van Occam: "Voor het overige is de eenvoudigste verklaring meestal de juiste".



Voorbeeld van een event log



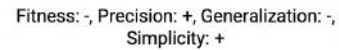
Model 1



#	trace
455	adodsh
191	ahidag
177	adidsh
144	adidsh
111	adidag
82	adidag
56	adidsh
47	adidshidsh
38	adidag
33	adidshidsh
14	adidshidag
11	adidshidag
9	adidshidsh
8	adidshidsh
5	adidshidag
3	adidshidshidag
2	adidshidag
2	adidshidshidag
1	adidshidshidsh
1	adidshidshidag
1	adidshidshidshidsh

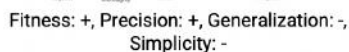
In deze model is alles toegestaan: je mag 10h doen ook als is dat niet in onze data. → Way to much behaviour

Model 2



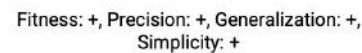
#	trace
456	acdeh
191	ebdeg
177	acdeh
144	abdeh
111	acdeg
82	acdeg
56	acbeh
47	acdefbdeh
38	acdeg
33	acdefbdeh
14	acdefbdeg
11	acdefbdeg
9	acdefcdeh
8	acdefbdeh
6	acdefbdeg
3	acdefbdefbdeg
2	acdefbdeg
2	acdefbdefbdeg
1	acdefbdefbdeh
1	acdefbdefbdeg
1	acdefbdefbdefbdeg

11 5



#	trace
456	acdeh
191	abdeag
177	acdeh
144	abdeh
111	acdeg
82	adeg
56	abdeh
47	acdefdebeh
38	addeg
33	acdefdebeh
14	acdefdebeh
11	acdefdebeh
9	acdefdebeh
8	acdefdebeh
5	acdefdebeh
3	acdefdebehdebeh
2	acdefdebehdebeh
1	acdefdebehdebeh
1	acdefdebehdebeh
1	acdefdebehdebehdebeh

2021 4



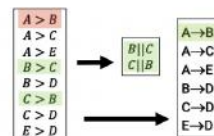
#	trace
455	acdeh
191	abdeq
177	acdeh
144	abdeh
111	acdeq
82	acdeq
56	acbeh
47	acdefdbeh
38	acdeq
33	acdefdeh
14	acdefdeq
11	acdefdeq
9	acdefdeh
8	acdefdeh
5	acdefdeq
3	acdefdeqdefdeq
2	acdefdeq
2	acdefdeqdefdeq
1	acdefdeqdefdeh
1	acdefdeqdefdeq
1	acdefdeqdefdeqdefdeq

α -algoritme

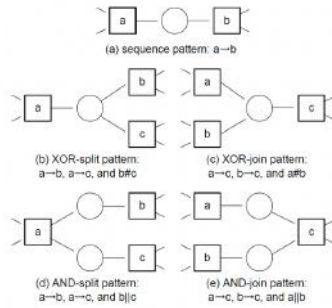
- α -algoritme - Ordeningsrelaties $>, \rightarrow, ||, \#$

ABCD
ACBD
AED

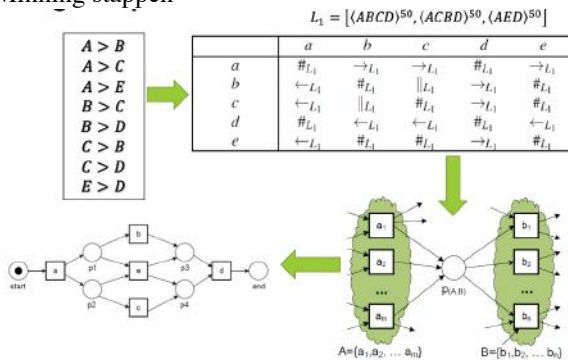
- **Direct succession:** $x > y$
iff for some case, x is directly followed by y .
- **Causality:** $x \rightarrow y$
iff $x > y$ and not $y > x$.
- **Parallel:** $x || y$
iff $x > y$ and $y > x$
- **Unrelated:** $x \# y$
iff not $x > y$ and not $y > x$.



Basis idee



Minning stappen

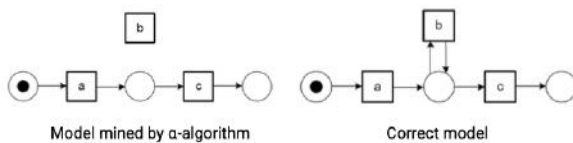


Beperkingen van het α -algorithm

- Problemen met representatieve vertekening
 - o Geen ontdekking van lussen van lengte één
 - o Geen ontdekking van lussen van lengte twee
 - o Geen ontdekking van niet-lokale afhankelijkheden
 - o Geen ontdekking van dubbele taken
 - o Geen ontdekking van stille/onzichtbare activiteiten (bv. overslaan)
- Ruis. (meestal bij echte data)
- Onvolledigheid (meestal bij echte data)

Lussen van lengte één

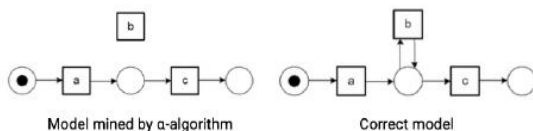
$$L_7 = [\langle a, c \rangle^2, \langle a, b, c \rangle^3, \langle a, b, b, c \rangle^2, \langle a, b, b, b, c \rangle^1]$$



$B > B$ and $\text{not } B > B$ implies $B \rightarrow B$, which is impossible

Lussen van lengte twee

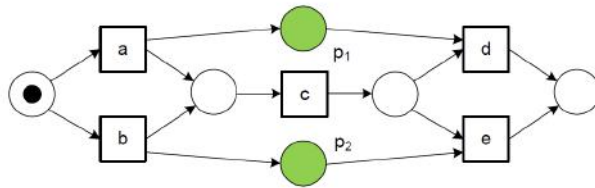
$$L_7 = [\langle a, c \rangle^2, \langle a, b, c \rangle^3, \langle a, b, b, c \rangle^2, \langle a, b, b, b, c \rangle^1]$$



$B > B$ and $\text{not } B > B$ implies $B \rightarrow B$, which is impossible

Niet-lokale afhankelijkheden

$$L_9 = [\langle a, c, d \rangle^{45}, \langle b, c, e \rangle^{42}]$$



Green places are not discovered!

4. Heuristic process discovery

Heuristische Miner

- Om met ruis (hoofdzakelijk) en onvolledigheid om te gaan
- Een betere representatieve bias hebben dan het α -algoritme
 - o Overslaan (onzichtbare taken)
 - o Niet-lokale afhankelijkheden
 - o OR-splitsingen en -verbindingen
- Gebruikt een andere procesmodelrepresentatie
 - o Eerste versie: Heuristische netten (ProM 5.2)
 - o Nieuwe versie: Causale netten of C-netten (ProM 6)
 - o Kan worden omgezet in Petri-net (om te vergelijken)

Afhankelijkheidsmeting

- Stap 1: Tellen van het aantal keren dat twee activiteiten elkaar direct opvolgen

$$\begin{array}{l} |a > b| = \text{the number of times activity } a \text{ is directly followed by activity } b \\ |b > a| = \text{the number of times activity } b \text{ is directly followed by activity } a \end{array}$$

- Stap 2: Berekening van de "afhankelijkheidsmaatregel" / "dependency measure" tussen taken a en b
 - o Als $a \neq b$: Afhankelijkheidsmaatstaf $(a, b) = \frac{|a > b| - |b > a|}{|a > b| + |b > a| + 1}$
 - o Als $a = b$: Afhankelijkheidsmaatstaf $(a, a) = \frac{|a > a|}{|a > a| + 1}$

Formules niet uit hoofd kennen !

Voorbeeld: de telling volgt direct

$$L = [\langle a, e \rangle^5, \langle a, b, c, e \rangle^{10}, \langle a, c, b, e \rangle^{10}, \langle a, b, e \rangle^1, \langle a, c, e \rangle^1, \langle a, d, e \rangle^{10}, \langle a, d, d, e \rangle^2, \langle a, d, d, d, e \rangle^1]$$

$ >_L $	a	b	c	d	e
a	0	11	11	13	5
b	0	0	10	0	11
c	0	10	0	0	11
d	0	0	0	4	13
e	0	0	0	0	0

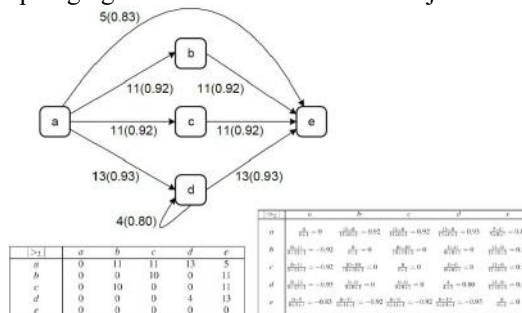
→ berekening van de afhankelijkheidsmaatregel

$ > $	a	b	c	d	e
a	0	11	11	13	5
b	0	0	10	0	11
c	0	10	0	0	11
d	0	0	0	4	13
e	0	0	0	0	0

$\Rightarrow L$	a	b	c	d	e
a	$\frac{0}{0+1} = 0$	$\frac{11-0}{11+0+1} = 0.92$	$\frac{11-0}{11+0+1} = 0.92$	$\frac{13-0}{13+0+1} = 0.93$	$\frac{5-0}{5+0+1} = 0.83$
b	$\frac{0-11}{0+11+1} = -0.92$	$\frac{0}{0+1} = 0$	$\frac{10-0}{10+0+1} = 0$	$\frac{0-0}{0+0+1} = 0$	$\frac{11-0}{11+0+1} = 0.92$
c	$\frac{0-11}{0+11+1} = -0.92$	$\frac{10-0}{10+0+1} = 0$	$\frac{0}{0+1} = 0$	$\frac{0-0}{0+0+1} = 0$	$\frac{11-0}{11+0+1} = 0.92$
d	$\frac{0-13}{0+13+1} = -0.93$	$\frac{0-0}{0+0+1} = 0$	$\frac{0-0}{0+0+1} = 0$	$\frac{4}{4+1} = 0.80$	$\frac{13-0}{13+0+1} = 0.93$
e	$\frac{0-5}{0+5+1} = -0.83$	$\frac{0-11}{0+11+1} = -0.92$	$\frac{0-11}{0+11+1} = -0.92$	$\frac{0-13}{0+13+1} = -0.93$	$\frac{0}{0+1} = 0$

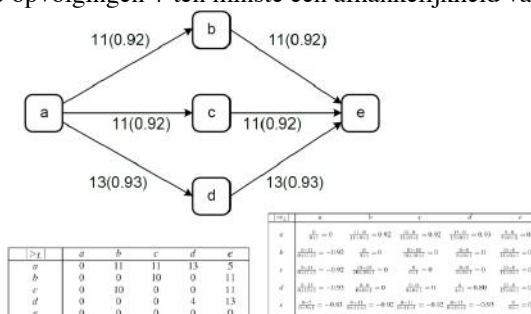
Lagere drempel

- Ten minste 2 rechtstreekse opvolgingen + ten minste een afhankelijkheid van 0,7



Hogere drempel

- Ten minste 5 rechtstreekse opvolgingen + ten minste een afhankelijkheid van 0,9



Waarom kunnen we nu omgaan met lawaai?

$$L = [\langle ABC \rangle^{100}, \langle BAC \rangle^1]$$

$$D_{A,B} = \frac{|A > B| - |B > A|}{|A > B| + |B > A| + 1} = \frac{100 - 1}{100 + 1 + 1} = 0.97$$

$$D_{B,A} = \frac{|B > A| - |A > B|}{|B > A| + |A > B| + 1} = \frac{1 - 100}{100 + 1 + 1} = -0.97$$

- Met een drempelwaarde van 0,9 maakt de relatie tussen B en A geen deel uit van het gedolven model.
- Het α -algoritme zou de relatie van B naar A opnemen!

Heuristische Miner: conclusie

- Voordelen
 - o Robuust
 - o Ruis
 - o Logische onvolledigheid
- Computationeel handelbaar
 - o Zeer snel

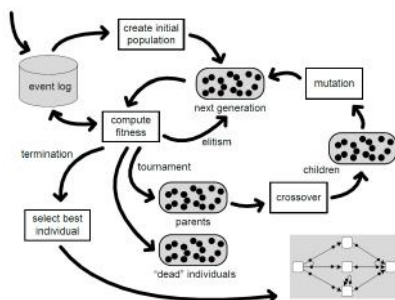
- Kan omgaan met veel representatieve bias problemen (onzichtbare taken, niet lokale afhankelijkheden, ...)
- Nadelen
 - o Geen dubbele taken: bijvoorbeeld je zet een handtekening in het begin maar ook op het einde
 - o Parameterinstellingen

Fodina - Verbeterde heuristische process mining

- Mijnbouwtechniek ontwikkeld in onze onderzoeksgroep (Dr. Seppe vanden Broucke)
 - o vanden Broucke, Seppe KLM, en Jochen De Weerd. "Fodina: een robuuste en flexibele heuristische procesontdekkingstechniek." Decision Support Systems (2017).
- Verschillende cruciale verbeteringen:
 - o Ontdekking van dubbele taken
 - o Robuuste heuristiek voor niet-verbonden taken
 - o Robuuste ontginning van enkelvoudige sporen
 - o Parallele patroongebaseerde berekening van splitsingen en verbindingen
 - o Vele andere, zie proefschrift

5. Other process discovery approaches

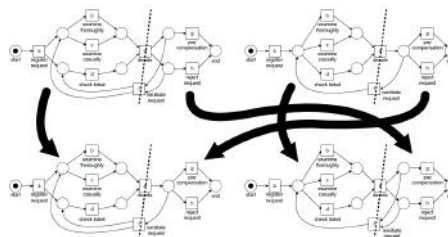
1) Genetische procesmining



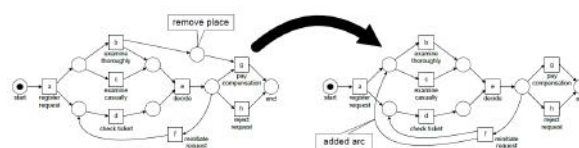
Ontwerpbesluiten

- Representatie van individuen
- Initialisatie
- Geschiktheidsfunctie
- Selectiestrategie (toernooi of elitisme)
- Crossover
- Mutatie

Voorbeeld: Crossover



Voorbeeld mutatie



- ⇒ Zo creëer je een next generation
- ⇒ Na 10000 heb je betere populaties op term van fitness

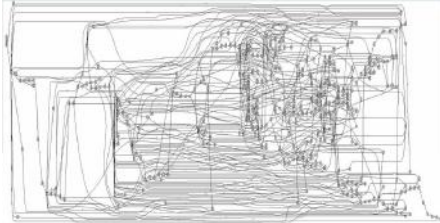
Kenmerken van genetic process mining

- Vereist veel rekenkracht
 - o Kan gedistribueerd worden
- Kan omgaan met ruis, onregelmatig gedrag, dubbele taken, onzichtbare taken
 - o Zeer expressieve ontdekkingstechniek

2) Proceskaarten – fuzzy miner

Motivatie: Mijn minder gestructureerde processen

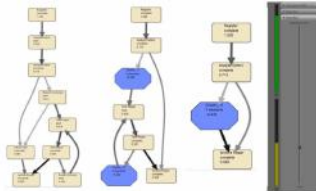
(In commercial tools)



Kaart metafoor: inzoomen/uitzoomen



Vereenvoudiging door abstractie + filtering



Fuzzy Miner: conclusie

- Voordelen
 - o Kan begrijpelijke procesmodellen afleiden uit zeer complexe eventlogs
 - o Gebruikt geavanceerde vereenvoudigingstechnieken om dit doel te bereiken
 - Filteren
 - Abstractie
- Nadelen
 - o Gebruikt eigen procesmodelleringstaal (Fuzzy model)
 - o Geen conversie mogelijk naar Petri-netten
 - Moeilijk om kwaliteitsmetriek te berekenen
 - Moeilijk te vergelijken met andere procesontdekkingstechnieken

→ Typische aanpak voor commercieel gereedschap, bv. Disco

3) Inductieve Miner

- Doel is het ontdekken van een **degelijk geluid en blok-gestructureerd** procesmodel
- Recursief algoritme vindt volgende beste opsplitsing om gebeurtenissenlogboek te verdelen
- Maakt het mogelijk afwegingen te maken tussen evaluatiedimensies (slider-gebaseerde benadering)
- "Procesboom" als onderliggend model

→ Beschikbaar in ProM: Inductive Visual Miner

4) Gesplitste Miner

- BPMN model discovery tool geïmplementeerd in het Apromore open source platform.
 - o Zie: <http://apromore.org/documentation/features/bpmn-miner>
 - o BPMN Miner 2.0
- Functionaliteiten
 - o Hiërarchische procesontdekkingstool (ontdekking van subprocessen)
 - o Standaard weergave in BPMN

6. Event data

Voorbereiding gebeurtenislogboek

- Data sourcing en preprocessing is een belangrijk aspect in elk process mining project
- Neemt gewoonlijk minstens 80% van de totale inspanning in beslag

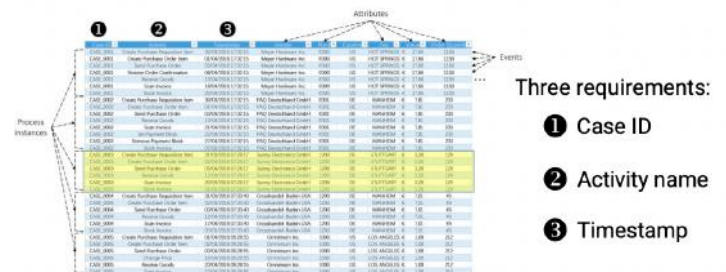
- Onderzoek naar datavoorbereiding bij process mining
 - o Beperkt
 - o Gefragmenteerd
 - Belangrijkste onderwerpen: gebeurtenisabstractie, gegevenskwaliteit

PM2: Een projectmethodologie voor process mining



=> Veel tijd op 2 stappen

Gebeurtenislogboeken: drie essentiële kenmerken

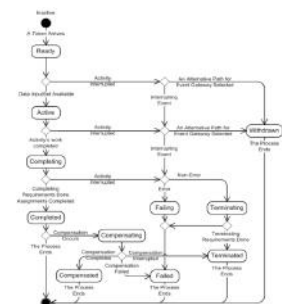


Extra gegevensattributen

- Zaakattributen: niet wijzigen
 - o Bijv. fabriek, land, stad
- Gebeurtenisattributen: zijn specifiek voor een stap/activiteit in het proces
 - o Bijv. de resource die de factuur boekt, het type van een vorderingsitem
- Gebruik:
 - o Filteren
 - o Contextuele informatie (bijv. tekstveld)
 - o Specifieke mijnbouwtechnieken (middelen, kosten, voorspellende procesbewaking)

Typen gebeurtenissen

- Een event kan transactionele informatie van activiteitinstanties weergeven
 - o Bv. BPMN 2.0 activity lifecycle model.
 - Twee standaardwaarden voor process mining
 - o Atomische activiteiten → "voltooid" gebeurtenissen
 - o Activiteiten met duur → "gestart" + "voltooid" - gebeurtenissen
- Al de rest wordt niet in beschouwing genomen



Event log bronnen

- BPMS (als je geluk hebt)
- Casemanagement- of ticketingsystemen
- ERP/CRM (SAP, Odoo, ...)
- Operationele databases / datawarehouses / data lakes
- Project management software (Trello, JIRA, ...)
- Webgegevens (clickstreamgegevens, vaak JSON-geformatteerd)
- Internet der dingen (IoT)
- ..

Uitdagingen bij het extraheren van gebeurtenislogs

- **Correlatie:** Gebeurtenissen in een gebeurtenissenlogboek worden gegroepeerd per geval. Deze eenvoudige eis kan een hele uitdaging zijn, omdat hiervoor event correlatie nodig is, d.w.z. events moeten aan elkaar gerelateerd worden.
- **Tijdstempels:** Gebeurtenissen moeten worden geordend per geval. Typische problemen: alleen data, verschillende klokken, vertraagde logging.
 - o Voorbeeld verpleegster die pas nadat ze al haat patiënten heeft gezien de data invoert : vertraagd
- **Snapshots:** Gevallen kunnen een langere levensduur hebben dan de geregistreerde periode, bv. een geval werd gestart vóór het begin van het gebeurtenissenlogboek.
- **Scoping:** Hoe beslissen welke "tabellen" worden opgenomen?
- **Granulariteit:** de gebeurtenissen in het event log zijn op een ander niveau van granulariteit dan de activiteiten die relevant zijn voor eindgebruikers.

→ Voorbeeld: Process Mining voor leeranalyse in MOOC'S

JavaScript Object Notatie (JSON)

- Elke gebeurtenis is een JSON-regel
- Representeert real-time server-naar-browser communicatie
- Ideaal voor machineverwerking, allesbehalve handig voor menselijke gebruikers
 - o Vrij omslachtige voorbewerking (vooral door geneste structuur)
 - o Veel overbodige/onbruikbare gegevens voor process mining

```
[{"username": "Student001", "event_type": "/courses/course-v1:eol+ENMAN303-01+2018-SV/xblock/block-v1:eol+ENMAN303-01+2018-SV+type@sequential+block@f69f91d8c1354a21a5428909d64c32fc/handler/xmodule_handler/get_completion", "ip": "anonymised_ip", "agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36", "host": "eol.uchile.cl", "referer": "https://eol.uchile.cl/courses/course-v1:eol+ENMAN303-01+2018-SV/courseware/d7597f7710a041d9920f93473ef9fd81/f69f91d8c1354a21a5428909d64c32fc/?child=first", "accept_language": "es-ES; es;q=0.9", "event": "{ \"POST\": { \"usage_key\": [\"block-v1:eol+ENMAN303-01+2018-SV+type@vertical+block@08d95c6a45bf4f96aeba164595a71afaf\"], \"GET\": {} }, \"event_source\": \"server\", \"context\": { \"course_user_tags\": {}, \"user_id\": 331, \"org_id\": \"eol\", \"course_id\": \"course-v1:eol+ENMAN303-01+2018-SV\", \"path\": \"/courses/course-v1:eol+ENMAN303-01+2018-SV/xblock/block-v1:eol+ENMAN303-01+2018-SV+type@sequential+block@f69f91d8c1354a21a5428909d64c32fc/handler/xmodule_handler/get_completion\", \"time\": \"2019-01-15T13:17:02.401630+00:00\", \"page\": null }
```

Target tabular data

Timestamp	Username	Event type	Referrer
2020-11-20 13:25:36.844	Student001	ENROL	weblink001
2020-11-21 01:31:58.687	Student002	PLAY VIDEO	weblink002
2020-11-22 20:23:23.042	Student002	SUBMIT TASK	weblink003
2020-11-22 20:28:34.492	Student002	PAUSE VIDEO	weblink004
2020-11-22 20:29:00.230	Student003	NEXT PAGE	weblink005
2020-11-22 20:30:48.177	Student003	VIEW FORUM	weblink006



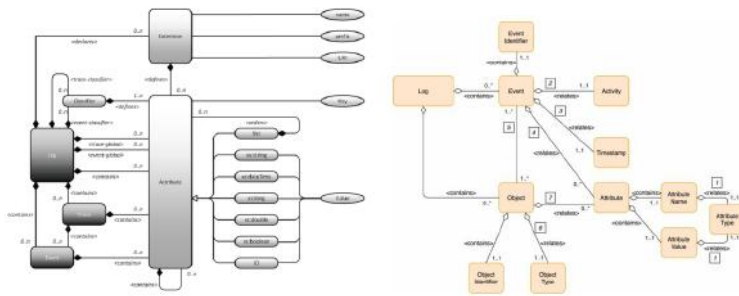
Timestamp	Username	Event type	Chapter	Section
2020-11-20 13:25:36	Student001	ENROL		
2020-11-21 01:31:58	Student002	PLAY VIDEO	C5	S3
2020-11-22 20:23:23	Student002	SUBMIT TASK	C1	S5
2020-11-22 20:28:34	Student002	PAUSE VIDEO	C2	S3
2020-11-22 20:29:00	Student003	NEXT PAGE	C3	S1
2020-11-22 20:30:48	Student003	VIEW FORUM	C2	S3

Navigeren door event log granulariteitsniveaus

- Event granulariteitsniveaus
- Gegevens kunnen op verschillende abstractieniveaus worden geanalyseerd met behulp van process mining/sequence mining-technieken.
 - o Zie bijv. Deeva, G., De Weerd, J. (sup.), Snoeck, M. (cosup.) (2021). Behavioral data analytics in intelligent tutoring systems. Proefschrift.
 - o Team van Jorge Munoz-Gama: <https://haplab.org/research/>

Timestamp	Username	EG1	EG2	EG3
2020-11-20 13:25:36	Student001	ENROL	ENROL	ENROL
2020-11-21 01:31:58	Student002	PLAY VIDEO	PLAY VIDEO/C5	PLAY VIDEO/C5S3
2020-11-22 20:23:23	Student002	SUBMIT TASK	SUBMIT TASK/C1	SUBMIT TASK/C1S5
2020-11-22 20:28:34	Student002	PAUSE VIDEO	PAUSE VIDEO/C2	PAUSE VIDEO/C2S3
2020-11-22 20:29:00	Student003	NEXT PAGE	NEXT PAGE/C3	NEXT PAGE/C3S1
2020-11-22 20:30:48	Student003	VIEW FORUM	VIEW FORUM/C2	VIEW FORUM/C2S3

Opslag van gebeurtenissenlogboek



⇒ Gebruik van standaarden om error codes te vermijden

XES (eXtensible Event Stream)

- Zie www.xes-standard.org
- Vastgesteld door de IEEE Task Force on Process Mining
- Voorganger: MXML en SA-MXML
- De indeling wordt ondersteund door tools als ProM (vanaf versie 6), Disco, XESame en OpenXES.
- ProMimport ondersteunt MXML

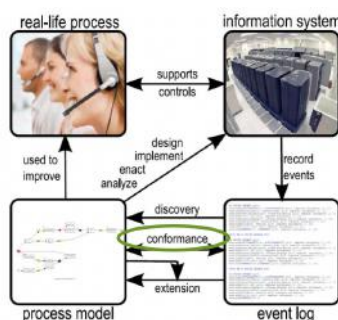
7. Conclusion

- Procesontdekking is de belangrijkste innovatie op het gebied van process mining.
- Er bestaan verschillende algoritmen om automatisch procesmodellen af te leiden uit event logs, gaande van het oorspronkelijke algoritme alpha tot de meest recente technieken Fodina, Inductive Miner en Split Miner.
- Toch zijn er nog ruime mogelijkheden om de algoritmische technieken te verbeteren, gezien de complexiteit van het probleem.
- Gebeurtenisgegevens gaan gepaard met belangrijke uitdagingen op het gebied van kwaliteit, sourcing, enz. voordat procesontdekking met succes kan worden toegepast.

BPM12. Process Mining – Conformance Checking and Extension

1. Introduction to Conformance Checking

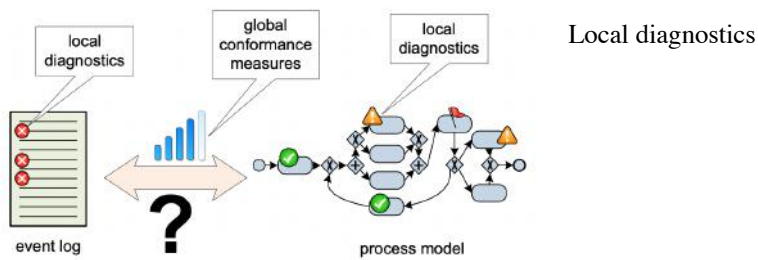
Conformiteitscontrole --> 2de grote techniek binnen process mining



Gedrag in data vergelijken (structureel en algoritmisch) met een normatieve (hoe. Het process model.

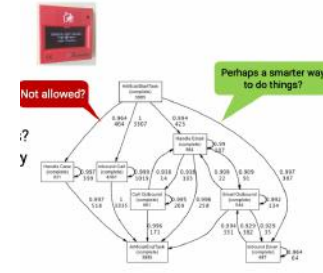
Afwijkingen vergelijken !

Conformiteitscontrole gebruiken



Afwijkingen?

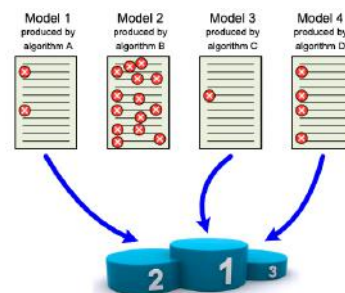
- Is het model of het logboek "fout"?
- "Gewenste" of "ongewenste" afwijkingen?
 - o Niet noodzakkelijk ongewenst
- "Het glas breken" kan levens redden!



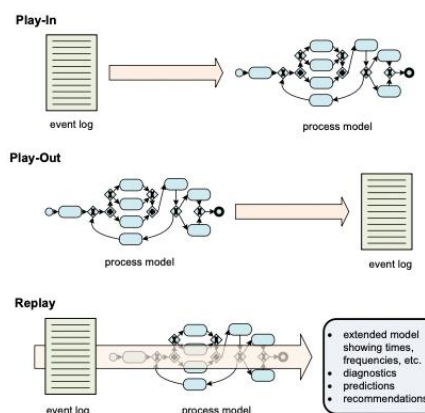
Process Mining voor audits

- Audits worden uitgevoerd om de geldigheid en betrouwbaarheid van informatie over deze organisaties en bijbehorende processen vast te stellen.
 - o Ze nemen samples
- Dit wordt gedaan om te controleren of bedrijfsprocessen worden uitgevoerd binnen bepaalde grenzen die zijn vastgesteld door managers, overheden en andere belanghebbenden.
- Uiteraard kan process mining helpen om fraude, malversaties, risico's en inefficiënties op te sporen.
- Alle gebeurtenissen in een bedrijfsproces kunnen worden geëvalueerd en dit kan ook gebeuren terwijl het proces nog loopt.

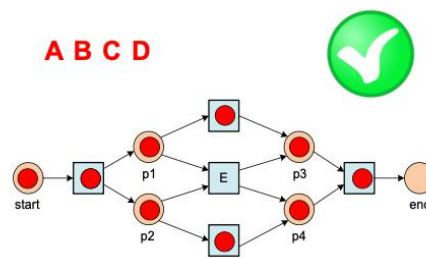
Een andere belangrijke use case: Evaluatie van process mining algoritmen



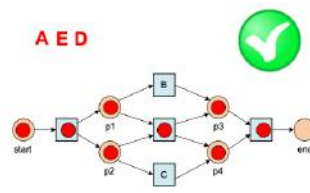
Replay: Gebeurtenissen verbinden met modelementen is essentieel voor process mining



Replay:

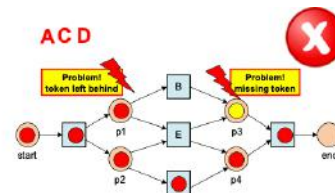


Replay:



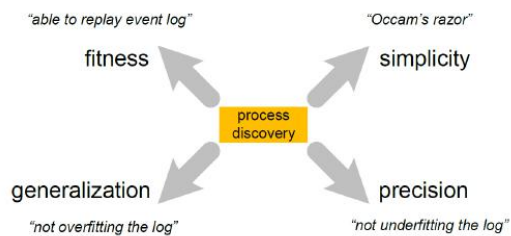
Replay kan problemen opsporen:

=> probleem met conformance

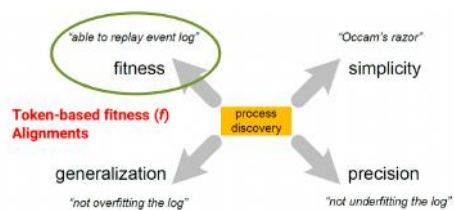


2. Measuring fitness

Conformiteit: vier dimensies



→ Geschiktheid



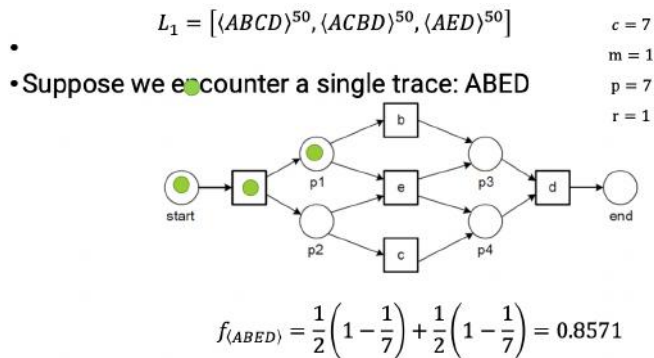
Op tokens gebaseerde fitness (f)

- Herhaling van de sporen in het ontdekte model
- Fitnessprobleem doet zich voor wanneer
 - o een of meer tokens ontbreken tijdens het overspelen (mi)
 - o een of meer tokens overblijven na het overspelen (ri)

$$f_{trace} = \frac{1}{2} \left(1 - \frac{m}{c} \right) + \frac{1}{2} \left(1 - \frac{r}{p} \right)$$

m = number of missing tokens
 c = number of consumed tokens
 r = number of remaining tokens
 p = number of produced tokens

Voorbeeld:



→ Wanneer je e moet hebben mis je een token want die werd al gebruikt voor b

→ Er zijn er 3 die bij d zijn maar er moeten er maar 2 gebruikt worden dus 1 remaining !

Geschiktheid van een volledig logboek L

- Aggregaat over alle sporen

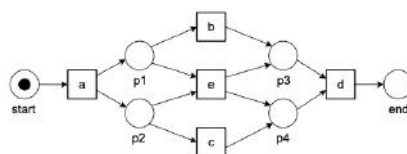
$$f_L = \frac{1}{2} \left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{\sigma}} \right) + \frac{1}{2} \left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{\sigma}} \right)$$

$L(\sigma)$ = frequency of trace σ
 m_{σ} = number of missing tokens for trace σ
 c_{σ} = number of consumed tokens for trace σ
 r_{σ} = number of remaining tokens for trace σ
 p_{σ} = number of produced tokens for trace σ

Oefening 1: fitness berekenen met ontbrekende en resterende tokens

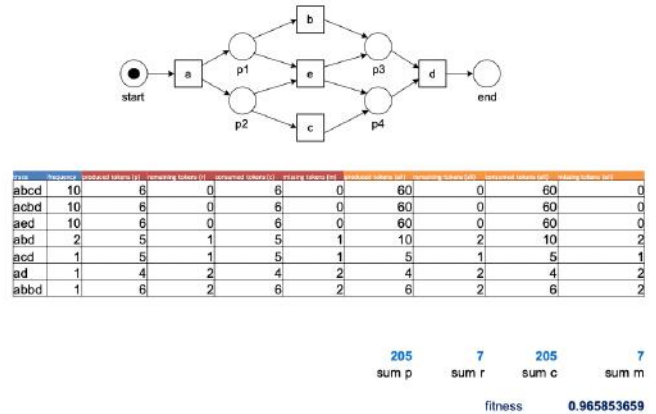
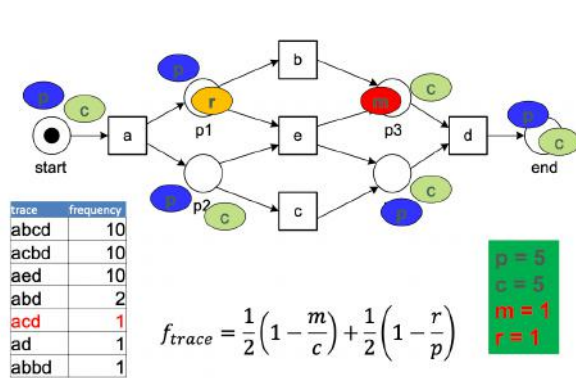
- Beschouw het gebeurtenissenlogboek met 35 gevallen.
- Wat is de fitness?

trace	frequency
abcd	10
acbd	10
aed	10
abd	2
acd	1
ad	1
abbd	1



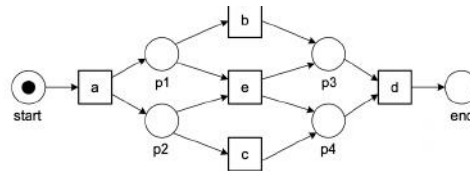
Laten we één spoor kiezen: acd

Fitness = 0.9658

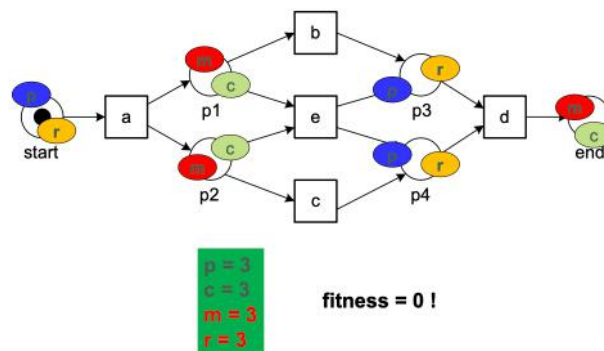


Oefening 2: fitness berekenen met ontbrekende en resterende tokens

- Beschouw een gebeurtenissenlogboek dat slechts één geval bevat dat bestaat uit één gebeurtenis, $L = \{e\}$.
- Wat is de fitness?



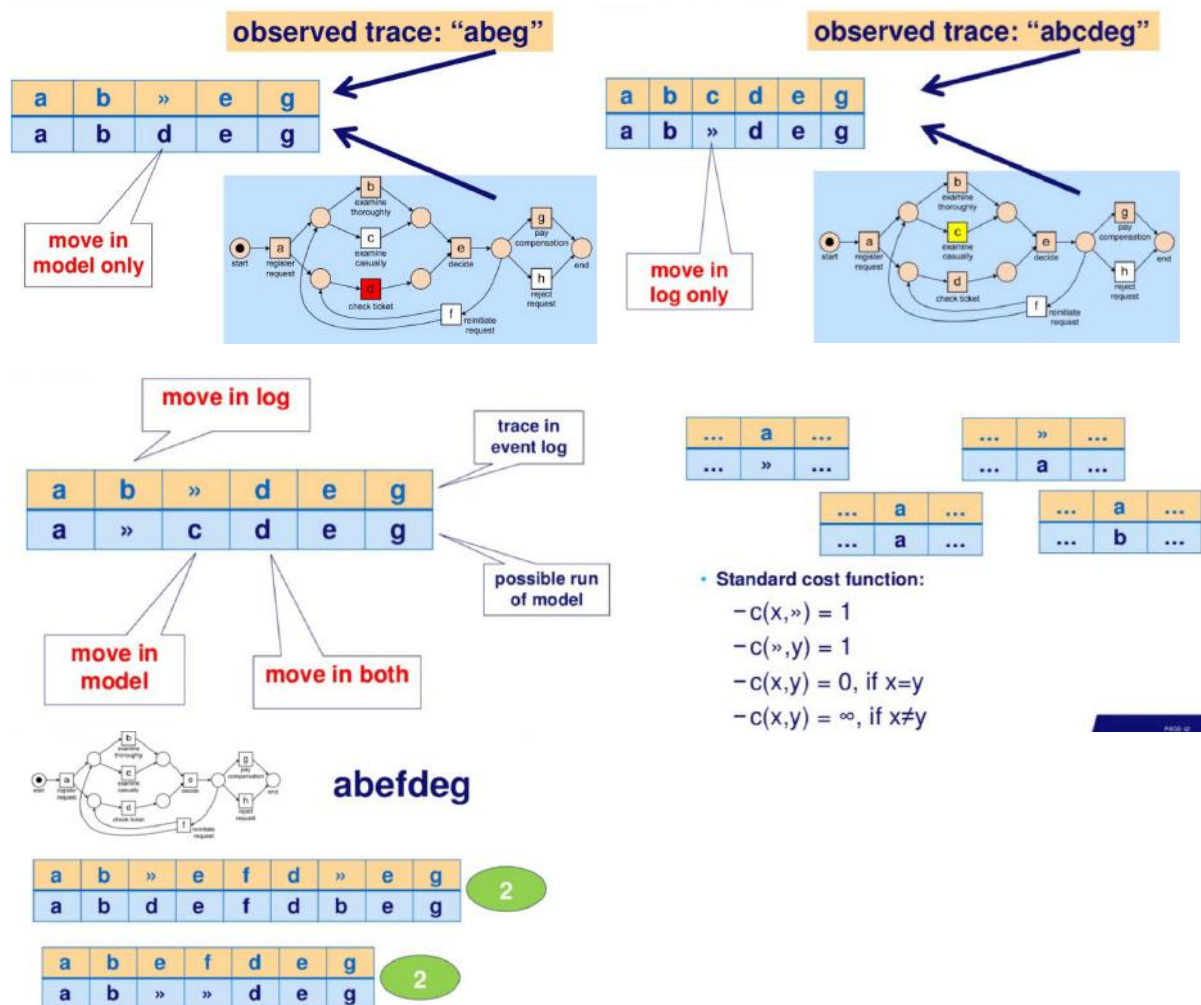
Replay $\{e\}$



Conformiteitscontrole op basis van uitlijning

- Op alignment gebaseerde technieken proberen de beste opeenvolging van overgangen door het model te vinden die zo goed mogelijk overeenkomt met een spoor.
 - o Laten model-only en log-only bewegingen toe
 - o Maar: afwijkingen hebben een kostprijs
 - o De beste oplossing is de oplossing met de laagste kosten
- Optimalisatie kan worden uitgevoerd met het A*-algoritme.

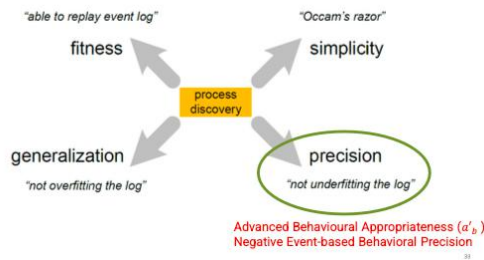
Op uitlijning gebaseerde conformiteitscontrole



Op uitlijning gebaseerde conformiteitscontrole

- Voordelen:
 - o Vermijdt het forceren van overgangen.
 - o introduceert geen tokens die de replay kunnen verstoren.
 - o Mooi optimalisatieprobleem: boomgebaseerd algoritme om een minimaal kostenpad te vinden.
 - o Kosten kunnen worden aangepast door gebruiker: bijv. hogere kosten wanneer activiteit wordt uitgevoerd door verkeerde gebruiker
- Nadelen:
 - o Zeer tijdrovend, slecht schaalbaar
 - o Model-only verhuizing moeilijk te rechtvaardigen
 - o Kosten instellen is leuk, maar wat als optimalisatie om je ingestelde kosten heen gaat? Aangepaste kosten zijn geen vervanging voor regelcontrole

3. Measuring precision

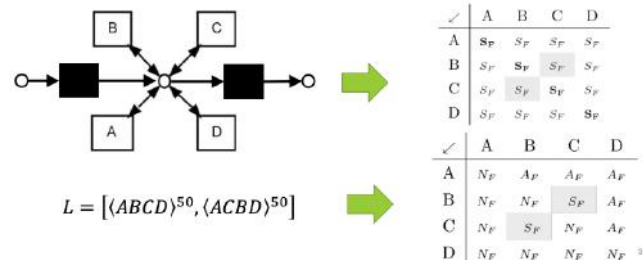


Geavanceerde gedragsaanpassing

- Een van de eerste precisiemetingen
- Idee: vergelijk het gedrag in het model met het gedrag in het gebeurtenissenlogboek (met zogenaamde footprints)
 - o "Als het model veel meer gedrag toelaat dan het logboek vertoont, straf dan!"
- De metriek vergelijkt twee soorten relaties tussen activiteiten
 - o Volgt relatie/ follows relationship:
 - Volgt altijd (AF) / Volgt nooit (NF) / Volgt soms (SF)
 - o Voorloopt relatie / precedes relationship:
 - Altijd voor (AP) / Nooit voor (NP) / Soms voor (SP)

Geavanceerde geschiktheid voor gedrag

- Voorbeeld: bloemenmodel
- **Volgt** relatiematrix volgens model en logboek



Gevorderde gedragsaanpassing

Log follows relations					Model follows relations				
✓	A	B	C	D	✓	A	B	C	D
A	N _F	A _F	A _F	A _F	A	S _F	S _F	S _F	S _F
B	N _F	N _F	S _F	A _F	B	S _F	S _F	S _F	S _F
C	N _F	S _F	N _F	A _F	C	S _F	S _F	S _F	S _F
D	N _F	N _F	N _F	N _F	D	S _F	S _F	S _F	S _F

- De relatie **Soms Volgt** is minder restrictief dan **Altijd Volgt** of **Nooit Volgt**
- Als we in het model veel meer "soms volgt" en "soms voorafgaat" relaties waarnemen dan in het logboek, laat het model te veel gedrag toe!

$$a'_B = \frac{|S_F^l \cap S_F^m|}{2 \times |S_F^m|} + \frac{|S_P^l \cap S_P^m|}{2 \times |S_P^m|} \quad a'_B = \frac{2}{2 \times 16} + \frac{2}{2 \times 16} = 0.125$$

Een waarde van 1 betekent perfecte precisie, lagere scores geven aan dat het model te veel gedrag toelaat (d.w.z. underfitting).

Op kunstmatige negatieve gebeurtenissen gebaseerde conformiteitscontrole

- In onze groep hebben wij conformiteitscontroletechnieken ontwikkeld op basis van kunstmatige negatieve gebeurtenissen. (in de andere model bekeken we enkel wat mogelijk was en niet wat NIET kan)
- Met de beschikbaarheid van negatieve gebeurtenissen (= activiteiten waarvan werd ontkend dat ze plaatsvonden) kunnen traditionele evaluatiemaatstaven voor classificatie worden gebruikt (recall, precision, F-measure).
- De beschikbaarheid van negatieve gebeurtenissen in reële logbestanden is echter onwaarschijnlijk.
- Daarom: inductie van kunstmatige negatieve gebeurtenissen in een gebeurtenissenlogboek.

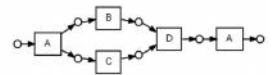
Kunstmatige negatieve gebeurtenissen

- Principe:
 - o Bij elke positieve gebeurtenis wordt nagegaan of een andere gebeurtenis die in het logboek voorkomt ook kan voorkomen, gezien het voorvoegsel ervan. Daarom wordt getest of er in het logboek een soortgelijke voorvoegselreeks bestaat met deze andere gebeurtenis op dezelfde positie als de positieve gebeurtenis. Indien dit niet het geval is, kan een negatief voorval in het logboek worden geïnjecteerd op de positie van het positieve voorval.

Artificial negative events

	Trace 1					Trace 2				
Positive events	A	B	C	D	A	A	C	B	D	A
Artificially generated negative events	B ⁿ	A ⁿ	A ⁿ	A ⁿ	B ⁿ	B ⁿ	A ⁿ	A ⁿ	A ⁿ	B ⁿ
	C ⁿ	D ⁿ	B ⁿ	C ⁿ	C ⁿ	D ⁿ	C ⁿ	B ⁿ	C ⁿ	C ⁿ
	D ⁿ	D ⁿ	C ⁿ	D ⁿ	D ⁿ	D ⁿ	C ⁿ	D ⁿ	C ⁿ	D ⁿ

No. of Instances	Log traces
50	ABCD A
50	ACBD A



Recall (fitness) en precision

- Confusion matrix na log replay
 - o Deze statistieken zijn niet gebaseerd op token maar op events

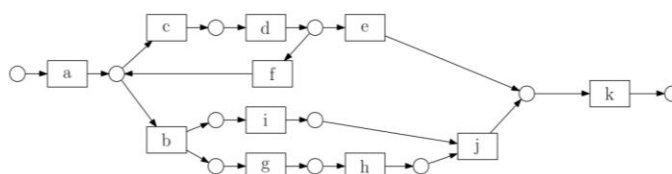
• Recall:
$$r_B^p = \frac{\sum_{i=1}^k n_i TP_i}{\sum_{i=1}^k n_i TP_i + \sum_{i=1}^k n_i FN_i}$$

• Precision:
$$p_B = \frac{\sum_{i=1}^k n_i TP_i}{\sum_{i=1}^k n_i TP_i + \sum_{i=1}^k n_i FP_i}$$

	True pos.	True neg.
Pred. pos.	True Pos. (TP)	False Pos. (FP)
Pred. neg.	False Neg. (FN)	True Neg. (TN)

Voorbeeld: Kunstmatige negatieve gebeurtenissen

MULTIPLICITY OF σ	TRACE $\sigma \in L$
113	$\langle a, c, d, e, k \rangle$
110	$\langle a, b, i, g, h, j, k \rangle$
74	$\langle a, b, g, i, h, j, k \rangle$
63	$\langle a, b, g, h, i, j, k \rangle$
39	$\langle a, c, d, f, c, d, e, k \rangle$
30	$\langle a, c, d, f, b, i, g, h, j, k \rangle$
19	$\langle a, c, d, f, b, g, i, h, j, k \rangle$
16	$\langle a, c, d, f, b, g, h, i, j, k \rangle$
8	$\langle a, c, d, f, c, d, f, b, g, h, i, j, k \rangle$
8	$\langle a, c, d, f, c, d, f, c, d, e, k \rangle$
8	$\langle a, c, d, f, c, d, f, b, i, g, h, j, k \rangle$
5	$\langle a, c, d, f, c, d, f, b, g, i, h, j, k \rangle$
3	$\langle a, c, d, f, c, d, f, c, d, f, b, i, g, h, j, k \rangle$
2	$\langle a, c, d, f, c, d, f, c, d, f, c, d, e, k \rangle$
2	$\langle a, c, d, f, c, d, f, c, d, f, b, g, h, i, j, k \rangle$



Algorithm 1.1 Basic artificial negative event generation in an event log.

Input: An event log L

Output: A set N of induced artificial negative events

```

1:  $N := \emptyset$ 
2: for all  $\sigma \in L$  do
3:   for all  $\sigma_i \in \sigma$  do
4:     for all  $a \in A_L \setminus \{\sigma_i\}$  do
5:       if  $\nexists \tau \in L : [\sigma \neq \tau \wedge a = \tau_i \wedge \forall l \in \{1, \dots, i-1\} : [\sigma_l = \tau_l]]$  then
6:         % Record negative event at position  $i$  in trace  $\sigma$  for activity  $a$ :
7:          $N := N \cup \{(\sigma, i, a)\}$ 
8:       end if
9:     end for
10:   end for
11: end for
12: return  $N$ 

```

Step 1 :

- [a, c, d, e, k]
- Eerste positie
 - o Voorvoegsel []
 - o Geen ander spoor met b, c, d, e, f, g, h, i, j, k na het lege voorvoegsel
 - o Alle gebeurtenissen worden als negatieve gebeurtenissen toegevoegd aan positie 1
- Resultaat: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, c, d, e, k]

MULTIPLICITY OF σ	TRACE $\sigma \in L$
113	$\langle a, c, d, e, k \rangle$
110	$\langle a, b, i, g, h, j, k \rangle$
74	$\langle a, b, g, i, h, j, k \rangle$
63	$\langle a, b, g, h, i, j, k \rangle$
39	$\langle a, c, d, f, c, d, e, k \rangle$
30	$\langle a, c, d, f, b, i, g, h, j, k \rangle$
19	$\langle a, c, d, f, b, g, i, h, j, k \rangle$
16	$\langle a, c, d, f, b, g, h, i, j, k \rangle$
8	$\langle a, c, d, f, c, d, f, b, g, h, i, j, k \rangle$
8	$\langle a, c, d, f, c, d, f, c, d, e, k \rangle$
8	$\langle a, c, d, f, c, d, f, b, i, g, h, j, k \rangle$
5	$\langle a, c, d, f, c, d, f, b, g, i, h, j, k \rangle$
3	$\langle a, c, d, f, c, d, f, c, d, f, b, i, g, h, j, k \rangle$
2	$\langle a, c, d, f, c, d, f, c, d, f, c, d, e, k \rangle$
2	$\langle a, c, d, f, c, d, f, c, d, f, b, g, h, i, j, k \rangle$

Step 2:

- [a, c, d, e, k]
- Tweede positie
 - o Voorvoegsel [a]
 - o Geen ander spoor met a, d, e, f, g, h, i, j, k na dit voorvoegsel.
 - o Deze gebeurtenissen worden als negatieve gebeurtenissen toegevoegd aan positie 2
 - o Niet b, want die wordt waargenomen na voorvoegsel [a]
- Resultaat: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, d, e, k]

Step 3

- [a, c, d, e, k]
- Derde positie
 - o Voorvoegsel [a,c]
 - o Geen ander spoor met a, b, c, e, f, g, h, i, j, k na dit voorvoegsel.
 - o Deze gebeurtenissen worden als negatieve gebeurtenissen toegevoegd aan positie 3
- Resultaat: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, (a-, b-, c-, e-, f-, g-, h-, i-, j-, k-), d, e, k].

Step 4

- [a, c, d, e, k]
- Vierde positie
 - o Voorvoegsel [a, c, d]
 - o Geen ander spoor met a, b, c, d, f, g, h, i, j, k na dit voorvoegsel.
 - o Deze gebeurtenissen worden als negatieve gebeurtenissen toegevoegd aan positie 4
 - o Niet f, want we nemen [a, c, d, f] waar.
- Resultaat: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, (a-, b-, c-, e-, f-, g-, h-, i-, j-, k-), d, (a-, b-, c-, d-, g-, h-, i-, j-, k-), e, k].

Stap 5

- [a, c, d, e, k]
- Vijfde positie
 - o Voorvoegsel [a, c, d, e]
 - o Geen ander spoor met a, b, c, d, e, f, g, h, i, j na dit voorvoegsel.
 - o Deze gebeurtenissen worden als negatieve gebeurtenissen toegevoegd aan positie 5
 - o Resultaat: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, (a-, b-, c-, e-, f-, g-, h-, i-, j-, k-), d, (a-, b-, c-, d-, g-, h-, i-, j-, k-), e, (a-, b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), k].

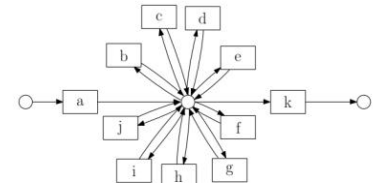
Berekening van recall en precisie

- [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), **a**, (a-, d-, e-, f-, g-, h-, i-, j-, k-), **c**, (a-, b-, c-, e-, f-, g-, h-, i-, j-, k-), **d**, (a-, b-, c-, d-, g-, h-, i-, j-, k-), **e**, (a-, b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), **k**].

	True pos.	True neg.
Pred. pos.	5	34
Pred. neg.	0	14

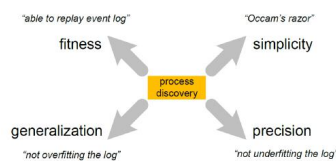
$$r_B^p = \frac{TP}{TP+FN} = \frac{5}{5+0} = 100\%$$

$$p_B = \frac{TP}{TP+FP} = \frac{5}{5+34} = 12,8\%$$



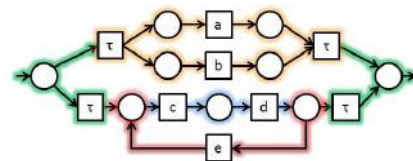
4. Measuring generalization/simplicity

Generalisatie? Eenvoud?



Eenvoud. (=> door het tellen van elementen)

- Gebaseerd op topologie, lay-out, "visueel" van het model.
- Bijv. tellen van aantal knooppunten, plaatsen, overgangen, bogen, AND vs XOR-splitsingen...
- Gemiddelde graad van knooppuntverbindingen
- SESE-knooppunten (single-entry-single-exit)
- Structuur van blokken



Generalisatie

- Er bestaan verschillende pogingen
 - o Met behulp van kruisvalidatie
 - o Gebruik van kunstmatige gegeneraliseerde gebeurtenissen
- Onvoltooid onderzoeksprobleem

5. Conclusion

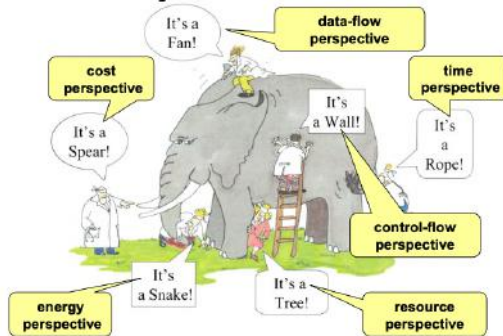
- Conformiteitscontrole gaat over het vergelijken van modelgedrag met waargenomen gedrag in het gebeurtenissenlogboek.
- Algoritmen voor conformiteitscontrole (replay, alignments) kunnen een dergelijke vergelijking automatisch uitvoeren.
- Het vergelijken van gemodelleerd en geobserveerd gedrag kan waardevolle inzichten opleveren, bijvoorbeeld voor auditors.
- Er bestaan verschillende metrieken om conformiteitsdimensies te kwantificeren, vooral fitness- en precisiewaarden.

BPM13. Process Mining – Extension and Tooling

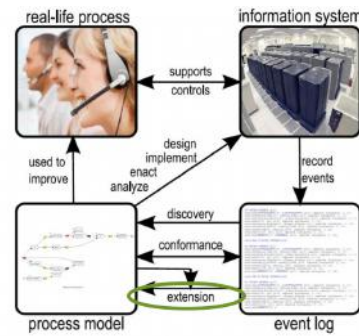
1. Extension Techniques

⇒ Uitbreiding : het systeem verrijken met informatie...

Processen gaan niet alleen over control-flow!



→ Afhankelijk van hoe je naar de data kijkt kan je verschillende perspectieven hebben maar dat zijn secundaire doelen. De hoofddoel is process discovery.



Attributen in gebeurtenislogboeken

- Een proces bestaat uit **zaken (zaak-id)**
- Een zaak bestaat uit **gebeurtenissen** zodat elke gebeurtenis betrekking heeft op precies één zaak
- Gebeurtenissen binnen een zaak zijn **geordend** (tijdstempel)
- **Gebeurtenissen kunnen attributen hebben !!**

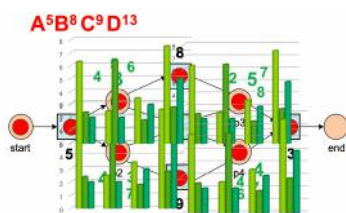
case id	event id	properties				
		timestamp	activity	resource	cost	...
1	35054423	30-12-2009 11:02	register request	Pete	50	...
	35054424	31-12-2009 10:06	examine thoroughly	Sara	400	...
	35054425	05-01-2010 11:12	check ticket	Mike	100	...
	35054426	06-01-2010 11:18	decide	Sara	200	...
	35054427	07-01-2010 11:24	reject request	Pete	200	...
2	35054483	30-12-2009 11:32	register request	Mike	50	...
	35054485	30-12-2009 12:12	check ticket	Mike	100	...
	35054487	30-12-2009 14:06	examine casually	Pete	400	...
	35054488	05-01-2010 11:22	decide	Sara	200	...
	35054489	06-01-2010 11:26	pay compensation	Ellen	300	...
3	35054521	30-12-2009 14:12	register request	Pete	50	...
	35054522	30-12-2009 15:06	examine casually	Mike	400	...
	35054524	30-12-2009 16:34	check ticket	Ellen	100	...
	35054525	06-01-2010 09:18	decide	Sara	200	...
	35054526	06-01-2010 12:18	reunite request	Sara	200	...
	35054527	06-01-2010 13:06	examine thoroughly	Scan	400	...
	35054530	08-01-2010 11:43	check ticket	Pete	100	...
	35054531	09-01-2010 09:55	decide	Sara	200	...
	35054532	15-01-2010 16:45	pay compensation	Ellen	200	...
4	35054641	06-01-2010 15:02	register request	Pete	50	...
	35054643	07-01-2010 12:06	check ticket	Mike	100	...
	35054644	08-01-2010 14:43	examine thoroughly	Scan	400	...
	35054645	09-01-2010 11:02	decide	Sara	200	...
	35054647	12-01-2010 13:44	reject request	Ellen	200	...

Uitbreidingstechnieken (Gewoon 4 van de vele technieken)

1) Prestatieanalyse

- Tijdsdimensie gebruiken
- Tijdens replay kunnen we timinginformatie bijhouden: wachttijd, doorlooptijd
- Kan gebruikt worden om procesmodellen te annoteren: identificeren van knelpunten

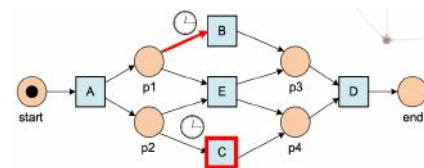
Replay kan tijdsinformatie onttrekken



Kleine letters tonen het tijd waarop het gebeurt !

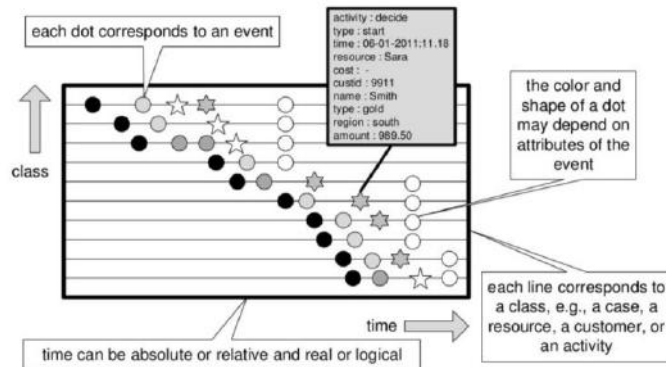
A duurt 5, B duurt 8

→ Als je dat voor alles doet krijg je statistieken en kan je deze in de model toevoegen door pijlen of activiteiten dikker te maken enzo.



2) Stippeldiagrammen

- "Helikopterview"

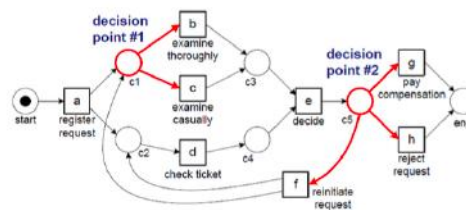


→ De venet voostellen in een 2-dimensionale diagram!
→ Het de venetlog op een visuele manier zonder een process model te tonen

→ Je krijgt inzicht over hoe een process wordt toegepast.

3) Decision mining

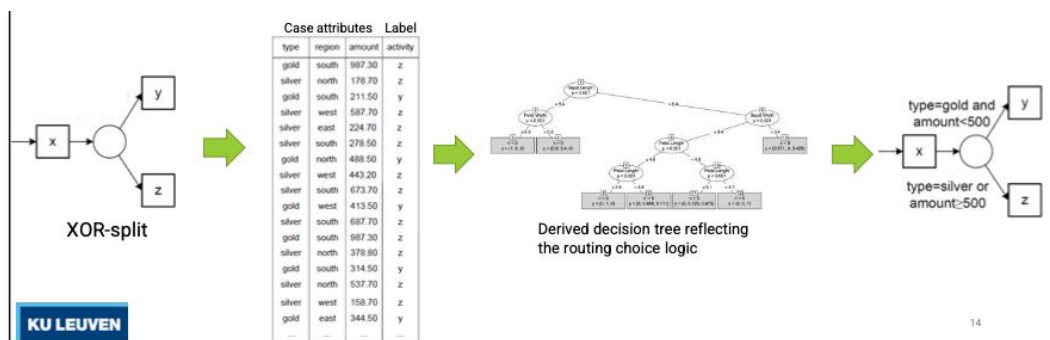
- Streeft ernaar regels te vinden die routekeuzes verklaren in termen van de kenmerken van een geval
- Beslissingspunten zijn van het type XOR-split



→ Decision mining wilkt de invloed van de beslissing tonen in dat punt: de logica daarachter "herontdekken"

Methodologie

- Het doel is de "kenmerken" (voorspellende variabelen) te vinden die de routeringsbeslissing beïnvloeden
- Een classificatietechniek zoals het leren van beslisbomen kan worden gebruikt om dergelijke regels te vinden.
- Uiteraard wordt alleen informatie uit het verleden gebruikt, d.w.z. het traject tot aan het beslissingspunt.



Decision mining: conclusie

- Toepassing van classificatietechnieken op procesanalyse
 - o Bijv. beslisbomen
- Doel: vinden van hoofdoorzaken die bepalend zijn voor routekeuzes in een procesmodel
- Waarde: kan bijkomende inzichten verschaffen over welke factoren de uitvoering van een bedrijfsproces beïnvloeden

4) Organisational mining

case id trace

1	$\langle a^{Pete}, b^{Sue}, d^{Mike}, e^{Sara}, h^{Pete} \rangle$
2	$\langle a^{Mike}, d^{Mike}, c^{Pete}, e^{Sara}, g^{Ellen} \rangle$
3	$\langle a^{Pete}, c^{Mike}, d^{Ellen}, e^{Sara}, f^{Sara}, b^{Sean}, d^{Pete}, e^{Sara}, g^{Ellen} \rangle$
4	$\langle a^{Pete}, d^{Mike}, b^{Sean}, e^{Sara}, h^{Ellen} \rangle$
5	$\langle a^{Ellen}, c^{Mike}, d^{Pete}, e^{Sara}, f^{Sara}, d^{Ellen}, c^{Mike}, e^{Sara}, f^{Sara}, b^{Sue}, d^{Pete}, e^{Sara}, h^{Mike} \rangle$
6	$\langle a^{Mike}, c^{Ellen}, d^{Mike}, e^{Sara}, g^{Mike} \rangle$
...	...

(a = register request, b = examine thoroughly, c = examine casually, d = check ticket, e = decide, f = reinstate request, g = pay compensation, and h = reject request)

- ⇒ Het analyseren van de resources information: het analyseren van de mensen/ systemen of de combinatie van beide die de taken verrichten.
- ⇒ Veel beslissingen voor manager zijn resource gericht: hoe een team organiseren, hoe werken teams tesamen, hoeveel mensen in een team...

Resource-activity matrix

mean number of times a resource performs an activity per case

	a	b	c	d	e	f	g	h
Pete	0.3	0	0.345	0.69	0	0	0.135	0.165
Mike	0.5	0	0.575	1.15	0	0	0.225	0.275
Ellen	0.2	0	0.23	0.46	0	0	0.09	0.11
Sue	0	0.46	0	0	0	0	0	0
Sean	0	0.69	0	0	0	0	0	0
Sara	0	0	0	0	2.3	1.3	0	0

Activiteit a wordt voor elk geval precies één keer uitgevoerd (vandaar dat de som van de eerste kolom 1 is). Pete, Mike en Ellen zijn de enigen die deze activiteit uitvoeren. In 30% van de gevallen wordt a uitgevoerd door Piet, 50% door Piet en 20% door Ellen. Activiteiten e en f worden altijd uitgevoerd door Sara. Activiteit e wordt gemiddeld 2,3 keer per geval uitgevoerd. Enz. (Maar houdt niet rekening met de belang van elke taak, als a bijvoorbeeld niet relevant zou zijn zou je die uit het tabel moeten wissen)

Profielen op basis van de middelen-activiteitenmatrix

	a	b	c	d	e	f	g	h
Pete	0.3	0	0.345	0.69	0	0	0.135	0.165
Mike	0.5	0	0.575	1.15	0	0	0.225	0.275
Ellen	0.2	0	0.23	0.46	0	0	0.09	0.11
Sue	0	0.46	0	0	0	0	0	0
Sean	0	0.69	0	0	0	0	0	0
Sara	0	0	0	0	2.3	1.3	0	0

→Mike is verschillend van Sue : Mike doet c,d, g en h terwijl Sue enkel b doet. Het kan zijn dat Sue een expert is in B en Mike is meer een generalist.

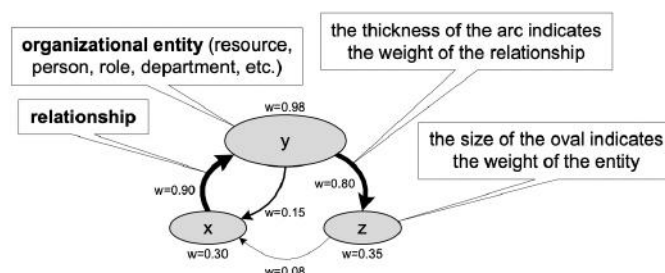
	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>	<i>h</i>
Pete	0.3	0	0.345	0.69	0	0	0.135	0.165
Mike	0.5	0	0.575	1.15	0	0	0.225	0.275
Ellen	0.2	0	0.23	0.46	0	0	0.09	0.11
Sue	0	0.46	0	0	0	0	0	0
Sean	0	0.69	0	0	0	0	0	0
Sara	0	0	0	0	2.3	1.3	0	0

→Sara is mogelijk een manager: e n f zijn beslissingactiviteiten.

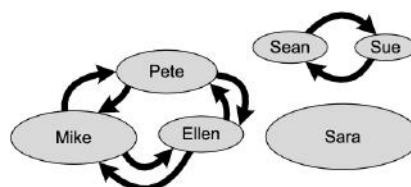
⇒ Profielen indetificeren

Sociaal netwerk

→data visualiseren aan de hand van een sociaal netwerk : hoe mensen samenwerken en hoe mensen gelinkt zijn.

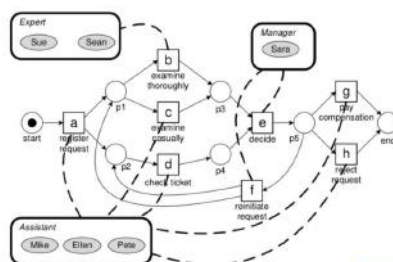


Sociaal netwerk gebaseerd op gelijkheid van profielen



Hulpbronnen die vergelijkbare verzamelingen activiteiten uitvoeren zijn aan elkaar gerelateerd. Sara is de enige bron die e en f uitvoert. Daarom is zij niet verbonden met andere bronnen. Self-loops worden onderdrukt omdat ze geen informatie bevatten (self-similarity).

Het samenvoegen van het proces en de maatschappelijke visie:

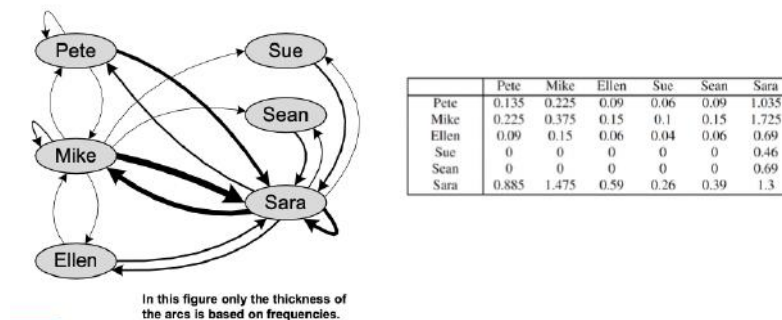


Overdracht van de werkmatrix :

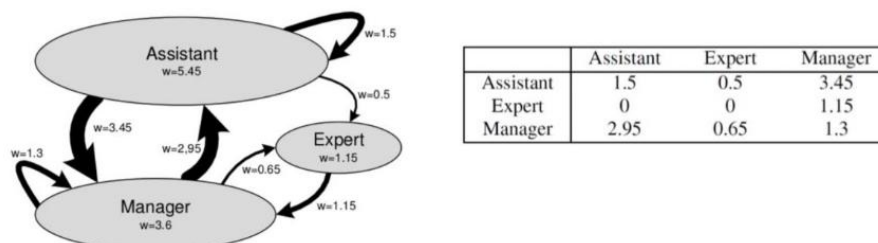
	Pete	Mike	Ellen	Sue	Sean	Sara
Pete	0.135	0.225	0.09	0.06	0.09	1.035
Mike	0.225	0.375	0.15	0.1	0.15	1.725
Ellen	0.09	0.15	0.06	0.04	0.06	0.69
Sue	0	0	0	0	0	0.46
Sean	0	0	0	0	0	0.69
Sara	0.885	1.475	0.59	0.26	0.39	1.3

- Tel het aantal keren dat werk wordt overgedragen van de ene bron naar de andere (gemiddeld per geval).
- De causale afhankelijkheden in het procesmodel worden gebruikt om de overdrachten in het gebeurtenissenlogboek te tellen.

Sociaal netwerk op basis van overdracht van werk (drempel van 0,1)

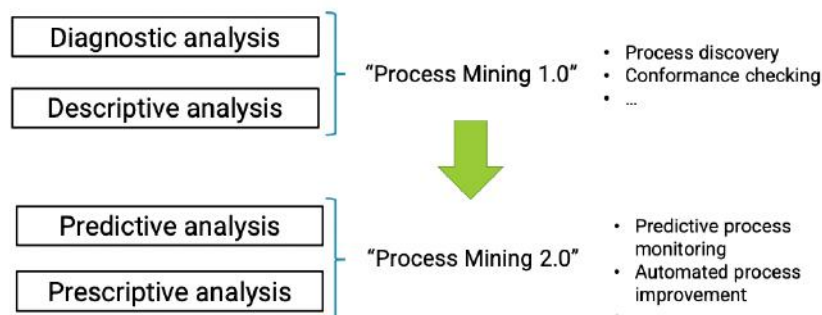


Sociaal netwerk gebaseerd op overdracht van werk



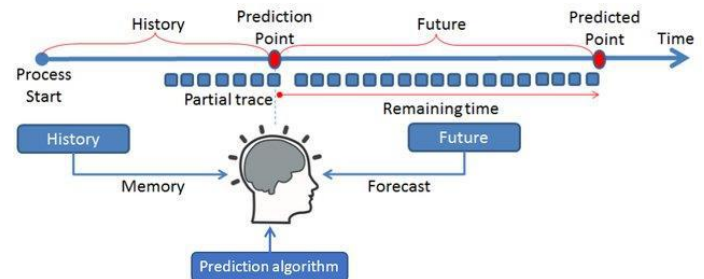
2. Predictive Process Mining

"Process Mining 2.0"

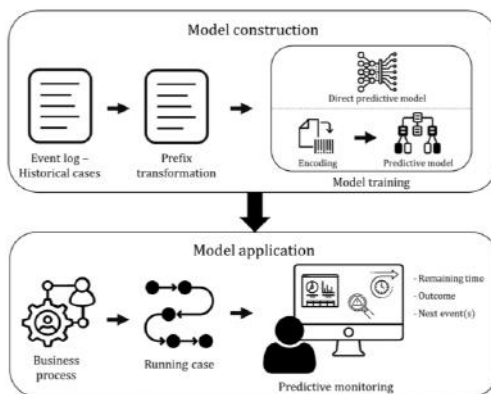


Van beschrijvend naar voorspellend

- Procesontdekking is "unsupervised" (geen label)
- Gesuperviseerde toepassingen worden steeds belangrijker geacht
- Gebeurtenisgegevens gebruiken voor voorspellende doelen
 - o Volgende gebeurtenis
 - o Resterende suffix
 - o Resterende tijd
 - o Resultaat



Voorspellende procesbewaking



→ Een dashboard voor bedrijfsownercreëren met informatie zoals remaining time.. adhv. Model creation

Machine Learning stuurt nieuwe ontwikkelingen

- Beslissingsbomen
 - o Random Forests, XGBoost
- Recurrente neurale netwerken (RNN's)
 - o Neurale netwerken met langetermijngeheugen (LSTM's)
 - o Transformatoren (Encoder-decoder)
- Algemene tegenstrijdige neurale netwerken
- Convolutionaire neurale netwerken (CNN's)

Ons onderzoek

- 1) Seq2Seq-LSTMs voor het voorspellen van complete suffix en resttijden
- Motivatie: Verwerking van bagage op Brussels Airport, traject en uitvoeringstijden van bagage beïnvloed door:
 - o Casuskenmerken: Aankomst/Vertrekkingsvlucht (bv. Vlucht ID, Luchthaven).
 - o Tijdskenmerken: Tijdsinformatie (bijv. uur van de dag, resterende tijd tot gepland vertrek)
 - o Gebeurteniskenmerken: Informatie over gelijktijdig lopende zaken (bv. belasting bij Check-in/Screening/Sorter).
 - Ons onderzoek:
 - o Een modelarchitectuur ontwikkelen die van nature alle relevante informatie kan opnemen om de resterende suffixen en runtimes te voorspellen van zakken die op de luchthaven worden verwerkt.

LSTM's

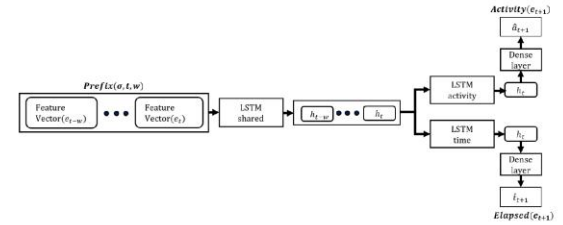
- Een bijzonder type terugkerende neurale netwerken (RNN) met een aangepast ontwerp om het "verdwijnende gradiëntprobleem" aan te pakken.

Klassieke SEP-LSTM's

- Belangrijkste modelcomponenten
- Voorvoegsel van kenmerkvectoren gepropageerd in een gedeelde LSTM-laag
- De gedeelde LSTM-laag stuurt een reeks verborgen toestanden terug.
- Deze reeks wordt doorgegeven aan twee afzonderlijke LSTM-lagen die hun laatst geleerde verborgen toestand teruggeven.
- Verborgen toestanden gebruikt om afzonderlijke voorspellingen te verkrijgen:
 - o Volgende activiteitslabel
 - o Tijd die zal verstrijken tot de volgende gebeurtenis wordt waargenomen

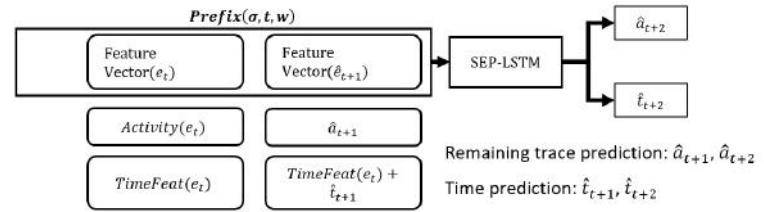
Table 1: Partitioning of traces into prefixes for SEP-LSTM model

t	w	$Prefix(\sigma, t, w)$
1	3	$\langle 0, 0, e_1 \rangle$
2	3	$\langle 0, e_1, e_2 \rangle$
3	3	$\langle e_1, e_2, e_3 \rangle$
4	3	$\langle e_2, e_3, e_4 \rangle$
5	3	$\langle e_3, e_4, e_5 \rangle$
		$\sigma = \langle e_1, e_2, e_3, e_4, e_5, e_6 \rangle$



Hallucinatie

- Het voorspelde activiteitslabel en de verstreken tijd worden gebruikt om een nieuwe gebeurtenis te construeren.
- De voorspelde gebeurtenis wordt toegevoegd aan het oorspronkelijke voorvoegsel.
- Dat op zijn beurt wordt ingevoerd in het SEP-LSTM-model om nieuwe voorspellingen te verkrijgen.

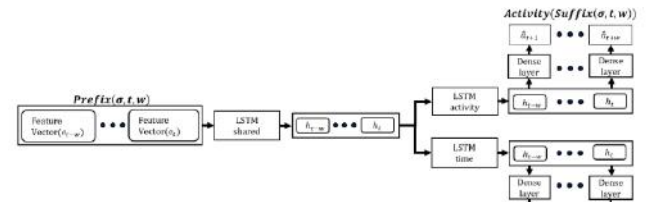


CRTP-LSTM: Seq2Seq-model

- Kenmerkvector en achtervoegsels
- Sporen in gebeurtenissenlogboek worden opnieuw geparseerd in meerdere prefixen (t) van vaste grootte (w) - Elke gebeurtenis in prefix wordt omgezet in een kenmerkvector
- Daarnaast worden twee achtervoegsels geconstrueerd bestaande uit:
 - o Resterende activiteitslabels
 - o Resterende runtijden
- Belangrijkste modelcomponenten:
 - o Voorvoegsel van kenmerkvectoren gepropageerd in een gedeelde LSTM-laag
 - o De gedeelde LSTM-laag stuurt een reeks verborgen toestanden terug.
 - o Deze reeks wordt doorgegeven aan twee afzonderlijke LSTM-lagen.
 - o De twee afzonderlijke LSTM-lagen sturen beide een reeks verborgen toestanden terug.
 - o Verborgen toestanden worden gebruikt om reeksen activiteitslabels en resterende runtijden te voorspellen.

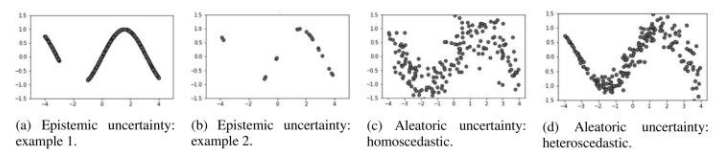
Table 2: Partitioning of traces into prefixes and suffixes for CRTP-LSTM model

t	w	$Prefix(\sigma, t, w)$	$Activity(Suffix(\sigma, t, w))$	$Remtime(Suffix(\sigma, t-1, w))$
1	5	$\langle 0, 0, 0, 0, e_1 \rangle$	$\langle a_2, a_3, a_4, a_5, a_6 \rangle$	$\langle r_1, r_2, r_3, r_4, r_5 \rangle$
2	5	$\langle 0, 0, 0, e_1, e_2 \rangle$	$\langle a_3, a_4, a_5, a_6, 0 \rangle$	$\langle r_2, r_3, r_4, r_5, 0 \rangle$
3	5	$\langle 0, 0, e_1, e_2, e_3 \rangle$	$\langle a_4, a_5, a_6, 0, 0 \rangle$	$\langle r_3, r_4, r_5, 0, 0 \rangle$
4	5	$\langle 0, e_1, e_2, e_3, e_4 \rangle$	$\langle a_5, a_6, 0, 0, 0 \rangle$	$\langle r_4, r_5, 0, 0, 0 \rangle$
5	5	$\langle e_1, e_2, e_3, e_4, e_5 \rangle$	$\langle a_6, 0, 0, 0, 0 \rangle$	$\langle r_5, 0, 0, 0, 0 \rangle$
		$\sigma = \langle e_1, e_2, e_3, e_4, e_5, e_6 \rangle$		

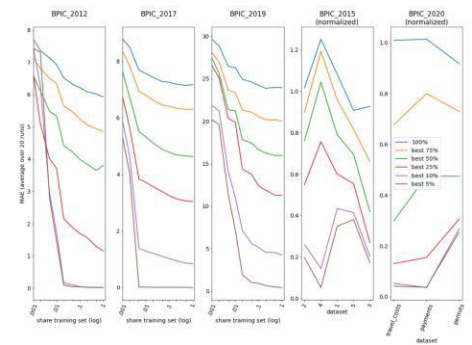


2) Op onzekerheid gebaseerd diep leren voor PPM

- Twee soorten onzekerheid

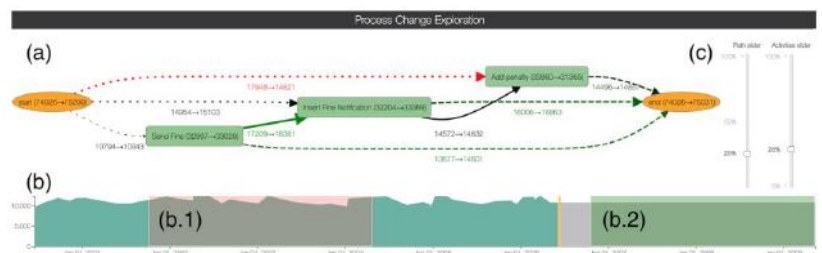
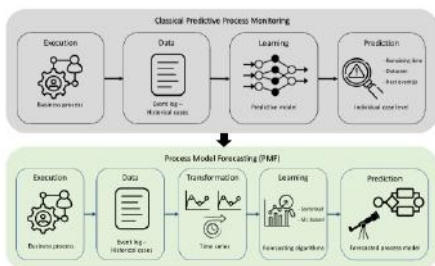


- Wij hebben een techniek ontwikkeld om deze onzekerheden in PPM-modellen te schatten.
- Zo kun je een waarde en de onzekerheid van de voorspelling voorspellen
 - o Beter voor kleinere datasets
 - o Strategieën voor verbetering



3) Procesmodelvoorspelling

- Van operationele voorspellingen tot tactische en strategische inzichten
- We kijken naar heel een model en kijken naar de verleden en voorspellen de toekomst.
- "Blijven mijn knelpunten in de tijd bestaan?", "Zal de ratio van toegekende kredietaanvragen in het volgende kwartaal veranderen?", enz.



3. Tool Support

A. Open-sourcetools

- ProM
 - o Ontwikkeld aan de TU Eindhoven
 - o Enorme lijst van plugins
 - o Voornamelijk gebruikt door academici/onderzoekers
 - o www.promtools.org
- Apromore
 - o Initiatief geleid door de Universiteit van Melbourne
 - o Online platform
 - o www.apromore.org

B. Commercial process mining tools

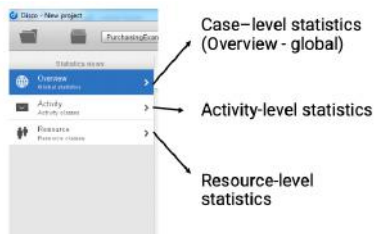
- **Celonis**
- **Disco (Fluxicon)**
- MyInvenio
- Minit
- ProcessGold
- LanaLabs
- ARIS Process Performance Manager
- QPR ProcessAnalyzer
- Interstage Process Discovery
- Discovery Analyst (StereoLOGIC)

Disco

- Commercieel hulpmiddel
 - Ontwikkeld door een Nederlands bedrijf genaamd Fluxicon
- Gestroomlijnd naar de belangrijkste event log analyse technieken
- Belangrijkste voordelen

- Gebruiksvriendelijk (import uit CSV-bestanden/Excel-bestanden)
- Snel
- Grondig inzicht in een event log
- Geavanceerde filtertechnieken
- Academische licentie beschikbaar met uw @student.kuleuven.be e-mail

Categorieën processtatistieken

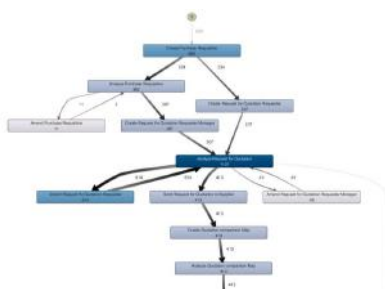


- Statistieken op zaakniveau/ Case-level statistics
 - Algemene gebeurtenislogboekstatistieken (aantal gebeurtenissen, zaakbronnen, activiteiten)
 - Diverse grafieken (bv. gebeurtenissen in de tijd, actieve gevallen in de tijd)
 - Geval-specifieke informatie (aantal gebeurtenissen, duur)
- Statistieken op activiteitsniveau
 - Algemene statistieken: aantal activiteiten, frequentie-informatie
 - Verscheidene grafieken
 - Activiteitspecifieke informatie: frequentie, duur
- Statistieken op middelenniveau
 - Vergelijkbaar met activiteitsniveau
 - Algemene statistieken
 - Grafieken
 - Bronspecifieke informatie

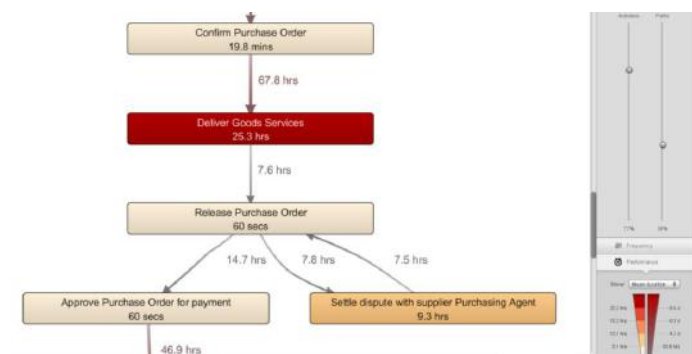
Proceskaarten

- Hulpmiddel voor procesontdekking in Disco
- Het is verwant aan fuzzy miner in ProM
 - Geen hiërarchische procesontdekking (geen groepering van activiteiten in clusters)
- Interactieve abstractie en vereenvoudiging op basis van relatieve frequenties
 - Activiteiten
 - Paden
- Twee verschillende weergaven
 - Frequentie
 - Prestaties

Frequentie-based process map



Performance-based process map



Filteren

- Disco-tool heeft rijke functies voor het filteren van gegevens
- Filteren kan op basis van:
 - Prestaties
 - Tijdsbestek
 - Procesvarianten
 - Attributen
 - Eindpunten
 - Volger
- Meerdere filterregels kunnen achtereenvolgens worden uitgevoerd
- De volgorde waarin verschillende filterregels worden uitgevoerd is van belang!

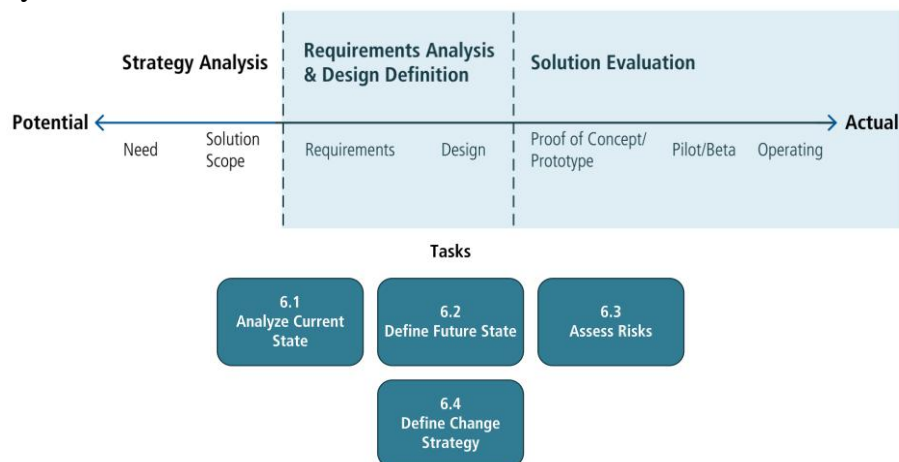
BUSINESS ANALYSE

BA2. Context Analysis

Waar situeren we ons ?



Strategie analyse



⇒ Taken in dit deeldomein van bussines analyse

1. Business Strategy

Wat is strategie?

- Een geïntegreerd geheel van acties gericht op het vergroten van het welzijn en de kracht van een onderneming op lange termijn (Porter, 1980)
- De bepaling van de fundamentele langetermijndoelstellingen van een onderneming, en de vaststelling van acties en de toewijzing van middelen die nodig zijn om deze doelstellingen te verwezenlijken (Chandler, 1962)
- De langetermijnrichting van een organisatie (Johnson et al., 2017)

⇒ Harry Mintzberg (2000) : grondlegger van strategic planning

- **Plan:** idee van hoe je gaat concurreren in de markt en je voert het uit
- **Ploy:** je zit in een dynamische concurrentie
 - Nadenken hoe je met specifieke zetten jouw concurrentie de loef kan afstekken.
 - Spel
- **Pattern:** bepaalde manier van waarde die waardegeneratie hebben en door dat vaak te doen creëer je onbewust een patroon die tot een strategie leidt
- **Position:** je neemt een bepaalde positie t.o.v uw klanten, concurrenten, suppliers...
- **Perspective:** impliciete manier van strategie maar op organisationele aspect. Het gaat meer rond het culturele aanpak van een bedrijf.

Belang van strategie

- Basis voor succes door de uitvoering af te stemmen op de context van de interne en externe omgeving
 - Externe omgeving bekijken en dat vertalen naar een intern plan
- Belangrijke overwegingen
 - Wereldwijde concurrentie
 - Toegenomen complexiteit
 - Voortdurend veranderende verwachtingen van de klant
 - Technologie

Waarom aandacht voor bedrijfsanalyse?

- Voordelen van kennis van de strategische context en het kunnen toepassen ervan
 - Strategische kansen analyseren en bespreken
 - Geloofwaardigheid opbouwen bij het bespreken van de organisatie met belanghebbenden
 - De geschiktheid en afstemming van beslissingen voorafgaand aan en tijdens een veranderingsinitiatief in twijfel te trekken
 - Leiderschap en invloed uitoefenen om strategisch afgestemde veranderingen tot stand te brengen.
- Werken zonder verwijzing naar de strategische context vermindert de waarde van bedrijfsanalysediensten

Inzicht in de strategische context

- Wat is de "huidige toestand" van de organisatie?
 - SWOT analyse
 - Strengths
 - Weaknesses
 - Interne context analysis
 - VMOST
 - Resource audit
 - BSC
 - Growth share matrix
 - Opportunities
 - Threats
 - External context analysis
 - PESTLE
 - Porters Five Forces
- Wat is de gewenste "doeltoestand" van de organisatie?
- Wat is het plan om te bewegen tussen de huidige toestand en de gewenste toestand?
 - Aka strategische missie of routekaart



2. External Context Analysis

Analyse van de externe context

- Extern perspectief
 - Economische groei
 - Monetair beleid
 - Regelgeving
 - Technologische trends
- Gericht op kansen en bedreigingen
- Twee belangrijke methoden/technieken
 - PESTLE-analyse
 - Vijfkrachtenmodel van Porter

PESTLE analysis

P Political	E Economic	S Sociological	T Technological	L Legal	E Environmental
Political Stability	Economic Growth	Population Growth Rate	Technology Incentives	Employment Laws	Environmental Policies
Government Policy	Exchange Rates	Age Distribution	Level of Innovation	Consumer Protection Laws	Climate Change
Tax Policy	Interest Rates	Career Attitudes	Automation	Copyright and Patent Laws	Pressures from NGO's
Foreign Trade Policy	Inflation Rates	Lifestyle Attitudes	R&D Activity	Health & Safety Laws	Climate
Corruption	Unemployment Rates	Safety Emphasis	Technological Awareness	Discrimination Laws	Weather
Labour Law	Disposable Income	Health Consciousness	Technological Change	Antitrust Laws	
Trade Restrictions		Cultural Barriers			

Voeg een extra E toe: PESTLEE – STEEPLE

- E voor Ethisch
 - Denk aan GDPR, AI-regelgeving, enz.
- Kerntechniek voor het beoordelen van de bredere macro-omgeving waarbinnen een organisatie opereert

Vijfkrachtenmodel van Porter



Blue Ocean = kritiek op Porter

- Porter's model niet altijd goed bij "ontwrichtende" verandering, vooral in technologische of technologiegedreven omgevingen

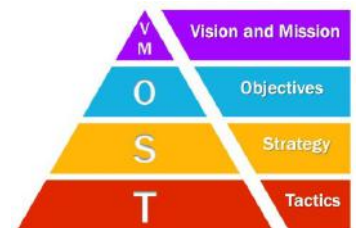
Red Ocean Strategy	VS	Blue Ocean Strategy
Compete in existing market space.		Create uncontested market space.
Beat the competition.		Make the competition irrelevant .
Exploit existing demand.		Create and capture new demand.
Make the value-cost trade-off.		Break the value-cost trade-off.
Align the whole system of a firm's activities with its strategic choice of differentiation or low cost .		Align the whole system of a firm's activities in pursuit of differentiation and low cost .

3. Internal Context Analysis (Interne contextanalyse)

- Intern perspectief
 - Organisatiestrategie
 - Bedrijfsmodel
 - Mogelijkheden
 - Organisatiecultuur en -structuur
- Gericht op sterke en zwakke punten (in plaats van kansen en bedreigingen)
- Technieken: VMOST, middelenaudit, BSC, Boston Box

1) VMOST

- **Visie:** definieert de beoogde toestand voor de organisatie
- **Missie:** Beschrijft wat de organisatie doet of gaat doen
- **Objectives / Doelstellingen:** De specifieke doelstellingen of resultaten die de organisatie wil bereiken
- **Strategie:** De langetermijnaanpak die de organisatie volgt om de visie, missie en doelstellingen te bereiken.
- **Tactiek:** De specifieke en gedetailleerde middelen waarmee de strategie moet worden uitgevoerd. Tactieken worden vaak aangepast wanneer feedback over het succes van de strategie wordt ontvangen.



VMOST-implementatie - kernvragen

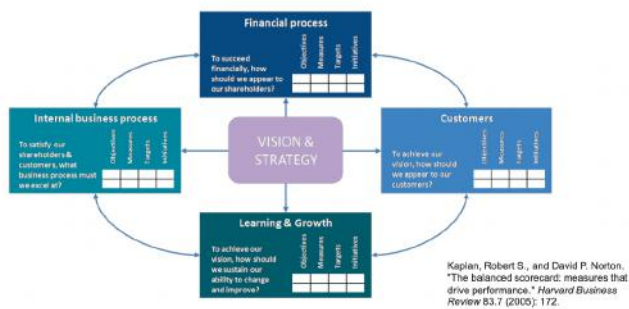
- Bestaat er een VMOST voor de organisatie?
- Zijn de elementen van de VMOST duidelijk gedefinieerd?
- Wordt de VMOST gecommuniceerd en begrepen binnen de organisatie?
- Zijn de visie en missie op elkaar afgestemd?
- Zijn de doelstellingen SMART (Specifiek, Meetbaar, Haalbaar, Relevant, Tijdgebonden)?
- Zijn de doelstellingen evenwichtig?
- Zijn de tactieken afgestemd op de strategie ?

2) Controle van de middelen / Ressource audit

- Identificeren van sterke en zwakke punten per type middel
- Kan worden toegepast op subeenheden (bijv. afdeling, team)

	Resource Type	Resource Heads
Tangible	Physical	Building, Land, Stock, Inventory, Equipment
	Financial	Financial health, Cash flow, Credit
	Human	Staff & resources, Skills & expertise
Intangible	Know-how	Knowledge, Patents, Trademarks
	Reputation	Brand, Goodwill, Image

3) Balances Scorecard (BSC)



Op een gestructureerde manier de KPI's van een onderneming op een strategische niveau te gaan definiëren en opvolgen. Op basis van international bussines proces, financial process, customers, learning & growth. KPI's worden opgevolgd en bijgestuurd indien nodig.

4) Growth share matrix

- BCG-matrix - Boston Box
- Categoriseer producten langs twee dimensies
- Identificeer sterke en zwakke punten van de productportefeuille



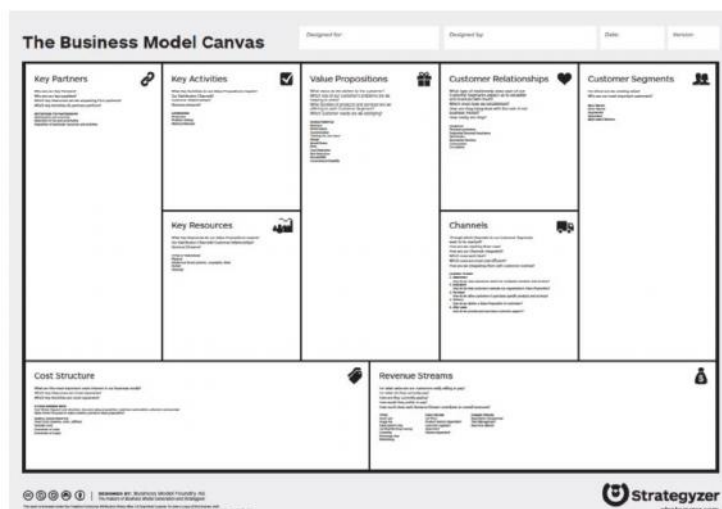
4. Strategy Execution

Ontwikkeling en uitvoering van de strategie

- Uitgaande van de resultaten van de interne en externe contextanalyse
- Analyse van de kloof tussen de huidige en de gewenste toestand
- Belangrijkste technieken
 - 1) Business Model Canvas (BMC)
 - Meest belangrijke voor business analysten want het is het volledigste.
 - 2) Business Capability Model (BCM)
 - Alternatieve aanpak

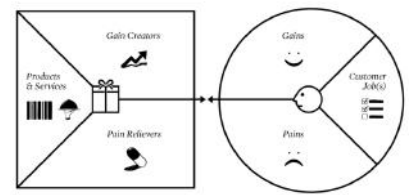
1) Business Model Canvas

- Een bedrijfsmodel beschrijft hoe een organisatie bedrijfswaarde creëert, beheert en levert.



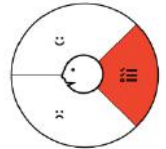
A. Klantsegmenten en waardeproposities

- Wie zijn de belangrijke klanten van de organisatie? Voor wie wordt waarde voorgesteld?
- **Waardepropositie Canvas (naam van het kleiner model)**
 - Klantenfuncties / Customer Jobs
 - Klantenpijn / Customer Pains
 - Voordelen voor de klant / Customer Gain



Customer Jobs

- Jobs beschrijven de dingen die uw klanten in hun werk of leven voor elkaar willen krijgen.
 - Functionele jobs
 - Het uitvoeren of voltooien van specifieke taken of het oplossen van specifieke problemen (tijd bijhouden, verslag schrijven, appartement schoonmaken)
 - Sociale jobs
 - Verwijzen naar momenten waarop de klant 'goed' wil overkomen of sociale waarde of reputatie wil verwerven. Deze jobs gaan over hoe de klant door anderen wil worden gezien.
 - Emotionele klussen
 - Verwijst naar momenten waarop de klant een bepaalde emotionele toestand of gevoel wil bereiken.



Customer Pains

- Customer Pains is alles wat ervoor zorgt dat de klant geïrriteerd raakt voor, tijdens of na het klaren van een klus, of wat hem ervan weerhoudt de klus te klaren.
- Ongewenste resultaten en problemen:
 - Dingen die niet werken
 - Dingen die niet goed werken
 - Dingen die negatieve bijwerkingen hebben
 - Dingen die een slecht gevoel veroorzaken (er slecht uitzien, zich vervelen)



Customer Gains

- Customer Gains zijn wanneer de klant het resultaat of de voordelen krijgt die hij wil.
 - **Required gains** - basisverwachting zoals bellen met smartphone.
 - **Verwachte winst** - basiswinst die wordt verwacht zoals een goed ontwerp van Apple
 - **Gewenste winsten** - meestal winsten die klanten zouden zeggen als ze ernaar gevraagd werden.
 - **Onverwachte voordelen** - gaan verder dan verwacht.



Producten en diensten

→ Een lijst van wat u aanbiedt - als weergave van wat de klant kan zien.

- Fysiek zoals goederen en producten
- Immaterieel zoals IP
- Digitaal zoals muziek
- Financieel zoals beleggingsfondsen

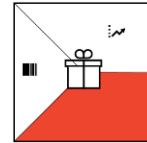


→ Wat doet uw product?

Pain relievers

→ Beschrijft hoe uw product of dienst precies specifieke pijnen van klanten verlicht.

- Besparingen opleveren?
- Zich beter voelen?
- Slechte oplossingen oplossen?
- ...

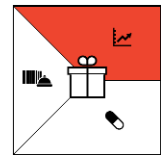


→ Niet alle pijnen zijn even relevant om te verlichten.

Gain creators / Makers van winst

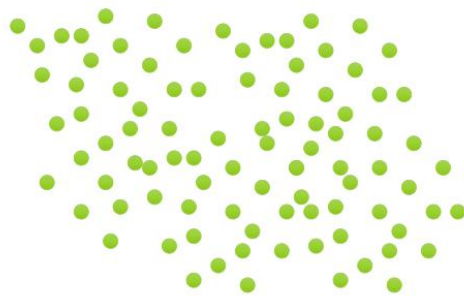
→ Beschrijft hoe het product of de dienst winst oplevert voor de klant.

- Blijde klanten?
- Beter dan bestaande oplossingen?
- Maakt werk of leven gemakkelijker?
- Positieve gevolgen?

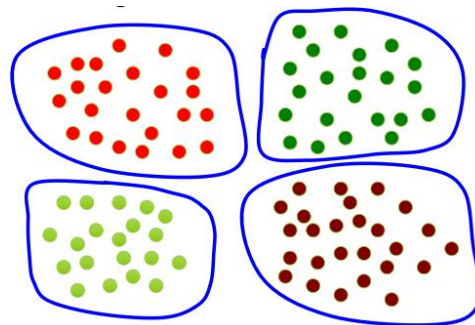


Vergeet niet dat niet alle voordelen even relevant zijn, richt u op de essentiële voordelen.

Klanten in een markt



Marktsegmentatie



Marktsegmentatie

- Marktsegmentatie is het verdelen van een markt in verschillende clusters van klanten.
- In elk segment zijn de klanten verschillend tussen de segmenten, maar vergelijkbaar binnen de segmenten.
 - Tool : Personas

Product Market Fit = Fit is wanneer uw waardevoorstel overeenkomt met de behoeften van de klant. (a.d.h.v. Waardepropositie Canvas)

Hoe kan waarde worden geproduceerd?

- **Nieuwheid** - Nieuwe reeks behoeften die voorheen niet werden waargenomen (mobiele telefoons, ethische fondsen)
- **Prestaties** - Verbetering van de prestaties van producten of diensten (PC's)
- **Maatwerk** - gericht op specifieke behoeften van individuele klanten of segmenten
- **Getting the job done** - helpen bij het klaren van bepaalde klussen (Rolls-Royce straalmotoren)
- **Design** - opvallen door beter design (mode en consumentenelektronica)
- **Merk/status** - waarde door merk/status (Rolex)
- **Prijs** - lagere prijs aanbieden (Ryanair)
- **Kostenreductie** - helpen om kosten te verlagen (Salesforce.com)

- **Risicobeperking** - risico's bij de aankoop van iets beperken (servicegarantie)
- **Toegankelijkheid** - producten of diensten beschikbaar maken voor klanten die geen toegang hadden (beleggingsfondsen, online handel)
- **Gemak/buikbaarheid** - het gebruik van iets vereenvoudigen

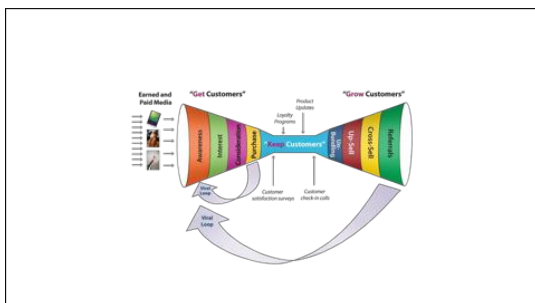
B. Channels & Customer Relationships

- Klantendag in kaart brengen



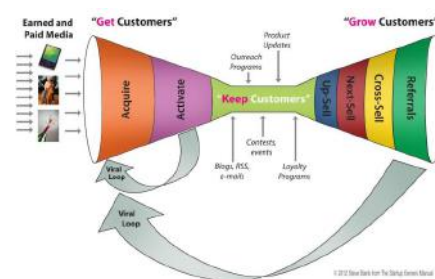
- Kanaal
 - Beschrijft hoe een bedrijf communiceert met zijn klantsegmenten en hen bereikt om zijn waardevoorstel te leveren.

The Funnel – Physical Products



→ Loop

Virtual Products



Soorten klantrelaties

- **Persoonlijke assistentie** - gebaseerd op menselijke interactie
- **Dedicated Persoonlijke assistentie** - toegewijde vertegenwoordiger voor een individuele klant
- **Self-Service** - geen direct contact maar zelf helpen
- **Geautomatiseerde diensten** - mix van zelfbediening en geautomatiseerde processen
- **Communities** - gebruikersgemeenschap waar gebruikers elkaar helpen en bedrijven hun klanten beter begrijpen.
- **Co-Creatie** - samen met de klant waarde creëren

C. Key Resources & Key Activities

Key Resources

- Beschrijft de belangrijkste activa (middelen) die nodig zijn om een bedrijfsmodel te laten werken.



Key Activities

- Beschrijft de belangrijkste dingen die een bedrijf moet doen om zijn bedrijfsmodel te laten werken.



D. Cost structure and revenue streams

Profit = Revenues – Costs

→ Revenues: Revenue Model

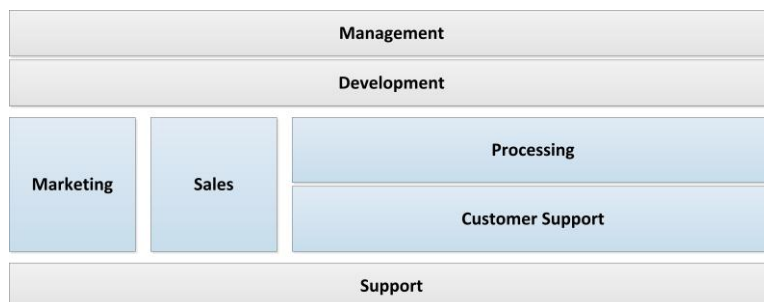
Soorten inkomstenstromen?

- Verkoop
- Gebruikskosten
- Abonnementsvergoeding
- Huur/licentievergoeding
- Makelaarsvergoeding
- Reclame vergoeding

2) Business Capability Model (BCM)

- Een capaciteit is een abstracte verzameling van middelen, processen en technologieën die, in welke combinatie dan ook, een organisatie in staat stelt een gewenst resultaat te bereiken.
- Een capaciteit
 - ... beschrijft "wat" een organisatie doet (uniek en onafhankelijk van elkaar)
 - ... is duurzaam (het "hoe" zal veranderen maar niet het "wat")
 - ... geen duplicatie

Generieke Capability Map



Voorbeeld : Capability map van een verzekeringsbedrijf



Niveaus van bekwaamheid/ vermogen/ capability



High level capability verfijnen en in kleinere capabilities.

→ Daarna capability modelleren

Een capability modelleren



- Performance gap
- Impact value
- Risk

⇒ Analyseren



Waarde van vermogensanalyse

- Bewustzijn van capaciteiten die initiatieven moeten ondersteunen
- Afstemming van bedrijfsinitiatieven op strategische doelen en richtingen
- Identificeren waar (behoeften) een capaciteit moet worden versterkt Voortbouwen op sterke punten voor nieuwe initiatieven
- Identificeren van hiaten en ontwikkelingsbehoeften (later meer hierover)

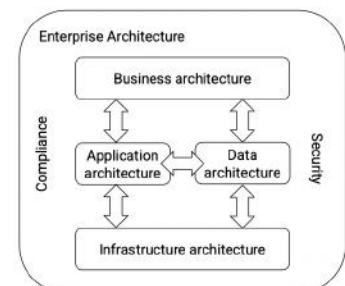
Woord van voorzichtigheid

- Capaciteitsanalyse wordt meestal gedaan op hoog managementniveau, niet het typische werk van een bedrijfsanalist.
- Het punt is om bewust te zijn, capaciteiten te begrijpen en te gebruiken om betere oplossingen te maken.
- Richt u niet op alle capaciteiten, maar alleen op de relevante voor het initiatief.
- Tenzij een capaciteitsanalyse waardevol is, doe het dan niet.

5. Link to Enterprise Architecture (EA)

Enterprise Architecture

- Elke organisatie heeft een EA
- Een EA vertegenwoordigt de kernbouwstenen van de organisatie en laat zien hoe deze is georganiseerd
- EA domeinen.

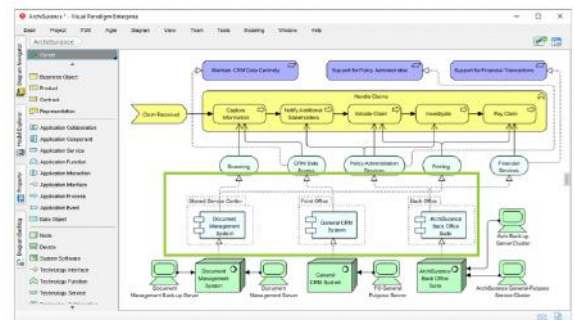


Bedrijfsarchitectuur

- "Een blauwdruk van de onderneming die zorgt voor een gemeenschappelijk begrip van de organisatie en wordt gebruikt om strategische doelstellingen en tactische eisen op elkaar af te stemmen."
- Gebieden
 - Zakelijke motivatie en strategie
 - Organisatiestructuur en -ontwerp
 - Operationeel model
 - Organisatiestructuur
 - Waardeproposities voor producten en diensten
 - Processen
 - Mogelijkheden
 - Informatie

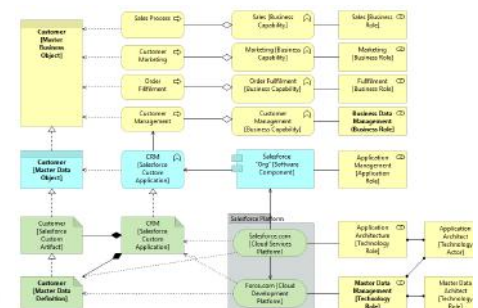
Toepassingsarchitectuur

- Portfolio van bedrijfsapplicaties: benoemen en in kaart brengen
- Afstemming op "bedrijfsarchitectuur" is cruciaal



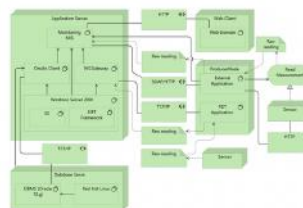
Gegevensarchitectuur

- Beschrijf alle gegevens (objecten) die binnen het bedrijf worden bewaard.
- Afstemming op
 - Bedrijfsarchitectuur
 - Toepassingsarchitectuur (Waar worden de gegevens opgeslagen binnen mijn portefeuille van toepassingen?)



Infrastructuurarchitectuur

- Hardware, OS, netwerken, cloud, enz.



Compliance en beveiligingsarchitectuur

- Compliance
 - Voldoen aan de interne en externe complianceverwachtingen en deze beheren
- Beveiliging
 - Activa beschermen tegen schade, verlies en gevaar

Develop an Adaptive Security Architecture



Belangrijkste raamwerken voor EA

- TOGAF (The Open Group Architectural Framework)
- Zachman Framework
- Federale bedrijfsarchitectuur (FEA)
- Gartner-methode

6. Conclusion

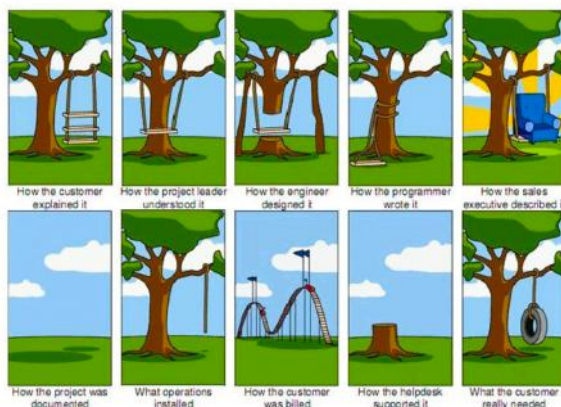
- Wij onderzochten enkele kaders voor
 - o Externe contextanalyse
 - o Interne contextanalyse
 - o Strategie uitvoering
- Link naar EA
- Zorg van de bedrijfsanalisten?
 - o Strategisch bewustzijn maakt betere BA dienstverlening mogelijk
 - o Strategische afstemming is een belangrijke succesfactor voor veel BA projecten
 - o Vooral belangrijk voor senior business analisten

BA3. Requirements Engineering

Waar situeren we ons ?

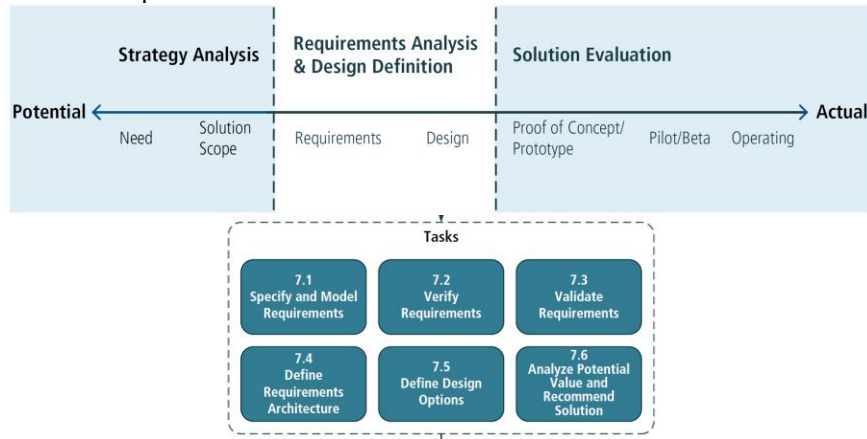


Wat willen mensen ?



- ⇒ Als een business analyst kan je niet “presume”: je kan er niet van uitgaan dat iets gewenst is, of dat iets een oplossing is, of dat een bepaalde eigenschap belangrijker is voor de eigenschap.... Anders krijg je zo’n plaat waar iedereen een verschillend perspectief heeft van wat de oplossing is of zou moeten zijn !
- o Problemen !!
 - o Je moet in het begin alles goed definiëren. Hoe beter je dat doet hoe kleiner de kans is op problemen later. (Hoe later je die problemen ziet, hoe meer ze kosten en hoe kleiner de kans op succes)

Analyse van eisen en ontwerpdefinitie



1. Requirements Definiton and Types

Wat is een eis / requirement ?

- Een eis is
 - o (1) een voorwaarde of mogelijkheid die een belanghebbende nodig heeft om een probleem op te lossen of een doel te bereiken
 - o (2) een voorwaarde of mogelijkheid waaraan een oplossing of oplossingscomponent moet voldoen of waarover deze moet beschikken om aan een contract, norm, specificatie of andere formeel opgelegde documenten te voldoen
 - o een gedocumenteerde weergave van een voorwaarde of vermogen als in (1) of (2)
- Met andere woorden
 - o Een eis is een kenmerk of eigenschap die door een belanghebbende is gevraagd en deel kan uitmaken van een oplossing.
- Zoals deze definitie impliceert, kan een eis onuitgesproken zijn, geïmpliceerd door of afgeleid van andere eisen, of direct aangegeven en beheerd worden.
- Een van de belangrijkste doelstellingen van bedrijfsanalyse is ervoor te zorgen dat de eisen zichtbaar zijn voor en begrepen worden door alle belanghebbenden.

Soorten requirements



- Business Requirements (strategische niveau)
 - o Verklaringen op hoger niveau over de doelen, doelstellingen of behoeften van de onderneming

- De redenen beschrijven waarom een verandering/project is geïnitieerd, de doelstellingen die het project zal bereiken en de maatstaven die zullen worden gebruikt om het succes ervan te meten.
- Stakeholder Requirements
 - Verklaringen van de behoeften van een bepaalde stakeholder of klasse van stakeholders
 - Beschrijven de behoeften die een bepaalde stakeholder heeft en hoe die stakeholder zal interageren met een oplossing.
 - Stakeholder vereisten dienen als een brug tussen business vereisten en de verschillende klassen van oplossingseisen.
- Solution Requirements
 - De kenmerken beschrijven van een oplossing die voldoet aan de bedrijfsvereisten en de eisen van de belanghebbenden
 - Ontwikkeld en gedefinieerd door analyse van eisen
 - Vaak verdeeld in twee subcategorieën, vooral wanneer de eisen een softwareoplossing beschrijven
 - Functionele eisen
 - Wat moet het kunnen
 - Niet-functionele eisen
 - Niet de kern maar wel belangrijk: vb. 99% van de tijd werken
 - Technische aspecten, reliability..
- Functional Requirements
 - Beschrijf het gedrag en de informatie die de oplossing zal beheren
 - Beschrijf mogelijkheden die het systeem zal kunnen uitvoeren in termen van gedragingen of operaties en specifieke acties of reacties van informatietechnologietoepassingen.
- Non-functional Requirements
 - voorwaarden vastleggen die niet rechtstreeks betrekking hebben op het gedrag of de functionaliteit van de oplossing, maar eerder omgevingsvoorwaarden beschrijven waaronder de oplossing doeltreffend moet blijven of kwaliteiten die de systemen moeten hebben
 - Zij worden ook wel kwaliteitseisen of aanvullende eisen genoemd
 - Dit kunnen eisen zijn met betrekking tot capaciteit, snelheid, veiligheid, beschikbaarheid en de informatiearchitectuur en presentatie van de gebruikersinterface.
- Transition requirements
 - Beschrijf de mogelijkheden die een oplossing moet hebben en de voorwaarden waaraan de oplossing moet voldoen om de overgang van de huidige naar de toekomstige toestand te vergemakkelijken, maar die niet nodig zijn zodra de verandering is voltooid.
 - Tijdelijk karakter
 - Voorbeelden
 - Migratie en conversie van gegevens
 - Communicatie met belanghebbenden en opleiding
 - Bedrijfscontinuïteit, releasestrategieën, klantenservice

Overzicht



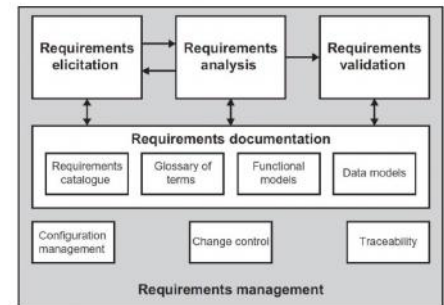
Requirements Engineering

- Het RE-kader verduidelijkt de activiteiten die moeten worden uitgevoerd bij het definiëren van eisen
- Eisen moeten ondubbelzinnig, goed gestructureerd, correct en relevant zijn

2. Requirements Elicitation

Elicitatie van eisen

- Eerste, belangrijkste en cruciale fase van het RE-proces
- Ontdekken, verwerven en uitwerken van eisen
- Verschillende bronnen
 - o Belanghebbenden
 - o Documentatie
 - o Bestaande systemen
- Verschillende technieken
 - o Kwalitatief
 - o Kwantitatief



Kwalitatieve vraagtechnieken

- indrukken en meningen verzamelen
- Soorten:
 - o Samenwerking
 - Workshop
 - Focusgroep
- Een-op-een
 - o Interviews
 - o Vergaderingen
 - o Observatie
- Scenario's
- Prototypes en wireframes
- Analyse van gebruikersrollen/persona's

Kwantitatieve uitvraagtechnieken

- Gericht op volumes, frequenties, ...
- Soorten:
 - o Enquêtes
 - o Vragenlijsten
 - o Analyse van documenten

3. Requirements Analysis

Doelstelling

- Identificeren van eisen:
 - o die elkaar overlappen
 - o in strijd zijn met andere eisen
 - o duplicaten zijn
 - o gescheiden moeten worden in individuele eisen omdat ze samengesteld of complex zijn

Taken

- Categoriseren van eisen
- Definiëren/accepteren van requirements

- Modelleren van requirements
- Prioriteren van eisen

→ Niet echt een stappenplan, het is heel flexibel

Eisen definiëren

- Uitgelokte vereisten onderzoeken, filteren en een goed gevormde set samenstellen
- Ervoor zorgen dat eisen duidelijk, correct en relevant zijn
- Filters:
 - o Het ontrafelen van meerdere eisen
 - o Controleer op overlappende of dubbele eisen
 - o Bevestig relevantie van de eis
 - o Evalueer haalbaarheid
 - o Verwijder conflicten
 - o Bevestig de kwaliteit van de uitdrukking

Evaluatie van de haalbaarheid

- Technische haalbaarheid
- Zakelijke haalbaarheid
- Financiële haalbaarheid

Kwaliteit van uitdrukking → Afvinken van deze lijst voor alle requirements

- Duidelijk
- Beknopt
- Samenhangend
- Relevant
- Ondubbelzinnig
- Correct
- Toetsbaar
- Traceerbaar

Eisen prioriteren

- Het aantal eisen dat in een project wordt verzameld kan enorm zijn, van honderden tot enkele duizenden.
- De requirements vertegenwoordigen wat stakeholders verwachten van een oplossing
- De middelen om die oplossing te bouwen zijn meestal beperkt, zodat het nauwelijks haalbaar is om meteen aan alle geëiste eisen te voldoen
 - o Oplossing: de eisen sorteren volgens hun relatief belang
- Verschillende benaderingen van prioritering:
 - o MoSCoW
 - o Kano-model

MoSCoW-techniek

- MoSCoW is een hulpmiddel om de discussie tussen belanghebbenden en bedrijfsanalisten te prioriteren.
- De techniek maakt gebruik van een nominale schaal om de eisen te sorteren
- MoSCoW is een zeer gebruikelijke techniek voor prioritering; de reden hiervoor is dat het zeer eenvoudig te gebruiken en uit te leggen is, en van toepassing is op verschillende toepassingsdomeinen.
- MoSCoW staat voor:
 - o Must have
 - o Should have
 - o Could have
 - o Would have (Sometimes Won't have)

Must have

- Verwijst naar eisen waaraan het systeem moet voldoen
- Als aan een Must Have eis niet wordt voldaan, kan de oplossing op geen enkele manier aan de belanghebbenden worden geleverd.
- Als ten minste één van de volgende voorwaarden waar is, dan is een eis een Must Have:
 - o Het leveren van de oplossing zonder de eis heeft geen zin voor de belanghebbenden (zij vinden geen nut)
 - o De oplossing leveren zonder de eis is illegaal
 - o De oplossing leveren zonder de eis is onveilig (er is een gevaar voor de belanghebbenden)

Should Have

- Verschilt van de Must Have in die zin dat het niet absoluut noodzakelijk is om aan de eis te voldoen.
- Als de eis niet wordt vervuld zal de belanghebbende ontevreden zijn over de oplossing.
- Als ten minste één van de volgende voorwaarden waar is, dan is een eis een Should Have:
 - o De eis is belangrijk maar niet essentieel
 - o Het kan schadelijk zijn om de oplossing te leveren zonder de eis, maar de oplossing blijft relevant
 - o De oplossing kan niet werken zonder de eis, maar een workaround (tijdelijke oplossing) kan worden gebruikt om dat hiaat op te lossen

Could Have

- Eisen die wenselijk zijn, maar die een matig effect hebben op de tevredenheid van de belanghebbenden
 - o Gewenst, maar minder belangrijk dan Should Have
 - o Een belanghebbende zou kunnen aanvaarden dat de eis wordt geschrapt

Would have (or Won't Have)

- Vereisten die op lange termijn interessant zouden kunnen zijn om te implementeren, maar die op het moment van implementatie geen middelen rechtvaardigen.
 - o Als we meer geld/tijd hadden, zouden we het doen

KANO-model

- KANO werkt op dezelfde manier als MoSCoW, maar maakt gebruik van 5 niveaus
- KANO drukt het belang van een eis voor een belanghebbende uit op basis van de tevredenheid en de mate van uitvoering van een eis

Author	Category 1	Category 2	Category 3	Category 4	Category 5
Kano (1984)	Must-Be	Attractive	One-Dimensional	Indifferent	Reverse
Cadotte & Turgeon (1988)	Dissatisfier	Satisfier	Critical	Neutral	
Brandt (1988)	Minimum Requirement	Value-Enhancing	Hybrid	Unimportant	
Venkitaraman and Jaworski (1993)	Flat	Value-Added	Key	Low	
Brandt and Scharioth (1998)	Basic	Attractive	One-Dimensional	Low-Impact	
Llosa (1997 and 1999)	Basic	Plus	Key	Secondary	
Pohl and Rupp (2011) [3]	Dissatisfier	Delighter	Satisfier		
KanoModel.com	Basic	Excitement	Performance	Indifferent	Reverse

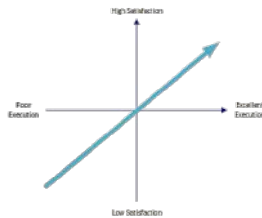
Must-Be

- Wat de stakeholder als vanzelfsprekend beschouwt
- Kan worden gezien als de toegangskosten om de oplossing te initiëren
- Bij goede werking bekom je maar minimum van tevredenheid



One-Dimensional (Should Have)

- Wat de klant van de oplossing verlangt, zodat de oplossing aanvaardbaar is.
- De tevredenheid wordt hoger (proportioneel) naarmate het implementatieniveau toeneemt



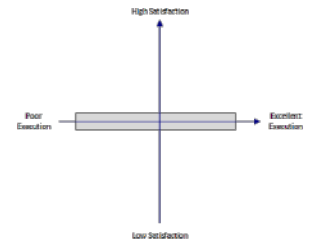
Attractive (Could Have)

- Wat de belanghebbenden niet eisen, maar zeer waarderen indien correct uitgevoerd
- De tevredenheid wordt hoger (niet proportioneel) naarmate het uitvoeringsniveau toeneemt



Indifferent

- Wat de belanghebbenden niet eisen en waarvoor ze onverschillig zijn
- De tevredenheid blijft stabiel naarmate het uitvoeringsniveau stijgt



Reverse

- Wat de klant niet geverifieerd wil zien in zijn oplossing
- De tevredenheid wordt lager (proportioneel) naarmate het implementatieniveau toeneemt
- Karakterstiek die de klant niet wil



4. Requirements Validation

Definitie

- Aantonen dat de eisen het systeem definiëren dat de klant werkelijk wil ("Quality Gateway"), gewoonlijk uitgevoerd door belanghebbenden buiten het project.
- Gerechtigvaardigd door de zeer hoge kosten van fouten in vereisten
 - o Het niet ontdekken van een fout in vereisten kan tot 100 keer meer kosten dan de kosten van een uitvoeringsfout.
 - o De fout wordt versterkt door de specificatie-, ontwerp- en uitvoeringsfase, en wordt significant bij de vrijgave van de oplossing.
- Requirements validatie is anders dan testen
 - o Validatie: verifiëren dat de gedocumenteerde vereisten overeenkomen met de werkelijke verwachtingen van de klant
 - o Testen: Verifiëren dat het geïmplementeerde systeem voldoet aan de specificaties, d.w.z. de verwachtingen die in de eisendocumenten zijn vastgelegd.

Formele vs. informele validatie

- Lineaire projectaanpak
 - o Gedocumenteerde eisen worden beoordeeld en "afgetekend" of "gebaseerd"
- Agile projectaanpak
 - o Minder formeel, maar toch voldoende duidelijk om ze op te nemen in de backlog
 - o Voortdurende verfijning in de backlog tot ze "klaar" zijn om aan een iteratie (bv. sprint) te worden toegewezen.

Criteria

- Geldigheid
 - o Stemmen de eisen overeen met de werkelijke verwachtingen van de klanten? Is het document geldig gezien de elicitation output?
- Consistentie
 - o Zijn er conflicten tussen eisen, of verschillende beschrijvingen van de dezelfde functie?
- Volledigheid
 - o Zijn alle door de klant vereiste functies (en beperkingen) opgenomen?
- Realisme
 - o Kunnen de eisen worden uitgevoerd gezien het budget en de beschikbare technologie?
- Controleerbaarheid
 - o Kunnen de eisen worden gecontroleerd (bijvoorbeeld tijdens het testen)?

Technieken: beoordeling van eisen

- Systematische handmatige analyse door de bedrijfsanalist van gedocumenteerde vereisten
 - o De documentatie moet regelmatig worden herzien terwijl de definitie van de eisen wordt ontwikkeld
 - o De klanten en belanghebbenden moeten worden betrokken bij de reviews (betrokkenheid van belanghebbenden)
 - o Reviews kunnen formeel zijn (met sjablonen, schriftelijke documenten) of informeel (mondelinge feedback)
 - o Goede communicatie tussen ontwikkelaars, klanten en gebruikers kan problemen in een vroeg stadium oplossen

Technieken: prototyping

- Een uitvoerbaar voormodel van de oplossing gebruiken om "in de praktijk" te controleren of de eisen correct zijn
 - o Onvolledige versies maken van de oplossing in ontwikkeling (mock- ups in het geval van software)
 - o Een prototype simuleert doorgaans enkele aspecten van de oplossing, en kan volledig verschillen van het eindproduct.

Technieken: test-case

- Test-case: testvoorbereiding voor testdoeleinden. Hier wordt niets getest - de testbaarheid wordt alleen geverifieerd. Moeilijk uitvoerbare tests wijzen op potentiële problemen in de eisen.
- Een testcase is een reeks testcomponenten
- Elke component beschrijft ingangen, acties of gebeurtenissen en de verwachte respons om te bepalen of een functie van een applicatie/oplossing correct werkt.

5. Requirements Documentation

Belang

- Maakt communicatie mogelijk
- Biedt een basis voor het verzekeren van consistentie in de vereisten
- Biedt een stevige basis om te valideren dat er een nauwkeurig verslag is van wat de oplossing moet bieden.
- Essentieel voor verdere activiteiten om de oplossing te ontwikkelen en te testen.

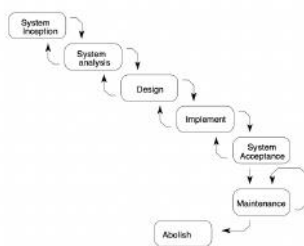
Documentatie stijlen

- Op tekst gebaseerd
 - o Eisencatalogus
 - o Gebruikersverhaal
- Diagrammatisch
 - o Use case model
 - o Gegevensmodel
 - o Bedrijfsprocesmodel

Lineair vs. Agile

- De aanpak van een project heeft gevolgen voor de documentatie
- Lineaire projectaanpak: formeler
 - o Eisen worden voorafgaand aan de ontwikkeling gedocumenteerd
 - o Eisen worden beoordeeld en "afgetekend".
 - o Je kan maar 1 keer uw requirements bepalen. Als ze fout zijn zit je in de problemen.
- Agile project aanpak: meer ad hoc
 - o Requirements worden in grote lijnen (schetsmatig) gedefinieerd, in een vroeg stadium
 - o Wijzigingen worden toegepast wanneer meer details naar voren komen tijdens de ontwikkeling
 - o Eisen evolueren tijdens de ontwikkeling
 - o Heel flexibel manier

Voorbeeld van lineaire aanpak: waterval



→ Je kan niet terug naar de vereisten analyse

Voorbeeld van agile aanpak: scrum



→ Requirements veranderen

Eisencatalogus

- Typisch voor lineaire projecten
- Eenvoudig, tabelvormig formaat voor het definiëren van eisen
- Er bestaan sjabloonlijsten met belangrijke kenmerken
- Opgenomen kenmerken kunnen verschillen afhankelijk van het type project

✓ Identifier	✓ Business area
✓ Name	✓ Stakeholders
✓ Description	✓ Associated non-functional requirements
✓ Source	✓ Related requirements
✓ Owner	✓ Acceptance criteria
✓ Author	✓ Rationale
✓ Type	✓ Version history
✓ Priority	

User Stories

- Vaak gebruikt in agile projecten: "backlog van user stories"
- Een user story definieert de features die actoren van een systeem verlangen.
- Geschreven vanuit het perspectief van een actor of gebruikersrol, meestal informeel
- Snel te ontwikkelen
- De bedoeling is om de geïdentificeerde behoefte te schetsen, maar de details over hoe deze te leveren zijn onderwerp van discussie, prototyping en evolutionaire ontwikkeling.

Ontwikkeling van User stories

- 3C's kader
 - o Kaart: Elke user story wordt gedocumenteerd met behulp van een kaartformaat, waardoor de hoeveelheid informatie wordt beperkt.
 - o Gesprek: Elke user story is de basis voor een conversatie waarin de user story verder wordt uitgediept.
 - o Bevestiging: User stories worden geaccepteerd wanneer ze worden geëvalueerd aan de hand van gedefinieerde acceptatiecriteria.
- Standaard formaat
 - o Als ... (wie is de gebruikersrol of actor?)
 - o Ik wil ... (wat is de mogelijkheid of functie die nodig is?)
 - o Zodat ... (waarom is de user story nuttig voor de gebruikersrol?)
- Verschillende user stories kunnen worden gecombineerd in een epic

Voorbeeld

- Name: View order
 - As a registered customer
 - I want to view the orders I have placed for products
 - So that I can track when the products will be delivered
- Priority: Should have
- Confirmations:
 - Only registered customers are able to view orders
 - Each registered customer can view orders they have placed
 - Only orders placed by a registered customer will be displayed
 - Information about product location will be displayed for orders not yet fulfilled
 - The delivery date will be displayed for all products that have been delivered

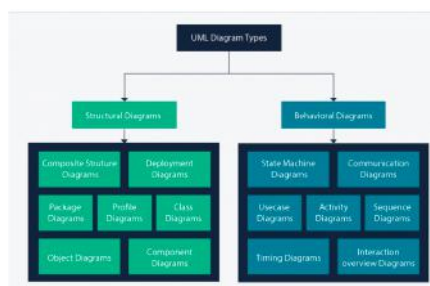
6. Requirements Modelling

Modellering van eisen

- Modellering kan worden toegepast op vele niveaus van eisen
 - o Business/stakeholder
 - Vaak geen echte "modellering"
 - Andere technieken die relevanter zijn: persona's, prototyping, business glossary, customer journey mapping,
 - Uitzondering: use case diagrammen, high-level procesmodellen
 - o Oplossingseisen
 - Niet-functionele eisen→niet echt van toepassing
 - Functionele-eisen→bijzondere aandacht voor diagramtechnieken
 - Gebruik case diagrammen
 - Bedrijfsprocesmodellen
 - Datamodellen

Diagrammatische modellen

- Het beschrijven van eisen in tekstvorm is vaak moeilijk
 - o Ambigüiteit
 - o Slechte precisie
 - o Onduidelijk
 - o Geen holistisch perspectief
- Modellen zijn ideale instrumenten om een beeld te geven van de gehele oplossing en de eisen die binnen het toepassingsgebied vallen te bevestigen.
- UML (Unified Modelling Language) biedt een modelleertaal die een verscheidenheid van verschillende diagramtypes implementeert.



1) Modellering van het bedrijfsperspectief

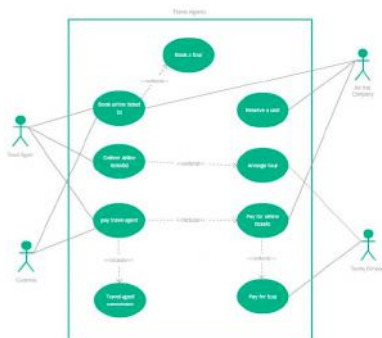
- Visualiseren van de actoren die met het systeem zullen werken en de functies waartoe zij toegang moeten hebben
- Uitstekend voor zakelijke belanghebbenden om hun visie op de oplossing weer te geven
- Het systeem/de oplossing blijft nog grotendeels een zwarte doos

- Verschillende detailniveaus:
 - o Context diagram
 - o Business use case diagram
 - o Systeem use case diagram
- UML Use Case Diagrammen zijn een standaard notatie

UML Use Case Diagram Elementen

- Actoren
 - o Wie of wat dan ook verwacht een dienst van het systeem
 - Gebruikersrollen, externe systemen, enz.
 - o Meestal weergegeven als luciferpoppetjes
 - o Softwareproducten die een actor zijn, kunnen worden aangeduid met een rechthoek
- Use case
 - o Iets dat een actor wil dat een systeem doet
 - o Gesteld doel + acties/verwerkingsstappen die van het systeem worden verwacht om dit doel te bereiken.
 - o Gemodelleerd met behulp van ovals - "werkwoord + zelfstandig naamwoord" naamgevingsconventie
- Systeemgrens
 - o Grote doos rond alle use cases
 - o Associaties
 - o Koppelen van actoren en use cases

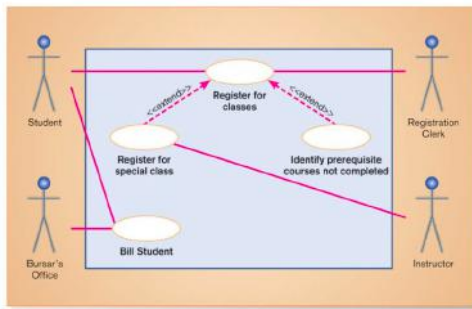
Voorbeeld Use Case diagram



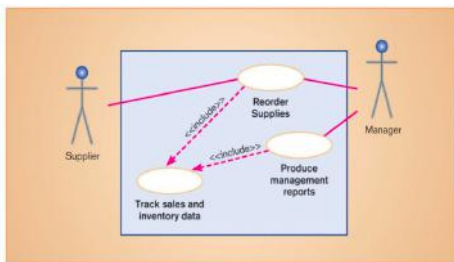
<<include>> en <<extend>>

- <<include>>
 - o Speciaal type link tussen use cases
 - o Laten toe om bepaalde stappen die gemeenschappelijk zijn voor meerdere use cases, te hergebruiken in een aparte use case
- <<extend>>
 - o Een tweede speciaal type link tussen use cases
 - o Voor uitbreidingsscenario's, bv. verwerkingsstappen die tijdens een latere iteratie zullen worden ontwikkeld

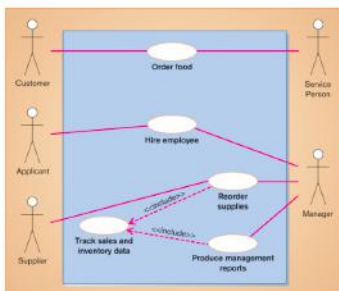
Example Use Case Diagram (2)



Example Use Case Diagram (3)



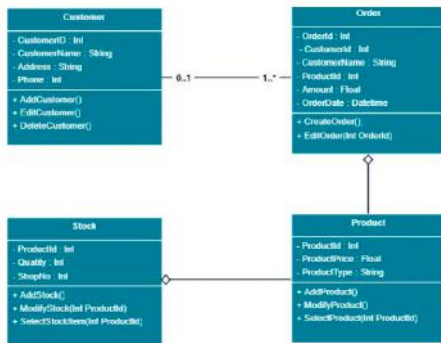
Example Use Case Diagram (4)



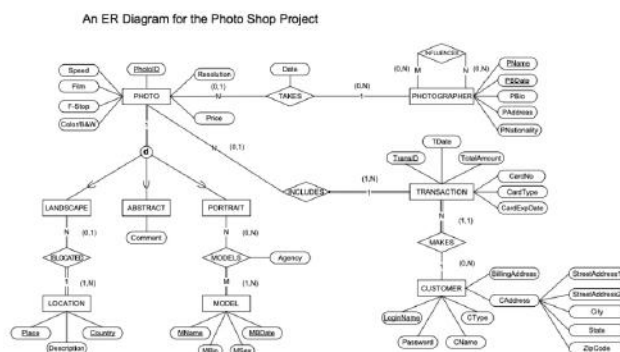
2) Modelleren van het gegevensperspectief

- Een gegevensmodel stelt de belanghebbenden die het systeem gebruiken in staat overeenstemming te bereiken over de gegevens die moeten worden vastgelegd en geraadpleegd.
- De eerste stap kan een gegevenswoordenboek zijn (zakelijke woordenlijst)
- Gegevensmodellering wordt niet alleen gebruikt als basis voor databaseontwerp
- Het is een belangrijke taak voor bedrijfsanalisten om te begrijpen wat de creatie, manipulatie en verwijdering van gegevens regelt.
- Vooral het conceptuele niveau is relevant, soms ook het logische.
- Twee standaardtechnieken
 - o Entiteitsrelatiediagrammen (ERD's)
 - o UML-klassediagrammen

Voorbeeld UML Class Diagram voor Orderverwerking

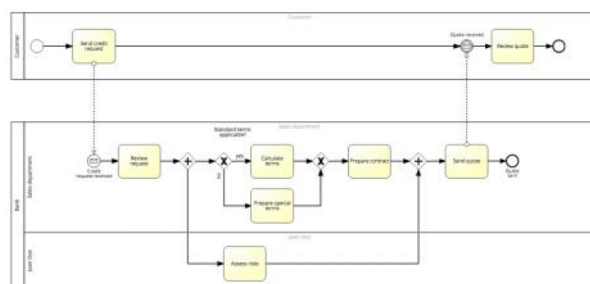


Voorbeeld ER-diagram

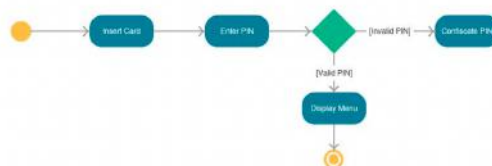


3) Modelleren van het procesperspectief

- Naast gegevens zijn bedrijfsprocessen uiteraard een belangrijk perspectief voor het definiëren en documenteren van functionele eisen.
- Technieken
 - o BPMN



- UML activiteitendiagrammen



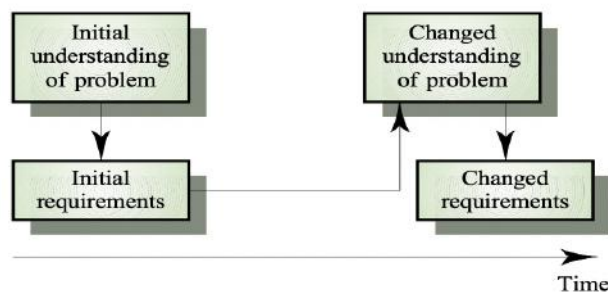
- enz.

7. Requirements Management

Beheer van eisen

- Requirements management is het algemene proces van het beheren van veranderende requirements
- Begint tegelijk met het project, maar gaat door na het einde van het project
- Requirements veranderen omdat:
 - o Verschillende belanghebbenden hebben verschillende eisen en deze zijn vaak tegenstrijdig, verschijnen op verschillende momenten
 - o Nieuwe eisen ontstaan tijdens het proces naarmate de zakelijke behoeften veranderen en een beter begrip van het systeem wordt ontwikkeld
 - o Ervaring van gebruikers met de oplossing of het prototype ervan genereren meer volwassen eisen
 - o Omgeving (domein) verandert

Eisen en verandering



Niveaus van eisenbeheer

- Wij moeten twee soorten eisen onderscheiden, die verschillende managementbenaderingen vereisen
 - o Duurzame eisen
 - Stabiele eisen die zijn afgeleid van de kernactiviteit van de klantorganisatie.
 - Bijvoorbeeld, een ziekenhuis heeft altijd artsen, verpleegkundigen, enz.
 - De eisen kunnen worden afgeleid van domein modellen
 - o Vluchtige eisen
 - Eisen veranderen tijdens de ontwikkeling of wanneer het systeem in gebruik is.
 - In een ziekenhuis zullen de eisen die voortvloeien uit het beleid van de gezondheidszorg waarschijnlijk veranderen.

Soorten vluchtige eisen

- Veranderlijke eisen
 - o Eisen die veranderen als gevolg van de systeemomgeving, het domein
- Opkomende eisen
 - o Eisen die ontstaan naarmate het begrip van het systeem zich ontwikkelt
- Consequentiële eisen
 - o Eisen die voortvloeien uit de invoering van een ander computersysteem of een andere oplossing die in het bedrijf wordt ingevoerd (verandering van werkprocessen)
- Compatibiliteitseisen
 - o Eisen die afhankelijk zijn van andere systemen of organisatorische processen

Traceerbaarheid

⇒ Nagaan waar de veranderingen inzitten

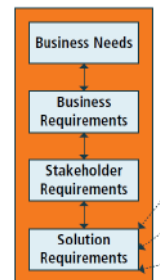
- Traceerbaarheid betreft de relaties tussen eisen, hun bronnen en het systeemontwerp.
 - Bron traceerbaarheid verbindt de eisen met de belanghebbenden die deze eisen hebben voorgesteld.
 - Vereisten traceerbaarheid verbindt de eisen met andere afhankelijke eisen in de oplossingen om te beoordelen hoeveel andere eisen zullen worden beïnvloed door een verandering in de oplossing (impactanalyse).
 - De traceerbaarheid van het ontwerp verbindt de eisen en de ontwerpmodules die ze in de oplossing implementeren.
- Traceerbaarheid betreft de relaties tussen eisen, hun bronnen en het systeemontwerp.
 - Horizontale traceerbaarheid: het traceren van de eis vanaf het begin tot de oplevering.
 - Verticale traceerbaarheid: het traceren van een eis in de hiërarchie van eisen.

Horizontale traceerbaarheid

- Achterwaarts van traceerbaarheid
 - Mogelijkheid om de bron van een functie of vereiste te traceren vanaf elk later punt in de bedrijfsverandering of softwareontwikkelingscyclus.
 - "Wat was de bron voor dit vereiste en wie heeft het naar voren gebracht?"
- Vooruit naar traceerbaarheid
 - Mogelijkheid om een requirement te identificeren en te traceren waar het is ontwikkeld en geïmplementeerd
 - "Wat is er met deze eis gebeurd?"

Verticale traceerbaarheid

- Afstemming op algemene/technische eisen
- Uiteindelijk ook afstemming op bedrijfswaarden, beleid, strategie en doelstellingen



8. Conclusion

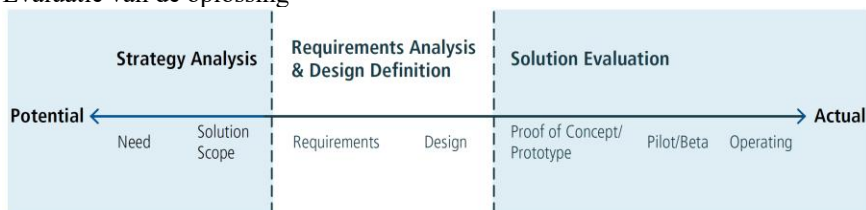
- Eisen zijn een fundamenteel concept bij het uitvoeren van een bedrijfsanalyseproject.
- Requirements Engineering omvat elicitering, analyse, modellering, documentatie, validatie en beheer.
- Requirements documentatie is een cruciaal aspect binnen een business analyse project
- Zakelijke en vooral functionele eisen worden vaak onderworpen aan schematische modellering
 - Zorgen voor duidelijkheid, consistentie, voldoende detail, afstemming, enz.
 - Belangrijkste gezichtspunten:
 - Business: usecase diagrammen
 - Proces: bedrijfsprocesmodellen
 - Gegevens: datamodelen

BA4. Delivery of Solution and Project Management

Waar situeren we ons?



Evaluatie van de oplossing



1. Making the Business Case

Business Case

- Een business case presenteert en evalueert een of meer acties die een probleem aanpakken of de organisatie in staat stellen een zakelijke kans te grijpen.
- Ondersteunt de besluitvorming, overtuigt belanghebbenden om verder te gaan
 - Benadruk de voordelen, niet de kenmerken
 - Verkoop de voordelen voordat u de kosten bespreekt
 - Zorg ervoor dat de 'kopers' de omvang van het probleem of de kans begrijpen voordat tijd, moeite en geld nodig zijn om de oplossing te implementeren.

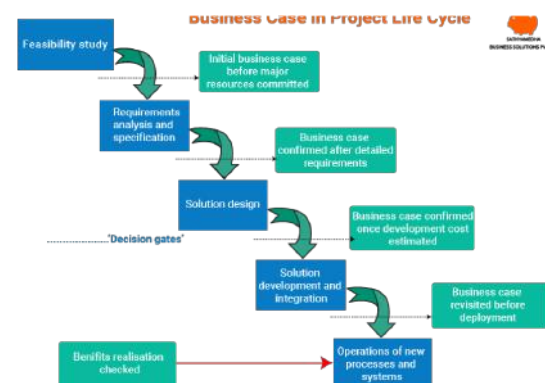
Wanneer een Business Case opstellen

- Onmiddellijk na vooronderzoek
- Levend document: voortdurende evaluatie
- Projecten moeten bepaalde tests doorstaan, vooral met betrekking tot zakelijke levensvatbaarheid, voordat ze naar een volgende fase kunnen gaan ("decision gates")

Structuur van een Business Case

- Inleiding
- Samenvatting
- Beschrijving van de huidige situatie
- Overwogen opties
 - Beschrijving van de opties
 - Analyse van kosten en baten
 - Effectbeoordeling
 - Risicobeoordeling
- Aanbevelingen

Analyse van kosten en baten



- Materiële en immateriële kosten
- Materiële en immateriële baten/ voordelen
- Beoordelingstechnieken
 - Payback (break-even)
 - Discounted Cash Flow (DCF) / Net Present Value (NPV)
 - Internal Rate of Return (IRR)

Sjabloon kosten-batenanalyse

	Initial/ Year 0	Year 1	Year 2	Year 3
Expected Benefits				
Increased Revenues		100	150	150
Reduced Operating Costs		50	50	50
Time Savings		50	50	50
Reduced Costs of Errors		50	50	50
Reduced IT costs		150	100	100
Other Costs		50	50	50
Total Annual Benefits		450	450	450
Costs				
Project Costs	500			
Licenses	100			
Marketing	50			
Ongoing Costs		50	50	50
Other Costs		50	50	50
Total Costs	650	100	100	100
Net Benefit	-650	350	350	350
Accumulated Net Benefits	-650	-300	50	400

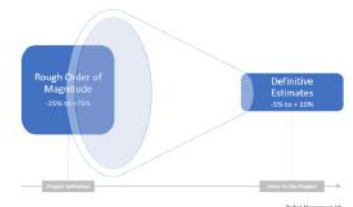
Kosten van de oplossing

- Top-down vs. bottom-up
- Ruwe orde van grootte / Rough Order of Magnitude
- PERT (Driepunten) schatting
- Bottom-up
 - Nauwkeuriger, maar vereist gedetailleerde oplossingsbeschrijving
- Top-down
 - Snel, minder duur, maar minder nauwkeurig



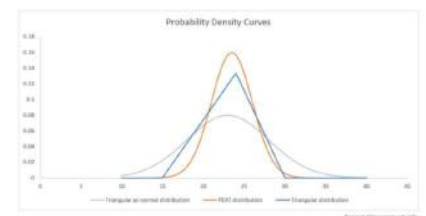
Ruwe orde van grootte / Rough Order of Magnitude (ROM)

- Doel: belanghebbenden en besluitvormers een ruw idee geven van de kosten van het project.
- Nauwkeurigheid van ROM-schattingen is gewoonlijk -25% tot +75% (PMBOK, 6e editie)
- $\text{Bovengrens} = \text{ROM_Estimate} \times 1,75$ - $\text{Ondergrens} = \text{ROM_Estimate} \times 0,75$



PERT-schatting (drie punten)

- Materiedeskundigen stellen drie verschillende schattingen voor
 - Optimistische schatting
 - Pessimistische schatting
 - Meest waarschijnlijke schatting
- PERT: Techniek voor programma-evaluatie en -evaluatie
 - PERT-verdeling overweegt de "meest waarschijnlijke" schatting in vergelijking met een driehoekige (normale) verdeling
 - $E = (O + 4 \cdot M + P) / 6$



Effect- en risicobeoordeling

- Voorbeeld effectgebieden:
 - Organisatiestructuur
 - Relaties tussen diensten
 - Werkmethoden
 - Managementstijl

- Aanwerving
- Beoordeling en promotie
- Relaties met leveranciers
- Risicobeoordeling
 - Beschrijving + Impact + Waarschijnlijkheid + Tegenmaatregelen + Eigendom

Business Cases in een Agile context

- Organisaties stappen af van grote, monolithische projecten naar meer incrementele benaderingen.
- Een aangepaste aanpak voor de productie van business cases is vereist - Een initiële business case wordt geproduceerd op basis van de haalbaarheidsstudie.
- MoSCoW-prioritering wordt toegepast om opties te formuleren.
- Oplevering van elke release is een gelegenheid om de business case opnieuw te bekijken of uit te breiden en de backlog opnieuw te prioriteren op basis van veranderingen in de omgeving.

2. Solution Development Approaches

De leveringsstijl

- Talrijke methoden, normen en levenscycli die kunnen worden gebruikt bij de ontwikkeling van oplossingen om aan bepaalde eisen te voldoen.
- Factoren
 - Rollen: sleutelrollen die tijdens het project moeten worden uitgevoerd
 - Deliverables: door het team op te leveren artefacten
 - Context: kenmerken van het bedrijf en het project
 - Levenscyclus: het aangenomen proces voor ontwikkeling en implementatie

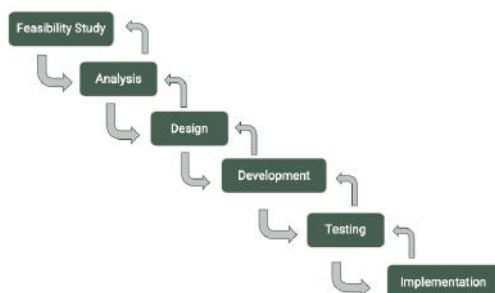
Context

- Cultuur en filosofie
- Zakelijke context
- Beperkingen
- Geprioriteerde bedrijfsbehoeften
- Projectdrijfveren
-

Opleveringslevenscycli / Delivery lifecycles

- Levenscycli voor softwareontwikkeling bieden een duidelijke basis voor het uitvoeren van ontwikkelingsprojecten
- Geeft een opeenvolging van stappen die nodig zijn om een IT-systeem te definiëren, te ontwikkelen en te implementeren.
- Deze levenscycli kunnen gemakkelijk worden aangepast aan bedrijfsveranderingsprojecten, zelfs als er geen software engineering-hoek aanwezig is.
- Belangrijkste SDLC's
 - Waterval
 - V-model
 - Incrementele levenscyclus
 - Iteratieve levenscyclus

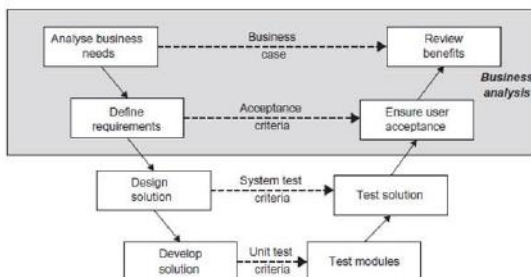
The Waterfall Lifecycle



Belangrijkste kenmerken

- Reeks opeenvolgende fasen
- Elke fase wordt "afgetekend" voordat de volgende begint
- Naar achteren gerichte pijlen geven de noodzaak aan om terug te kijken
- Voordelen
 - Sterke basis voor stevig en duidelijk projectbeheer
 - Moet de levering van een oplossing van hoge kwaliteit ondersteunen
- Nadelen
 - Hoog risico op projectvertragingen door uitputtende kwaliteitsfocus
 - Maakt aanpassing en verandering niet goed mogelijk

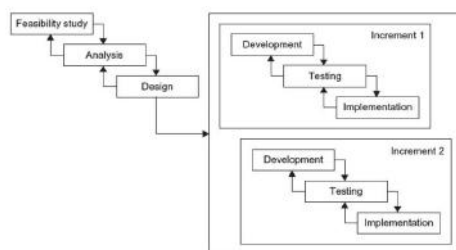
De V model



Belangrijkste kenmerken

- Variant van watervalmodel
- Testfasen worden expliciet getoond
- In elk stadium wordt de afleiding en het gebruik van testcriteria expliciet gemaakt
- Business case wordt na implementatie gebruikt om het succes van de aanbevolen oplossing te beoordelen
- Vergelijkbare voor- en nadelen in vergelijking met waterval

The Incremental Lifecycle

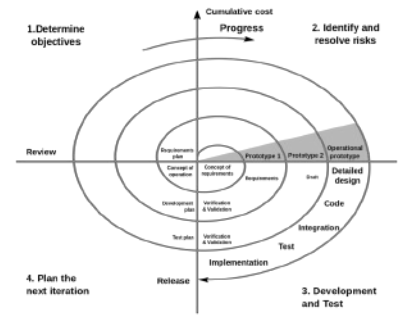


Belangrijkste kenmerken

- Erkent het verschil in belang van bepaalde eisen
- De behoefte blijft bestaan om aan het begin een volledige reeks eisen en een algemeen ontwerp te hebben
- Doel: ontwikkelen en leveren van de oplossing in een reeks stappen
 - Eisen met hoge prioriteit eerst
 - Eisen met lagere prioriteit uitgesteld
- Nadelen
 - Totale leveringskosten waarschijnlijk hoger dan levering in één release
 - Analyse op hoog niveau en ontwerp van de oplossing moeten toekomstbestendig zijn, zoniet zijn inconsistenties, fouten en conflicten mogelijk.

De iteratieve levenscyclus

- Spiraalmodel (Boehm, 1986)
 - Introduceerde het concept van iteraties
 - Sterk gebaseerd op prototyping
- Wordt de basis voor vroege Agile benaderingen, inclusief Rapid Application Development (RAD)
- Nieuwere Agile benaderingen
 - DSDM
 - Scrum



Agile principles



⇒ Belangrijkste kenmerken

- Samenwerking
- Geprioriteerde eisen
- Tijdgebonden iteraties
- Evolutionaire ontwikkeling
- Mondige teams
- Incrementele oplevering
- Continu testen
- Ervaringsleren

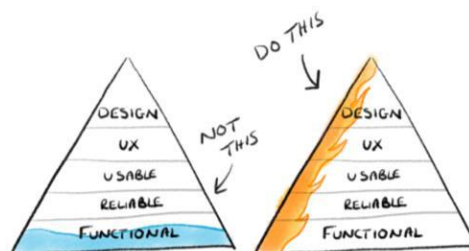
Agile methodologies

- Scrum
- Lean
- Kanban
- Crystal
- Extreme Programming (XP)
- Feature Driven Development (FDD)
- Dynamic System Development Methodology (DSDM)

Scrum



Minimum viable Product (MVP)



Lifecycles - samenvatting

SDLC	Predictive/Adaptive	Linear/Evolutionary
Waterfall	Predictive	Linear
V model	Predictive	Linear
Incremental lifecycle	Predictive	Evolutionary
Agile	Adaptive	Evolutionary

- Agile benaderingen zeer populair tegenwoordig
 - Ze kunnen het snelle tempo van bedrijfsveranderingen aan
 - Voorspellende technieken gaan ervan uit dat zakelijke belanghebbenden aan het begin van een project precies weten wat ze willen. Dit is vaak problematisch.
 - Fragmentatie kan een probleem zijn omdat er geen overzicht is over de beoogde oplossing

Een aanpak kiezen

Predictive Approach	Factor	Agile (Adaptive) Approach
A larger and more complex project	Project size and complexity	A smaller and less complex project
Customers have difficulties being extensively involved during the project duration	Customer availability	Customers are willing and available to frequently be involved during project duration
Unknown or several complex integrations required	Integration level	None or few simple integrations required
Budget/time schedule is fixed and difficult to change/adapt	Flexibility and tolerance for changes	There is flexibility (budget/time)
Solution requires full feature set to be delivered	Time to Market	Solution can be initially launched with limited feature

3. Testing the solution

Evaluatie van de oplossing



Alternatieven

- Niet zoals verwacht?
 - Vertraging
 - Niet gerealiseerd
- Acties
 - Niets doen
 - Organisatorische verandering
 - De oplossing veranderen
 - De oplossing intrekken

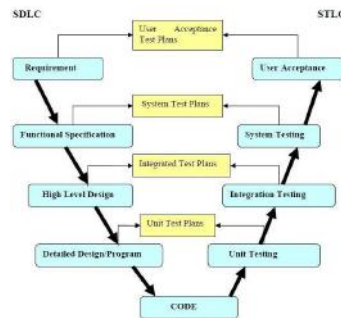
Testen

- Belangrijk onderdeel van de uitvoeringsfase
- Verschillende definities
 - Testen is proberen aan te tonen dat het systeem goed werkt
 - Testen is het uitvoeren van het systeem met als doel fouten te vinden
 - Testen is het vinden van de verschillen tussen de gewenste resultaten en de werkelijke resultaten
 - Testen is het meten van softwarekwaliteit

Testniveaus

- Testen gebeurt op verschillende niveaus:
 - Unit/Module testen (laagste niveau)
 - Integratietesten (het systeem als geheel)
 - Systeemtesten (specifieke aspecten)
 - Veiligheidstest
 - Volumetest
 - Prestatietest
 - Stresstest (extreme situaties)
 - Acceptatietests (door de business)

Levenscyclus testen: V-model



Levenscyclus van softwaretests

- Unit tests: Valideren dat elke eenheid van de software presteert zoals ontworpen
 - Voor elke module, presteert elke bewerking zoals verwacht
 - Voor methode xyz(), zou er een andere methode testxyz() kunnen zijn
- Integratietesten: Controleren of modules in combinatie samenwerken
 - Fouten in de interactie tussen geïntegreerde eenheden blootleggen
 - Allerlei verborgen aannames komen aan het licht wanneer code geschreven door verschillende ontwikkelaars samen worden gebruikt
 - Gewoonlijk liggen de meeste projecten op schema totdat ze dit punt bereiken
- Systeemtesten: Evalueren of het systeem voldoet aan de gespecificeerde eisen
 - Zijn alle user stories geïmplementeerd en functioneren ze correct ?
- Acceptatietests: Evalueren of het systeem voldoet aan de bedrijfsvereisten en beoordelen of het acceptabel is voor oplevering
 - Interne acceptatietesten worden meestal uitgevoerd door leden van de organisatie die de software heeft ontwikkeld, maar die niet direct bij het project betrokken zijn (ontwikkeling of testen). Bijvoorbeeld de leden van Product Management, Sales en/of Customer Support die alfa tests uitvoeren.
 - Externe acceptatietesten worden uitgevoerd door mensen die geen deel uitmaken van de organisatie die de software heeft ontwikkeld. Bijvoorbeeld gebruikers die bèta tests uitvoeren

Integratietests

- Verschillende benaderingen:
 - Bottom-up integratietesten
 - Lege controlemodule die de te testen module aanroept (Driver)
 - Top-down integratietesten
 - Lege modules op het lagere niveau (die de juiste resultaten terugsturen) (Stub)
- Big-bang integratietesten
 - Nadat alle modules zijn getest, worden ze samengevoegd en wordt het hele systeem getest.

Unit/module testen

- Statisch testen (geen uitvoering)
 - Desk checking
 - Gestructureerde walkthroughs
 - Control flow en bereikbaarheid
 - Gegevensstroom
- Dynamisch testen (uitvoering met testgevallen)

- Probleem: als we niet uitputtend kunnen testen, welke testgevallen moeten we dan selecteren?
 - "Black box testing" (functioneel testen)
 - "White box testing" (inhoudelijk testen, structureel testen)-.

Black box testen

- Onafhankelijk van de code (niet naar de code kijken)
- Willekeurige testen (willekeurige selectie van invoerwaarden)
 - Gevarieerde selectie uit het probleemdomein
 - Grenswaarden uit het probleemdomein

Nadeel:

Wat staat er nog meer in de code?

White box testen

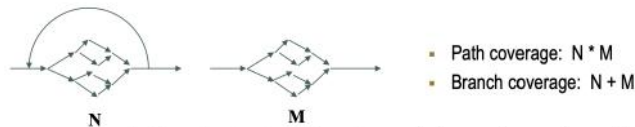
- Kijken naar de inhoud van de module
 - De tester moet een grondige kennis hebben van de te testen broncode
 - (predicaten, paden, selecties, lussen)
- Invoerwaarden selecteren die zoveel mogelijk dekken (paden, takken, lussen, speciale grenzen)

Nadeel:

Wat als het niet in de code staat?

⇒ Traditionele "white box"-criteria

- Paddekking: proberen elk mogelijk pad door de module te testen.



- Takdekking: proberen om ten minste elke tak en elk predicaat eenmaal te testen.
- Gestructureerd testen: grensgevallen van lussen.
- Testen van speciale waarden: uitzonderingen, limieten

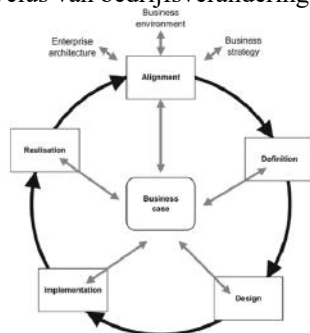
Takdekking

- Testgevallen
 - $a+c$
 - $b+d$
- Niet getest:
 - $a+d$
 - $b+c$



4. Delivering the solution

De levenscyclus van bedrijfsveranderingen

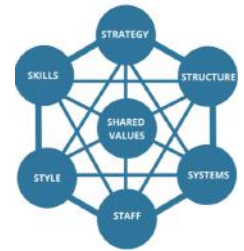


Uitvoeringsfase

- De uitvoering van een veranderingsprogramma vereist planning en zorgvuldige uitvoering.
- Drie belangrijke aspecten
 - Beoordeling van de bedrijfsgereedheid
 - Transitie en migratie
 - Reactie van mensen op verandering

Beoordeling van de bedrijfsgereedheid (business readiness)

- Is het bedrijfsonderdeel waar veranderingen moeten worden doorgevoerd voldoende voorbereid om de nieuwe werkwijzen te aanvaarden en toe te passen?
- Het McKinsey 7S-model kan nuttig zijn.
 - Belangrijkste gebieden om te beoordelen
 - Noodzaak om de "fit" tussen deze gebieden te evalueren



Overgang en migratie

- Migratie van gegevens
- Opleidingssessies
- Opstellen van gebruikershandleidingen, procedurebeschrijvingen, checklists, enz.
- Beslissen over implementatiestrategie
 - Directe omschakeling
 - Parallelle uitvoering
 - Proefdraaien
 - Gefaseerde implementatie

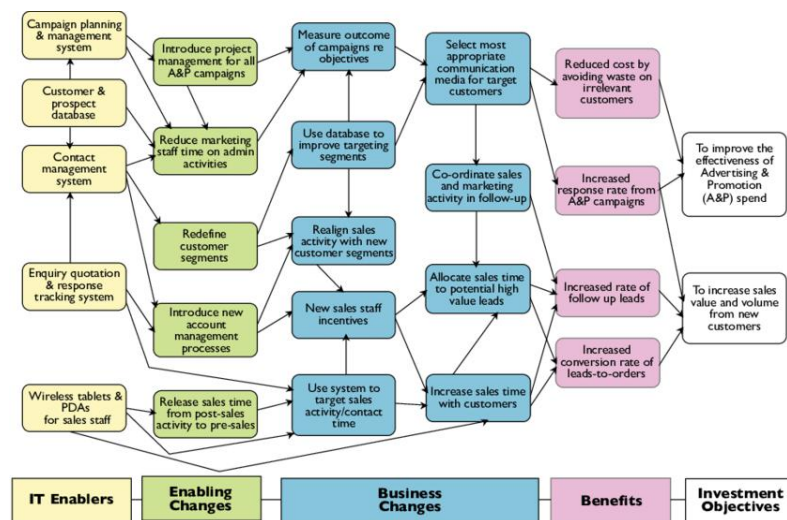
Menselijke reactie op verandering

- SARAH-curve
 - Niet altijd even makkelijk om te bereiken door de reactie van de mensen



Realisatiefase

- Focus op hoe de verwachte bedrijfsvoordelen moeten worden bereikt
- Aspecten
 - Batenplan
 - Netwerk voor afhankelijkheid van voordelen
 - Evaluatie van de voordelen/business case management



5. Conclusion

- Veel aandacht in bedrijfsanalyseprojecten gaat uit naar de analyse en het ontwerp van de oplossing.
- Maar ook het opleveren van de oplossing en het managen van de manier waarop de oplossing wordt ontwikkeld (en getest) zijn cruciaal voor succes
- Belangrijke elementen
 - De business case
 - De stijl van oplevering en de levenscyclus van het projectmanagement
 - Het testen van de oplossing
 - Oplevering van de oplossing

Chapter 2: process mining

What is a process ?

- A business process or business method is a collection of related, structured activities and tasks....
→ vb pizzas

How do you get insight into business processes?

- Map the process and try to investigate
- Interviews with the employees; ask them about their process
- Mystery shopping: go to the pizza shop without them knowing that you are “researching”

- ⇒ Look at process documentation
 - Process models
 - Guidelines
- ⇒ Conduct Manual investigation
 - Expert interviews
 - Consulting
 - But: fragmented, bias ...
- ⇒ Find and analyze data:
 - a. Define KPI'S, monitor Kips
 - b. Analyze data from process
- ⇒ Process mining
 - Full transparency

Combination of 2 disciplines: process science and data science

Process mining theory

- ⇒ Def: a data analytics technology to reconstruct, analyze.....

Event log

Important information :

- Case ID
- Activity
- Timestamp
- ⇒ Trace

You have also additional information

Order to cash process

- Create order
- Confirm order
- Create shipping
- Send goods
- Send invoice
- Receive payment

- ⇒ Some elements we are interested : bussines objectives
 - Revenue increases, working capital opti...