

Finale opleveringsdocument

HAALBAARHEIDSONDERZOEK NAAR HET KOPPELEN VAN ORBA MET FINANCIËLE
SOFTWAREPAKKETTEN UIT DE CENTRUMSTEDEN



Kenniscentrum Vlaamse Steden

Datum 10 juli 2020

Dit rapport mag enkel intern gebruikt worden door de deelnemende centrumsteden en de VGC en mag niet zonder toestemming van het Kenniscentrum Vlaamse Steden openbaar gemaakt worden of ter beschikking gesteld worden aan derden.

Doel van het document

Dit document is de einddeliverable van het haalbaarheidsonderzoek naar het koppelen van ORBA met financiële softwarepakketten uit de centrumsteden.

Document Control

Datum	Versie	Auteur	Opmerkingen
12/06/2020	001	Annelies Fiers	Initiële versie
18/06/2020	002	Sylvie Baes	Update versie
22/06/2020	003	Peter Hellemans	Update versie
10/07/2020	004	Koen Vlemincx	Update versie

Distributielijst

Naam	Organisatie	Functie

Documenten lijst

Naam	Organisatie	Referentie

1. MANAGEMENT SAMENVATTING

Eén van de strategische lijnen die in het kader van de samenwerkingsovereenkomst met de 13 centrumsteden en het Kenniscentrum werd bepaald, is het domein O(mgeving) B(eleid) M(anagement) informatie.

In dit kader werd Cronos gevraagd een haalbaarheidsonderzoek uit te voeren, op vraag van het Kenniscentrum en de centrumsteden, om de haalbaarheid van een koppeling tussen ORBA (Opvolging Realisatie Bestuursakkoord) met de in de steden gebruikte financiële pakketten te onderzoeken.

Daarbij werden drie te onderzoeken scenario's vooropgesteld:

- Scenario 1 : Directe koppeling tussen ORBA en de financiële software via API's
- Scenario 2a: "Single Purpose" Data warehouse project
- Scenario 2b: Toekomstgericht Data warehouse project

Met de bedoeling te bepalen in welke mate elk scenario de behoeften van de centrumsteden vervuld.

Bij de opdracht werden de behoeften op hoog niveau als volgt bepaald:

- Onmiddellijke, punctuele uitwisseling van gegevens tussen beide systemen;
- Geïntegreerde rapportering;
- Rapportering op onderwerp-niveau:
- (Financiële) opvolging meerjarenplanning.

Door de status van projecten, acties, actieplannen en beleidsdoelstellingen te linken met de financiële informatie krijgen verschillende betrokkenen in de organisatie, op hun maat, een overzicht van essentiële project- en financiële informatie.

Analyse Fase – Behoeften analyse en AS-IS inzicht

Vanuit het opdracht-kader hebben we vooreerst de functionele behoeften verder geanalyseerd. Hieruit is gebleken dat:

- Er een hoge gezamenlijke behoefte is m.b.t. pure ORBA projectopvolging en financiële opvolging van doelstellingen, actieplannen en acties;
- Bovendien werd er door een beduidend aantal steden een financiële opvolging op project niveau gewenst
- Tot slot werd ook een meer uitgebreid financiële rapportering vanuit het financiële systeem vooropgesteld.

Daarnaast werd ook bevestigd dat de behoefte naar verdere uitbreiding van een rapporteringsomgeving naar andere data en domeinen aanwezig is.

Naar niet functionele behoeften en technologie standaarden is er een grote consensus onder de steden rond het gebruik van Microsoft BI technologie, omdat deze momenteel bij hen reeds ingeburgerd is.

Ook hebben we gedurende deze fase onszelf een beter inzicht verschaft op de bestaande data warehouse situatie van de verschillende centrumsteden.

Definitie Fase – Scenario analyses

Goed gewapend met de nodige functionele en niet-functionele behoeften analyse en een degelijk inzicht in de bestaande situatie hebben we vervolgens de verschillende scenario naast elkaar gezet met betrekking tot hun haalbaarheid en de voor- en nadelen van deze scenario's.

Door onze iteratieve aanpak met de centrumsteden en het Kenniscentrum hebben we de definities van de verschillende scenario's verder kunnen scherp stellen.

Eén van de eerste en meteen belangrijke vaststellingen in deze fase is de gedeeltelijke incompatibiliteit tussen het ORBA systeem en de financiële systemen.

ORBA Niveaus	Mercurius	New Horizon	UBW	SAP Gent
Beleidsdoelstelling	✓	✓	✓	✓
Actieplan	✓	✓	✓	✓
Actie	✓	✓	✓	✓
Project	✗	✗	?	?

Figuur 1 – Details opvolgingsniveau's

Om financiële data te kunnen opvolgen / rapporteren op projectniveau zal een uitbreiding van het financiële pakket van o.a. CIPAL & CEVI noodzakelijk zijn

In de definitie van de verschillende scenario's zijn we uiteindelijk tot volgende definities gekomen:

- **Scenario 1** : Directe koppeling tussen ORBA en de financiële software via API's
 - Via API, informatie uit ORBA op een gestandaardiseerde manier ontsluiten
- **Scenario 2a** : "Single purpose" data warehouse project
 - Het enige doel hier is gegevens uit ORBA combineren met specifieke gegevens uit financiële software pakketten (enkel die specifieke informatie wordt uit de financiële pakketten ontsloten).
 - Focus ligt op de door de centrumsteden op korte termijn geïnventariseerde informatie behoeften voor de rapportering tussen ORBA en het financieel systeem.
 - Deze oplossing heeft geen blauwdruk op basis van een erkende referentie architectuur.
- **Scenario 2b** : Toekomstgericht data warehouse project
 - Het doel is een data warehouse uit te werken volgens duurzame data warehouse concepten die toelaten om informatie ook uit andere software toepassingen en of thema's bij te voegen.

- Alle financiële data relevant voor financiële rapportering wordt uit de financiële pakketten gehaald, dus niet enkel die data die nodig is voor de rapportering met ORBA.
- De oplossing dient gebaseerd te zijn op een Data warehouse blauwdruk opgezet met meerdere lagen volgens een erkende referentie architectuur (bv. Vooropgestelde referentie architectuur volgens B. Inmon (architectuur Brugge), of de referentie architectuur volgens R. Kimball (architectuur Antwerpen))

Afwegingsfase – Voor- en nadelen scenario's, deelbaarheid en hefboomen, korte en lange termijn

Voor scenario 1 is onze analyse kort en bondig geweest. Het gebruik van ORBA als rapportering omgeving voor zowel ORBA rapportering als gemengde rapportering tussen ORBA en het financieel systeem is een niet geëigende oplossing die ten zeerste af te raden is. Deze oplossing is dan ook nog eens sterk afhankelijk van de nodige medewerking van de leveranciers van de financiële systemen en is zeer onderhoudsonvriendelijk.

Conclusie : Scenario 1 is geen valabele oplossing en wordt door ons sterk afgeraden

De afwegingen tussen scenario 2a en 2b zijn een stuk uitgebreider en bevatten veel meer nuances.

Onder de “single purpose” definitie van scenario 2a verstaan we dat het enige doel van deze oplossing erin bestaat om data uit ORBA te combineren met de geassocieerde gegevens uit financiële software pakketten. In dit scenario worden specifieke data marts gebouwd en worden niet alle goede principes vanuit de markt erkende referentie architecturen toegepast.

Belangrijkste bedenkingen met betrekking tot scenario 2a zijn:

- Snelle oplossing voor de korte termijn en geëigend wanneer toekomstig geen bijkomende uitbreidingen gewenst worden.
- Is een oplossing die niet geëigend is, in zijn “single purpose” vorm voor de financiële rapportering naast de gecombineerde ORBA – Financiële rapportering, ondanks de duidelijke vraag, zoals bleek uit onze analyse fase.
- Door een intrinsiek lagere deelbaarheid van 2a zou gemiddelde kost per stad hoger kunnen liggen dan voor scenario 2b.
- Scenario 2a zal snel naar 2b moeten evolueren bij de minste extra behoeften anders zullen de kosten van de diverse puntoplossingen snel deze van de structurele 2b oplossing overstijgen.

Conclusie : Scenario 2a is mogelijk, maar is eerder een korte termijn oplossing is, omdat telkens er wijzigingen nodig zou zijn in de toekomst er een volledige aanpassing van de code nodig is en de totale kosten snel deze van Scenario 2b kunnen overstijgen

In Scenario 2b wordt een oplossing gebouwd volgens duurzame data warehouse principes die zullen toelaten om ook andere analyses uit het systeem te bekomen dan de ORBA – Financiële rapportering.

Met de oplossing uit dit scenario kunnen ook uitgebreide financiële rapportering gebeuren, met data die ver voorbij gaan aan het bereik van de ORBA opvolging. Deze oplossing laat trouwens ook toe op termijn nog extra databronnen toe te voegen.

Volgens de referentie architectuur (Patroon B. Imnon) wordt een integratie laag gebouwd. Die ontkoppelt de bron-en publicatie laag waardoor impact van wijzigingen veel kleiner wordt en ervoor zorgt dat de oplossing deelbaar is door een groter aantal steden, ook al hebben ze verschillende financiële pakketten.

Deelbaarheid van de investeringen

Binnen de afwegingsfase hebben we ook de deelbaarheid van de verschillende investeringen (Code ontwikkeling en onderhoud, Platform opzet en onderhoud) grondig bekeken.

De realisatie van een dergelijk data warehouse omgevingen, volgens scenario 2a of 2b, kunnen de centrumsteden al dan niet trachten te delen. De voor en nadelen van deze keuzes worden uitgebreid toegelicht in dit rapport.

Hoewel de toelichting van de verschillende analyses hier te ver zou leiden, komt het er samenvattend op neer dat de kosten van ontwikkeling, onderhoud, platform en ondersteuning en de daarbij horende kennis en inzichten maximaal kunnen gedeeld worden maar men dan rekening dient te houden met de andere centrumsteden qua keuzes, verdere uitbreidbaarheid en timing van het project.

Conclusie : Een gedeelde omgeving en code biedt het hoogste schaalvoordeel voor de centrumsteden. Dit gaat echter wel ten koste van de individualisatiegraad.

Verder wordt in dit document ook nog beschreven hoe de (gedeelde) oplossingen verder door de centrumsteden kunnen uitgebreid worden op individuele basis. Hiervoor zijn er echter een aantal spelregels in acht te nemen die wel éénduidig dienen gevolgd te worden.

Conclusie : Geïndividualiseerde uitbreidingen op de gedeelde omgeving zijn mogelijk, maar zijn afhankelijk van het type uitbreiding gebonden aan een aantal gemeenschappelijke spelregels die dienen gerespecteerd te worden.

Ook belichten we in dit haalbaarheidsonderzoek de eventuele keuze tussen een publieke of een private Cloud gebaseerde oplossing.

Conclusie : Beide zijn goede alternatieven. Keuze zal ingegeven worden door berekening van de kosten en gevoelsmatige voorkeur van de Centrumsteden

Bestaande data warehouse integratie

Binnen de afwegingsfase hebben we ook de situatie bekeken voor steden die reeds in meer of mindere mate een uitgebouwde data warehouse omgeving hebben met een uitgebouwde data warehouse ondersteuning binnen de context van de stad.

Van bovenaf bekeken komt het er op neer dat centrumsteden die reeds een data warehouse hebben gebouwd een keuze kunnen maken tussen 3 opties, met name:

1. Men beslist om de ORBA en financiële data in het eigen data warehouse via eigen ontwikkeling te integreren.
2. Men verkiest om in te stappen in één van de gedefinieerde oplossingen en integreert deze oplossing verder met zijn eigen data warehouse.
3. Men verkiest om in te stappen in één van de gedefinieerde oplossingen en integreert deze oplossing niet met zijn eigen data warehouse maar gebruikt ze beide als gescheiden omgevingen.

Beslissingsfase – Keuze afwegingen

Gewapend met al deze afwegingsinformatie kunnen de centrumsteden individueel en in gemeenschap een afweging maken van de keuzes die ze willen nemen m.b.t. de verdere exploratie van dit haalbaarheidsonderzoek.

Hiervoor werd ook een beslissingsschema uitgewerkt dat kan helpen als leidraad voor de beslissingen die finaal dienen genomen te worden.

Vanuit dit haalbaarheidsonderzoek kunnen we geen beslissing definiëren per centrumstad, maar enkel een voorkeur oplossing aanreiken voor de centrumsteden in gemeenschap.

Conclusie : We geloven dat een gedeelde omgeving op basis van de scenario 2b omgeving de economisch en operationeel meest gunstige oplossing is voor de centrumsteden op middellange en lange termijn.

Projectfase – Vervolgstappen

Tot slot bevat dit haalbaarheidsonderzoek een eerste insteek met betrekking tot de volgende stappen die kunnen gezet worden na deze fase.

Om het data warehouse met de ORBA en financiële data te bouwen stellen we voor om dit progressief op te bouwen in iteraties. Op deze manier kan men snel toegevoegde waarde realiseren en ervaringen opdoen op een relatief beperkte scope, die meegenomen worden bij een verdere iteratie.

We stellen voor om de iteraties in de volgende volgorde te implementeren:

- Eerst, de combinatie ORBA data, financiële informatie tot op actie niveau
- Daarna de bijkomende financiële data
- Daarna op het moment dat ook de financiële systemen zijn aangepast, de financiële opvolging op project niveau

Een data warehouse implementatie dient niet beschouwd te worden als een éénmalig project, maar wordt beter bekeken als een traject.

Succesvolle implementaties zullen aanleiding geven tot meer en meer data gedreven opdrachten om beslissingen beter onderbouwd te kunnen nemen op basis van data.

Verder mag men er ook vanuit gaan dat bestaande omgevingen steeds een evolutief onderhoud zullen vereisen ten gevolge van een wijzigend contextueel kader en wijzigende behoeften.

We hopen alvast dat dit haalbaarheidsonderzoek een betrouwbare gids mag zijn voor de data gedreven reis die kan ingezet worden bij de uitwerking van de aanbevelingen.

INHOUDSTAFEL

1. Management samenvatting.....	3
Inhoudstafel.....	9
2. Introductie.....	11
2.1. Context	11
2.2. Plan van Aanpak	12
2.2.1. Overzichtsplan van aanpak	12
2.2.2. Onderzoeksgebieden	12
3. Behoeften Analyse.....	14
3.1. AS-IS.....	14
3.1.1. Omgeving per stad	14
3.1.2. Opzet van de bronsystemen	15
3.2. Functionele behoeften	16
3.2.1. Overzicht informatie behoeften	16
3.2.2. Verrijking en analyse met andere data	17
3.2.3. Type rapportering	17
3.3. Mappen van de behoeften en opvolgingsniveaus per stad	18
3.4. Niet-functionele behoeften-analyse	19
3.4.1. Type gebruikers en ondersteuning	19
3.4.2. BI Infrastructuur.....	19
3.4.3. Veiligheid	19
3.4.4. BI-Technologie	20
4. Technische haalbaarheid.....	21
4.1. Definitie van de vooropgestelde alternatieve scenario's.....	21
4.1.1. Scenario 1 : Directe koppeling tussen ORBA en de financiële software via API's.....	21
4.1.2. Scenario 2a – “Single Purpose” data warehouse project	22
4.1.3. Scenario 2b – Toekomstgericht datawarehouse project	23
4.1.4. Conclusie Scenario 2a versus 2b	25
4.2. Deelbaarheid van de oplossing	25
4.2.1. Schaalvoordeel tegenover individualisatie graad	25

4.2.2.	Enkelvoudige of gescheiden data warehouse	27
4.2.3.	Private of public Cloud oplossing	28
4.3.	Uitbreidbaarheid van de oplossing	29
4.3.1.	Uitbreidbaarheid van de gezamenlijke scenario 2a oplossing versus scenario 2b oplossing	29
4.3.2.	Draagvlak voor verdere gezamenlijke uitbreiding van de ORBA / financiële data oplossing	30
4.3.3.	Mogelijkheden tot uitbreiding van de Scenario 2b oplossing	30
4.4.	Integratie van de oplossing met een bestaand data warehouse	34
4.4.1.	Optie 1:	34
4.4.2.	Optie 2:	35
4.4.3.	Optie 3:	36
4.5.	Bronextracties	37
4.6.	Outbound API	37
4.6.1.	Gekozen technologie	37
4.6.2.	Backward compatibiliteit	39
4.6.3.	Authenticatie & autorisatie	40
4.6.4.	Overzicht van de API endpoints	41
4.7.	Data model	45
4.8.	Beslissingsboom	47
5.	Financiële overzichtstabel	48
6.	Implementatie Plan.....	51

2. INTRODUCTIE

2.1. Context

In het kader van de samenwerkingsovereenkomst met de 13 centrumsteden en VVSG werden de thema's/ programma's/ uitdagingen bepaald die voor de centrumsteden de volgende jaren prioritair op te nemen zijn in de planning/werking van het Kenniscentrum. Deze prioriteiten werden in strategische lijnen gebundeld waaronder O(mgeving) B(eleid) M(anagement) informatie.

Binnen de strategische lijn O(mgeving) B(eleid) M(anagement) informatie werd programmaspoor 1 als volgt gedefinieerd:

De nood aan gebruiksvriendelijke data-visualisatietools laat zich zowel op het vlak van omgevingsdata als qua beleids- en managementinformatie voelen. De omgevingsanalyses gehanteerd in de beleidsvoorbereiding vragen 'leesbare' informatie over de context waarin beleid tot stand komt. De monitoring van lokale meerjarenplannen en de opvolging van concrete actieplannen en acties vergen geïntegreerde rapporteringssystemen waarin data uit diverse bronnen en databanken verbonden en geïntegreerd raken. Geavanceerde visualisatiesystemen verschaffen het college en de leidinggevenden daarbij inzicht in de recentste status en knelpunten, in vergelijkende gegevens en in de realisatievoortgang.

ORBA (Opvolging Realisatie Bestuursakkoord) is een registratie-en monitorinstrument dat een organisatie in staat stelt om de realisatiegraad van hun doelstellingen en de bijdrage vanuit de reguliere werking op te volgen. ORBA werd naar analogie van de Antwerpse ORBA oplossing ontwikkeld en in 8 deelnemende steden en één OCMW voor de opvolging van de meerjarenplanning 2020-2025 uitgerold.

In dit haalbaarheidsonderzoek wordt op vraag van het Kenniscentrum en de centrumsteden, de haalbaarheid van een koppeling met de in de steden gebruikte financiële pakketten te onderzoeken.

Daarbij worden drie te onderzoeken scenario's vooropgesteld:

- Scenario 1 : Directe koppeling tussen ORBA en de financiële software via API's
- Scenario 2a: Single Challenge Data warehouse project
- Scenario 2b: Toekomstgericht Data warehouse project

Met de bedoeling te bepalen in welke mate elk scenario de behoeften van de centrumsteden vervuld. Deze behoeften zijn:

- Onmiddellijke, punctuele uitwisseling van gegevens - Informatie uit ORBA is snel opvraagbaar in de financiële pakketten, met een "druk op de knop";
- Geïntegreerde rapportering:

- Voor een (beperkt) aantal standaardrapporten (bijv. de Digitale rapportering aan de Vlaamse overheid) wordt informatie uit ORBA gecombineerd met financiële informatie. De rapporten worden gebouwd via standaarden opgelegd door de hogere overheid.
- Rapportering op onderwerp-niveau:
 - Financiële informatie en informatie over de voortgang van een project / doelstelling / actieplan / actie (Status, Indicatoren, kwesties, ...) wordt samengebracht om een overzicht te geven voor één onderwerp.
- (Financiële) opvolging meerjarenplanning:
 - Door de status van projecten, acties, actieplannen en beleidsdoelstellingen te linken met de financiële informatie krijgen verschillende betrokkenen in de organisatie, op hun maat, een overzicht van essentiële project- en financiële informatie.

2.2. Plan van Aanpak

2.2.1. Overzichtsplan van aanpak

De verschillende fasen in het onderzoek waren:

- **Onderzoeksfase:** verwerven van zoveel mogelijk inzicht aan de hand van beschikbare informatie
- **Behoeftanalyse:** om de functionele en niet-functionele behoeften van de steden te kennen werd er een vragenlijst opgesteld.
- **Functionele workshop:** voorstel van de geconsolideerde resultaten van de behoeftanalyse tijdens een workshop met het Kenniscentrum en de centrumsteden
- **Technische analyse van de bronsystemen:** er werden verschillende afstemworkshops ingepland om zowel de financiële pakketten en ORBA beter te begrijpen
- **Afstemming met DWH team Antwerpen:** met als doel hun DWH oplossing beter te begrijpen
- **Technische workshop:** voorstel van een geconsolideerde technische analyse tijdens een workshop met het Kenniscentrum en de centrumsteden
- **Detailvoorstel:** op basis van voorgaande analyses en workshops werden de resultaten van dit haalbaarheidsonderzoek voorgesteld
- Kostenraming
- Oplevering haalbaarheidsonderzoek

2.2.2. Onderzoeksgebieden

De volgende gebieden werden in detail bekeken tijdens dit haalbaarheidsonderzoek:

- AS-IS:
 - Bestaande omgeving in de centrumsteden, zowel technisch als organisatorisch
 - Gebruik van de bronsystemen per stad
- Toekomstig DWH:
 - Functioneel:
 - Strategische projectopvolging
 - Wettelijke beleidsopvolging

- Dimensies en meetwaarden ORBA data
 - Dimensies en meetwaarden financiële data
 - Verrijking met andere bronnen
- Niet-functioneel
 - Type gebruikers en ondersteuning
 - Infrastructuurbehoeften
 - Veiligheidsverwachtingen
- Technisch
 - BI Technologie & software
 - Platform
 - DWH methodologie
 - Deelbaarheid
- Scenario 2a versus 2b: verschil in functionaliteit, technische oplossing en implementatie van beide scenario's verder uitwerken
 - **Deelbaarheid** van de oplossing analyseren
 - **Uitbreiding** van de oplossing analyseren
 - **Integratie** met/in bestaand DWH

3. BEHOEFTE ANALYSE

3.1. AS-IS

3.1.1. Omgeving per stad

Onderstaand overzicht geeft per stad de AS IS situatie weer. De verschillende kolommen geven weer of de stad al dan niet ORBA gebruikt, waar ORBA gehost is, het Identity management oplossing de stad heeft, welk financieel pakket, of dit “on-premise” of gehost wordt en tenslotte of ze reeds een data warehouse / SSIS (SQL Server Integration Services) gebruiken en of daar reeds financiële data in is opgenomen.

STAD	ORBA	Hosting	IM	Fin. Sys.	Hosting	IM	DWH	SSIS	Fin. Data in DWH
Aalst	✓	CPC	AD	Cipal – Mercurius	On-Prem	UID			
Brugge	✓	CPC	AD	Unit4 – UBW	On-Prem		✓	✓	✓
Genk	✓	CPC	ADFS	Cipal – Mercurius	Cipal SaaS	UID			
Gent	In opstart		AD	Digipolis – SAP	On-Prem		✓	✓	✓
Hasselt	✓	CPC	AD	Cipal – Mercurius	On-Prem	UID	✓	✓	*.CSV
Leuven	Opgestart	CPC	AD	Cipal – Mercurius	On-Prem	UID			
Mechelen	✓	CPC	AD	Cipal – Mercurius	On-Prem	UID	✓	✓	DB Views ?
Oostende	✓	CPC	AD	Cevi – New Horizon	On-Prem	AD			
Sint-Niklaas	✓	CPC	AD	Cipal – Mercurius	On-Prem	UID		✓	
Turnhout	✓	CPC	AD	Cipal – Mercurius	On-Prem	UID			

✓ beschikbaar
 ✓ deels beschikbaar
 CPC = Cronos Private Cloud

Tabel 1 – Overzicht AS IS situatie in de centrumsteden

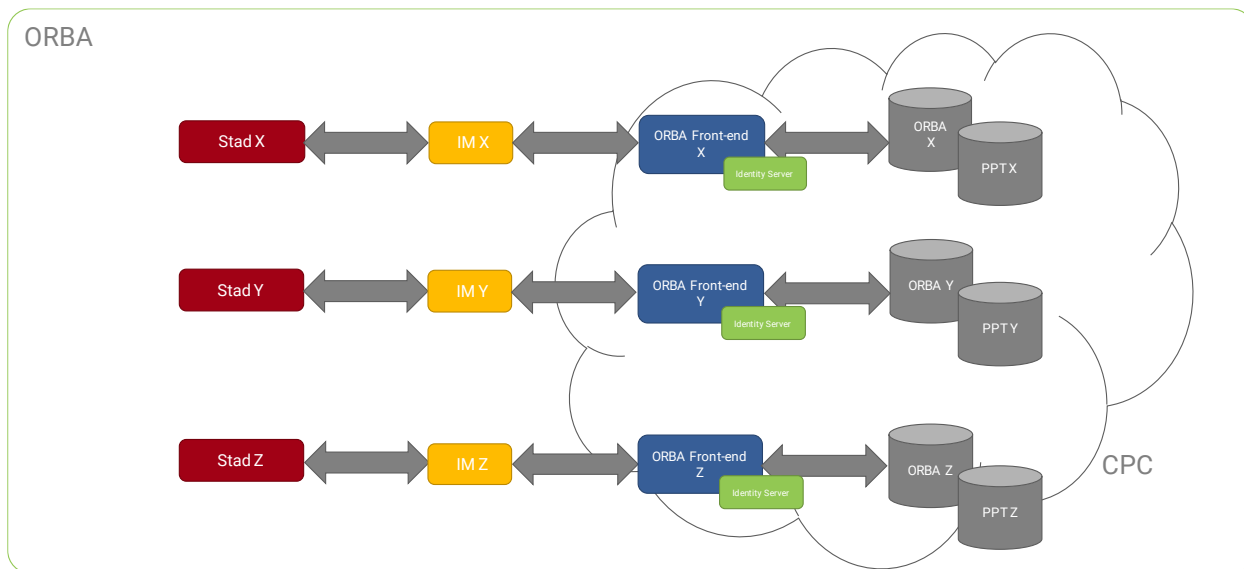
De steden betrokken in dit haalbaarheidsonderzoek kunnen opgedeeld worden volgens:

- ORBA reeds opgestart en in gebruik: de meeste steden hebben ORBA reeds in gebruik.
- Financieel pakket: er zijn 4 financiële pakketten aanwezig in de steden, waarvan Cipal-Mercurius het meeste verspreid is in de steden in het studiegebied
- Aanwezigheid van een DWH incl. of excl. financiële data

3.1.2. Opzet van de bronsystemen

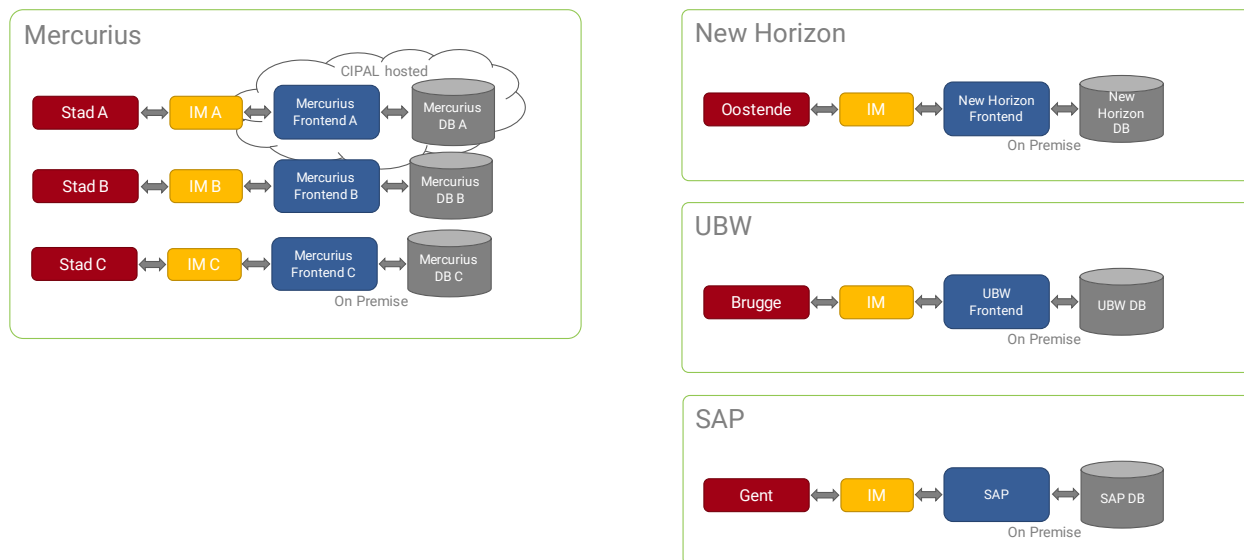
Onderstaande figuur geeft weer hoe de bronsystemen in de verschillende systemen is opgezet.

ORBA



Figuur 2 – Opzet bronsystemen - ORBA

FINANCIELE SYSTEMEN



Figuur 3 – Opzet bronsystemen – financiële systemen

3.2. Functionele behoeften

3.2.1. Overzicht informatie behoeften

Tijdens de behoefte analyse werd er gepolst naar de behoeften in de verschillende steden en werden deze in kaart gebracht en opgedeeld in Hoge / Medium / Lage behoefte.

3.2.1.1. Strategische projectopvolging

Behoefte	Hoog	Medium	Laag	Bijkomende info
Pure ORBA Projectopvolging				
Koppeling Financieel-ORBA op basis van doelstellingen – actieplan – acties				ORBA heeft meer planning details dan de (meeste) financiële systemen
Financiële projectopvolging om o.a. geschatte en werkelijke projectkost te kunnen opvolgen, restbudget te kunnen heralloceren, ...				- Link moet eerst mogelijk gemaakt worden in de Financiële toepassingen en steden moeten op die manier budgetteren - Niet voor alle steden
Gedetailleerde projectplannings- en mijlpalenopvolging				Nood voor enkele steden, maar beperkte gemeenschappelijke nood
Procesopvolging				Opvolging door het koppelen van reguliere werking aan indicatoren
Opvolging sleutelkwesities				Weinig steden volgen dit op
Historische rapportering en trend analyse				- Focus ligt vooral op actuele opvolging en meerjarenplannen - Maximale tijdspanne: legislatuur
Benchmarking van andere steden en binnen clusters				Momenteel niet prioritair, focus op lokaal doelstellingsbeleid

Tabel 2 – Overzicht behoeften met betrekking tot de strategische project opvolging

3.2.1.2. Wettelijke beleidsopvolging

Behoefte	Hoog	Medium	Laag	Bijkomende info
Financiële rapportering: Budgetten en effectieve uitgaven opvolgen van acties / actieplannen / doelstellingen per budgettype, per boekjaar,...				Koppelen van de financiële aspecten aan de inhoudelijke aspecten (doelstellingen en actieplannen) van het beleid, in de plannings- als in de rapporteringsfase
Lokale BBC Rapportering				- Nood aan eigen financiële analyse van MJP en jaarrekeningen lokaal, zowel per bestuursniveau als geconsolideerd - Uitbreiding op digitale rapportering en van de budgettaire boekhoudsleutel in sommige steden
Historische rapportering en trend analyse				Koppelen van de beleidsplanning op middellange termijn (6 jaar) met de korte termijn
Benchmarking van andere steden en binnen clusters				- Momenteel niet prioritair, focus op lokaal doelstellingsbeleid - Benchmark rapporten beschikbaar op Portaal Vlaamse Overheid

Tabel 3 – Overzicht behoeften met betrekking tot de wettelijke beleidsopvolging

3.2.2. Verrijking en analyse met andere data

Behoefte	Hoog	Medium	Laag	Bijkomende info
Behoefte om andere informatie te ontginnen door ontsluiten van andere bronsystemen: ● Personeelsdata ● Organogram ● Bevolkingsdata ● Demografische data ● Tijdsregistratie ● ...				● Koppelen van de financiële aspecten aan de inhoudelijke aspecten (doelstellingen en actieplannen) van het beleid, in de plannings- als in de rapporteringsfase

tabel 4 Overzicht behoeften met betrekking tot andere data

3.2.3. Type rapportering

Behoefte	Hoog	Medium	Laag	Bijkomende info
Uniforme en transparante rapporteringsomgeving				● Meer kwaliteit, meer uniformiteit, minder administratieve last
Standaard Rapporten				● Zowel voor lokale opvolging als verplichte rapportering
Ad-hoc rapporten				● Specifieke vragen dienen ten alle tijden opgevangen worden
Dashboards				● Visuele voorstelling van KPI's

Tabel 5 – Overzicht behoeften qua type rapportering

3.3. Mappen van de behoeften en opvolgingsniveaus per stad

Stad	ORBA Doelstelling – actieplan - Actie	ORBA Budgetteren op project	ORBA Mijlpalenopvolging	ORBA Procesopvolging	Financieel Doelstelling – actieplan - actie	Financieel Projectniveau	Financieel Uitgebreide fin rapportering
Aalst	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	● (nog) niet van toepassing	● (nog) niet van toepassing	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	● PowerBi dashboard aangekocht bij TriFinance
Brugge	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	● Eigen DWH incl fin data
Genk	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	opvolging gewenst	opvolging gewenst	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	gewenste opvolging
Gent	● opstart						● Eigen DWH incl fin data
Hasselt	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	opvolging gewenst	opvolging gewenst	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	gewenste opvolging
Leuven	niveau aanwezig opvolging nodig	● niveau aanwezig ● opvolging nodig	● niet duidelijk	● niet duidelijk	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	gewenste opvolging
Mechelen	niveau aanwezig opvolging nodig	● niveau aanwezig ● opvolging nodig	● niveau aanwezig ● opvolging nodig	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	niveau deels aanwezig opvolging gewenst	gewenste opvolging Beperkte info beschikbaar in DWH
Oostende	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	● niveau NIET aanwezig ● opvolging gewenst	● niveau NIET aanwezig ● geen opvolging gewenst	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	gewenste opvolging
Sint-Niklaas	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	● niet duidelijk	● niet duidelijk	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	● niet duidelijk
Turnhout	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	niveau aanwezig opvolging nodig	● (nog) niet van toepassing	niveau aanwezig opvolging nodig	● niveau NIET aanwezig ● opvolging gewenst	gewenste opvolging

Tabel 6 – Overzicht behoeften met betrekking tot de opvolgingsniveau per stad

3.4. Niet-functionele behoeften-analyse

3.4.1. Type gebruikers en ondersteuning

Behoefte	Hoog	Medium	Laag	Bijkomende info
Informatiegebruikers				● Grote getalen van gebruikers per stad: managers + medewerkers
Analisten				● Beleids- en financiën medewerkers
Publieke gebruikers				● Geen prioriteit voor ORBA en/of Financiële data
Lokale BI Team(s)				● Momenteel niet in de meeste steden aanwezig ● Kleine investeringen in de toekomst voorzien
Centrale ondersteuning voor gebruikers en DWH / Data model				● Te weinig lokale ondersteuning voor BI nu en later voorzien
Gebruiksvriendelijkheid				● Gezien het groot aantal informatiegebruikers dient de rapporteringsomgeving makkelijk te gebruiken zijn door de eindgebruikers, zonder dat er gespecialiseerde training nodig is

Tabel 7 – Overzicht behoeften met betrekking tot het type gebruikers en ondersteuning

3.4.2. BI Infrastructuur

Behoefte	Hoog	Medium	Laag	Bijkomende info
Gestandaardiseerde toegang				● De BI infrastructuur moet haar diensten ontsluiten m.b.v. standaardtechnologieën, bv. Windows AD
Beschikbaarheid				● De BI infrastructuur dient beschikbaar te zijn binnen uitgebreide kantooruren om de toegang tot de data en rapporteringsomgeving te garanderen. ● De beschikbaarheid moet uitbreidbaar zijn in bepaalde periodes, vastgelegd in een SLA.
Schaalbaarheid				● De architectuur moet openstaan voor integratie van nieuwe bronnen en moet in staat zijn een groeiende werklast op een efficiënte manier aan te kunnen. ● Geen prio nummer 1.

Tabel 8 – Overzicht van de behoeften met betrekking tot de BI infrastructuur

3.4.3. Veiligheid

Behoefte	Hoog	Medium	Laag	Bijkomende info
Beveiliging				● De (cloud)infrastructuur moet effectieve beveiligingsmechanismen bieden om gegevensbescherming, identiteitsbeheer en privacy te waarborgen
Gegevensbescherming				● Het BI systeem moet de te ontginnen gegevens en afgeleide kennis beschermen tegen ongeoorloofde toegang door: ● Systeembeveiliging (rollen en privileges) ● Objectbeveiliging ● Databeveiliging
Anonimisering / Pseudonimisering				● Enkel van belang in geval personeelsdata en bevolkingsdata in scope is

Tabel 9 – Overzicht van de behoeften met betrekking tot veiligheid

3.4.4. BI-Technologie

Behoefte	Hoog	Medium	Laag	Bijkomende info
Microsoft Technologie				● Microsoft technologie wijdverspreide standaard in de steden ● SQL Server ● SSIS ● SSAS ● PowerBI
Onafhankelijke Front-End infrastructuur				● Andere Front-end tools moeten aan de backend infrastructuur kunnen gekoppeld worden

Tabel 10 – Overzicht van de behoeften met betrekking tot de BI technologie

4. TECHNISCHE HAALBAARHEID

4.1. Definitie van de vooropgestelde alternatieve scenario's

4.1.1. Scenario 1 : Directe koppeling tussen ORBA en de financiële software via API's

Volgens het bestek heeft de vraag van scenario 1 het volgende uitgangspunt :

- Directe koppeling tussen ORBA en de financiële software via API's – Via zogenaamde API's wordt informatie uit ORBA op gestandaardiseerde wijze ontsloten. Leveranciers van financiële softwarepakketten kunnen een rechtstreekse koppeling bouwen naar deze API's om informatie met ORBA uit te wisselen.
- Onmiddellijke, punctuele uitwisseling van gegevens – Informatie uit ORBA is snel opvraagbaar in de financiële pakketten, met een “druk op de knop”.

Als representatief financieel pakket werd Mercurius van Cibal/Schaubroeck aangeduid, met als deskundige Gert Steegmans van de Stad Hasselt welke vroeger voor Cibal heeft gewerkt.

Op woensdag 13/05 hebben we een conference call gehad met als deelnemers Gert Steegmans (Hasselt), Annelies Fiers (Cronos), Sylvie Baes (Cronos), Inge Marchal (Hasselt) en Kim Huysmans (Cronos) over de koppeling ORBA - Mercurius.

Uit dit gesprek bleek al snel dat er niet echt een behoefte was om data van ORBA op te vragen of te visualiseren in Mercurius, maar eerder een omgekeerde behoefte om financiële gegevens van Mercurius opvraagbaar te maken in ORBA.

In Mercurius worden de vastleggingen en toekenningen gedaan van de budgetten op Beleidsdoelstelling – Actieplan – Actie niveau. Deze niveaus kunnen uit Mercurius worden geëxporteerd door een op maat gemaakte export, en worden dan in ORBA opgeladen. In ORBA worden aan deze doelstellingen projecten gekoppeld. Op een project kan nu een “geschatte kost” aangegeven worden, maar dit is een vrij veld. De nood is dus om de echte financiële gegevens in Mercurius te kunnen koppelen aan de Projectplanning gegevens in ORBA.

Om deze koppeling te kunnen automatiseren is hetzelfde niveau van detail nodig in zowel Mercurius als in ORBA. Dat wil zeggen dat moet kunnen aangegeven worden in Mercurius aan welke (ORBA) projecten de financiële gegevens gekoppeld kunnen worden.

Bij verdere bevraging en analyse is gebleken dat deze koppeling momenteel nog niet mogelijk is in Mercurius. Er zal dus een wijziging noodzakelijk zijn in Mercurius zodat deze koppeling automatisch kan gelegd worden.

Daar de behoefte er is om de financiële gegevens te visualiseren/rapporteren in ORBA, en niet hetzelfde financiële pakket aanwezig is in alle deelnemende steden, heeft dit een aantal consequenties voor ORBA:

- ORBA moet dan met verschillende pakketten kunnen communiceren waardoor er geen eenduidige Inbound API kan gebouwd worden. Het zal eerder ORBA zijn die bij de verschillende financiële pakketten de gegevens moet gaan halen, en daarom zal er met elk pakket afspraken moeten gemaakt worden.
- Inbound vs Outbound ORBA API
- Inbound API maakt het mogelijk om externe (financiële) data op te laden in ORBA (Scenario 1)
- Outbound API maakt het mogelijk om ORBA data te extraheren (Scenario 2a en 2b)
- Aangezien ORBA gelijk is voor alle deelnemende steden, en momenteel niet voorzien is voor uitgebreide rapportering zal:
- Er moeten vast gelegd worden welke rapportering er moet worden gebouwd. Er zal dus een consensus moeten gevonden worden.
- Aangezien deze rapporten enkel door IT kan worden gebouwd zal dit een financiële impact hebben op ORBA. Ook wijzigingen of toevoegingen aan de rapportage kan enkel uitgevoerd worden door IT.
- De ORBA code van de centrumsteden zal verder afwijken van de versie van Antwerpen waardoor de herbruikbaarheid van nieuwe functionaliteiten beperkt zal worden

Al deze punten in acht genomen is het ons inziens dat het opzetten van een datawarehouse (dus scenario 2a of 2b) een betere oplossing is om aan deze rapporteringsvereisten te voldoen.

Echter de behoefte om in Mercurius (of een ander financieel) pakket de koppeling te kunnen maken tussen ORBA projecten, en de financiële gegevens blijft in deze scenario's ook overeind.

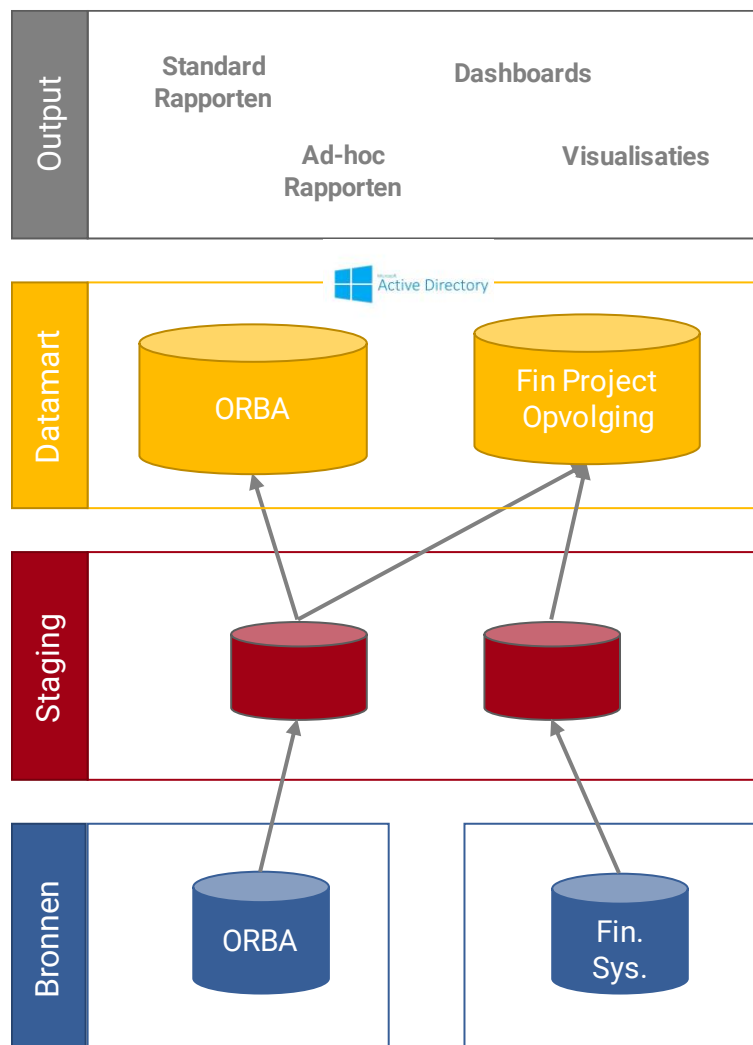
4.1.2. Scenario 2a – “Single Purpose” data warehouse project

Onder “single purpose” verstaan we dat binnen dit scenario het enige doel erin bestaat om data uit ORBA te combineren met de gegevens uit financiële software pakketten om enkel en alleen aan de daaruit voortvloeiende behoeften van de centrumsteden te voldoen.

Dit betekent echter wel dat uitgebreide en/of lokale financiële rapportering niet mogelijk is door de beperkte ontsluiting van de financiële gegevens.

Tenzij de verschillende leveranciers van de fin software pakketten dezelfde interface kunnen aanbieden, is deze oplossing vooral deelbaar voor de steden met eenzelfde financieel pakket. (dit wordt scenario 2a') genoemd.

Scenario 2a is een snelle oplossing waar specifieke data marts worden gebouwd en waarbij niet alle lagen van de referentie architectuur opgezet worden. Er wordt dus bijgevolg geen integratie laag gecreëerd.



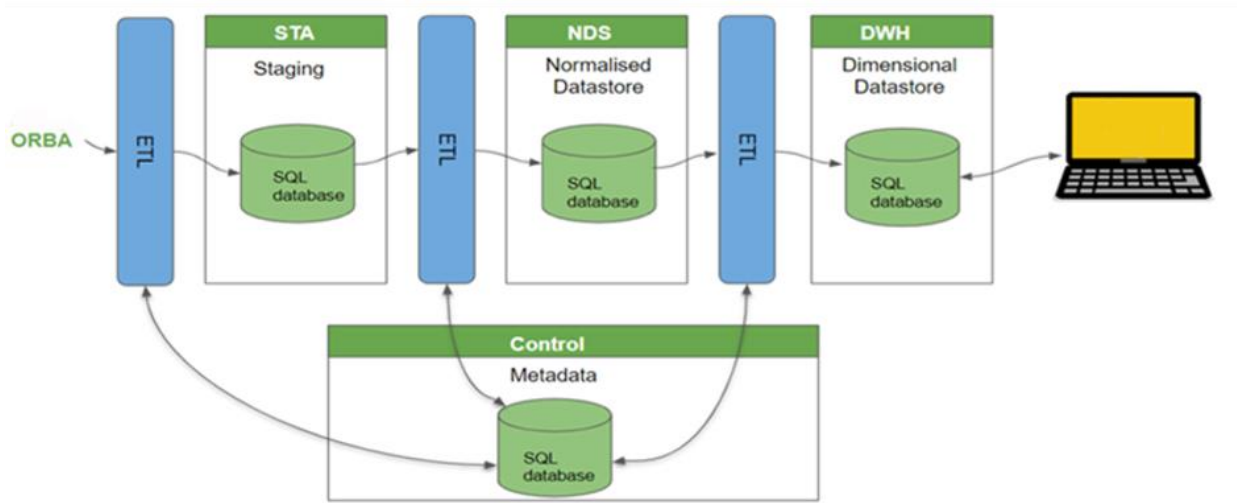
Figuur 4 – Voorstelling scenario 2a

4.1.3. Scenario 2b – Toekomstgericht datawarehouse project

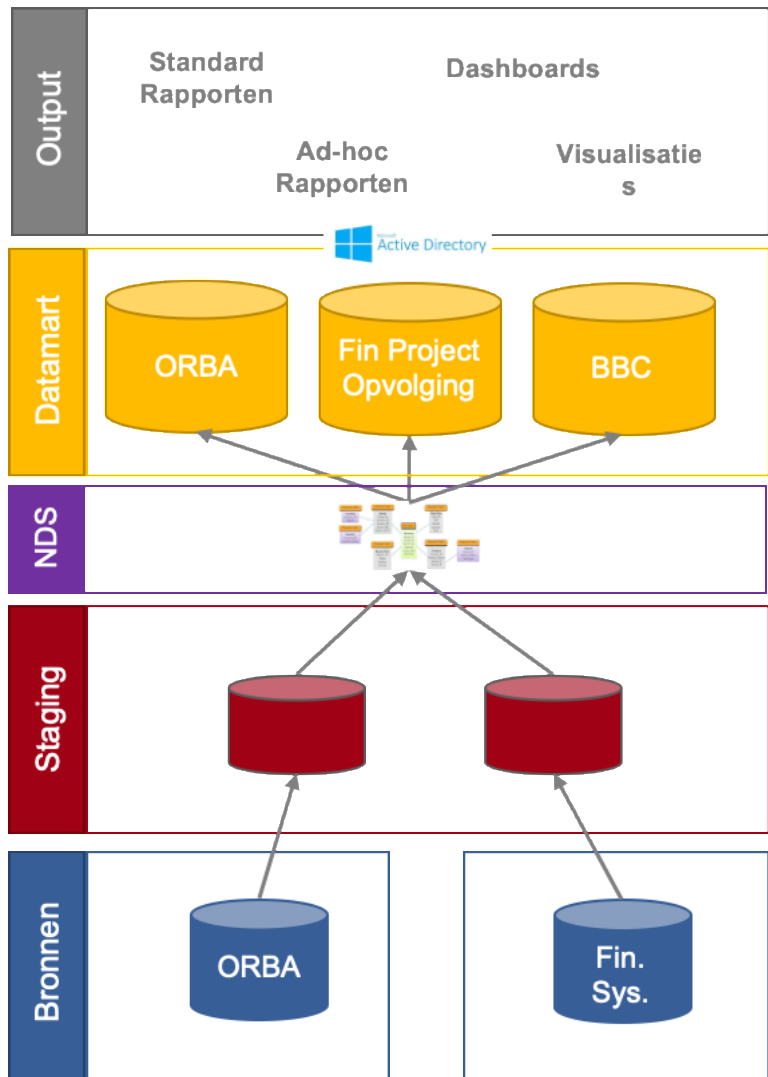
Scenario 2b is een oplossing gebouwd volgens duurzame datawarehouse principes die niet alleen toelaat om te voldoen aan de behoefte met betrekking tot beleidsopvolging maar ook toelaat andere financieel relevante analyses te doen, alsook op termijn bijkomende informatie of databronnen toe te voegen.

Een duurzaam datawarehouse wordt opgezet met meerdere lagen zoals de vooropgestelde referentiearchitectuur volgen B. Inmon, reeds aanwezig in Brugge. Het is ook mogelijk om andere methodes te gebruiken om een duurzaam DWH op te zetten, zoals de methodologie van R. Kimball (aanwezig in Antwerpen).

In tegenstelling tot de Kimball methodologie wordt in de Inmon methodologie een integratie laag gebouwd. Die ontkoppelt de bron- en publicatie laag waardoor impact van wijzigingen veel kleiner wordt en zorgt ervoor dat de oplossing deelbaar is door verschillende steden met verschillende financiële pakketten. Om deze redenen verkiezen we inderdaad voor deze implementatie de Inmon methodologie.



Figuur 5 – Referentie Architectuur (B. Inmon)



Figuur 6 - Voorstelling scenario 2b

4.1.4. Conclusie Scenario 2a versus 2b

Scenario 2a is mogelijk maar is eerder een kortetermijnoplossing omdat telkens er wijzigingen nodig zijn een volledige aanpassing van de code nodig is.

Door een intrinsiek lagere deelbaarheid van 2a is de gemiddelde kost per centrumstad hoger dan voor scenario 2b.

Scenario 2a zal snel naar 2b moeten evolueren bij de minste extra noden anders zullen de kosten van de diverse puntoplossingen snel deze van de structurele 2b oplossing overstijgen.

Indien er geen verdere samenwerking tussen de centrumsteden zal zijn dan voor 'ORBA + financiële data' dan is de oplossing 2a een alternatief dat kan gebruikt worden. Maar zoals eerder aangegeven, enkel voor de korte termijn en enkel indien er geen wijziging komt op de aangeleverde data structuur. Elke wijziging is een aanpassing van de volledige code.

In alle andere gevallen is 2b de betere lange termijn oplossing

Elke stad kan verder eigen SILO's toevoegen zonder rekening te moeten houden met wat er in de andere SILO's eerder werd gebouwd.

Scenario 2a is mogelijk maar is eerder een korte termijn oplossing met weinig visie en potentieel voor verdere gezamenlijke uitbreiding hogere kost op langere termijn. De gemiddelde kost voor alle steden zou ook hoger kunnen liggen door de intrinsieke deelbaarheidsbeperking.

4.2. Deelbaarheid van de oplossing

4.2.1. Schaalvoordeel tegenover individualisatie graad

De realisatie van een data warehouse project, volgens scenario 2a of 2b, kunnen de centrumsteden gezamenlijk aanpakken waarbij zowel de ontwikkeling en het onderhoud van de oplossing, als de infrastructuur of de omgeving kunnen gedeeld worden.

Het volgend overzicht geeft de voor – en nadelen van het delen van de ontwikkeling en het onderhoud van de oplossing.

Eigen ontwikkeling

- Strategische keuzes gebeurt op niveau van de centrumstad
- Functioneel autonome keuzes per centrumstad
- Volledig individualiseerbaar
- Implementatiesnelheid begrensd door centrumstad
- Software keuzes zijn individueel
- Eigen ontwikkelings- en onderhoud team nodig
- Ontwikkelings- en onderhoudskosten niet of beperkt deelbaar
- Eigen authenticatie oplossing is mogelijk
- Beperkt de kruisbestuiving (Kennis & inzichten)

Gedeelde ontwikkeling

- Strategische keuzes gebeurt op niveau van de centrumsteden samen
- Functionele consensus tussen de centrumsteden nodig
- Gedeeltelijk individualiseerbaar
- Implementatiesnelheid begrensd door gemeenschap
- Software keuzes zijn gemeenschappelijk (excl. Rapporteringstool)
- Geen eigen ontwikkelings- en onderhoud team nodig
- Ontwikkeling – en onderhoudskosten 100% deelbaar.
- Eigen authenticatie oplossing is mogelijk
- Maximaliseert de kruisbestuiving (Kennis & inzichten)
- Eigen rapporten kunnen nog steeds bovenop de gedeelde code ontwikkeld worden

Een gedeelde oplossing kan ofwel op een eigen omgeving in de stad geïnstalleerd worden of ter beschikking gesteld worden via een gedeelde omgeving. Ook deze keuze heeft een aantal voor- en nadelen die hieronder weergegeven worden in volgend overzicht.

Eigen omgeving

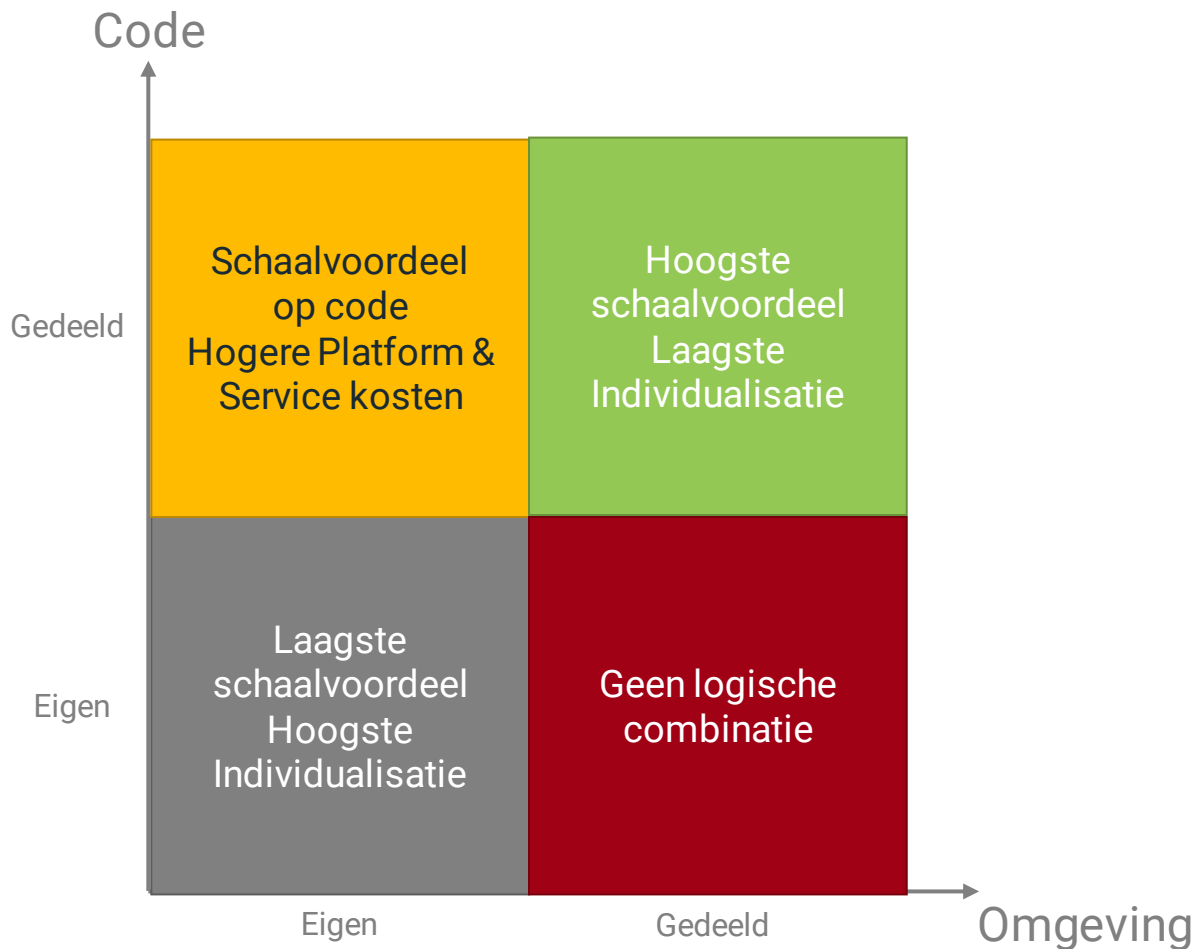
- Strategische keuzes gebeurt op niveau van de centrumstad
- Autonome platform keuzes per centrumstad
- Platform keuzes zijn individueel
- Eigen platform team nodig en ondersteuning tijdens uitgebreide kantooruren
- Platformkosten zijn niet deelbaar

Gedeelde omgeving

- Strategische keuzes gebeurt op niveau van de centrumsteden samen
- Platform consensus tussen de centrumsteden nodig
- Platform keuzes zijn gemeenschappelijk
- Geen eigen platform team nodig en ondersteuning tijdens uitgebreide kantooruren
- Platformkosten zijn voor een groot stuk deelbaar

Samenvattend komt het erop neer dat de kosten van ontwikkeling, onderhoud, platform en ondersteuning en de kennis en inzichten maximaal kunnen gedeeld worden maar men rekening dient te houden met de andere centrumsteden qua keuzes, verdere uitbreidbaarheid en timing van het project.

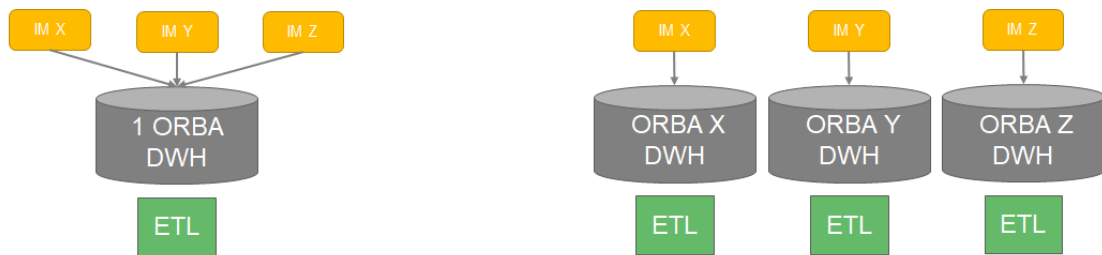
Dit schaalvoordeel t.o.v. de individualisatie graad wordt in onderstaande figuur tegenover elkaar weergegeven.



Figuur 7 – Schaalvoordeel t.o.v. individualisatie graad

4.2.2. Enkelvoudige of gescheiden data warehouse

Vervolgens dient ook een keuze gemaakt te worden over de manier waarop een gedeelde data warehouse omgeving zal worden opgezet. Er kan één oplossing worden gebouwd waarbij de data van alle centrumsteden in één data warehouse wordt ondergebracht en via security wordt aan elke stad toegang verleent tot de eigen data en rapportering. Op een gedeelde omgeving kan ook een aparte instantie van het data warehouse (maar ook de bijhorende ETL en analyse omgeving) gebouwd en ondersteund. In onderstaande figuur worden deze beide mogelijkheden schematisch weergegeven en de voor - en nadelen opgelijst.



- Makkelijker om benchmarking tussen steden toe te laten (maar niet prioritair)
- Complexer wat de IM koppeling betreft. Mogelijk extra technologie nodig
- “ROW-based” security nodig
- Enkel mogelijk bij gedeelde omgeving & gedeelde code
- Licentie kosten worden voor een stuk deelbaar
- Databank structuur moet identiek bij gedeelde code
- Eenvoudiger wat de IM koppeling betreft.
- Segregatie security zit in de oplossing
- Noodzakelijk bij gedeelde code of eigen omgeving
- Eventuele customisaties eenvoudiger
- Licentie kosten mogelijk niet deelbaar

De eerste mogelijkheid geeft nog een grotere deelbaarheid van de kosten maar ook een grotere afhankelijkheid (veiligheid, individualisaties, ...).

De eerste mogelijkheid is uiteraard enkel van toepassing indien men op een gedeelde omgeving werkt en de tweede mogelijkheid is de facto het geval indien men kiest voor een gedeelde code maar op de eigen omgeving. Elke instantie van de oplossing wordt dan op de eigen infrastructuur gehost.

4.2.3. Private of public Cloud oplossing

Bij een gedeelde omgeving kan men voor een publiek Cloud oplossing kiezen of voor een private Cloud oplossing.

De voordelen van een Private Cloud oplossing zijn:

- Exacte definitie van waar de data zich bevindt
- Duidelijk juridisch kader in België (Als private Cloud in België staat)
- “Gevoelsmatig” mogelijk veiliger gepercipieerd

De voordelen van een Publieke Cloud oplossing zijn:

- “Technisch” zeer sterke veiligheidsuitbouw t.g.v. enorme schaalvoordelen
- Elastisch
- Nieuwste technologieën beschikbaar en eenvoudig combineerbaar met andere technologie componenten (enorm breed spectrum aan technologie componenten)

Beide Cloud oplossingen zijn goede alternatieven en de keuze zal bepaald dienen te worden door berekening van de kosten en gevoelsmatige voorkeur van de centrumsteden.

4.3. Uitbreidbaarheid van de oplossing

4.3.1. Uitbreidbaarheid van de gezamenlijke scenario 2a oplossing versus scenario 2b oplossing

Indien er geen verdere samenwerking tussen de centrumsteden zal zijn dan voor 'ORBA + financiële data' dan is de oplossing 2a een alternatief dat kan gebruikt worden. Maar zoals eerder aangegeven, enkel voor de korte termijn en enkel indien er geen wijziging komt op de aangeleverde data structuur. Elke wijziging is een aanpassing van de code

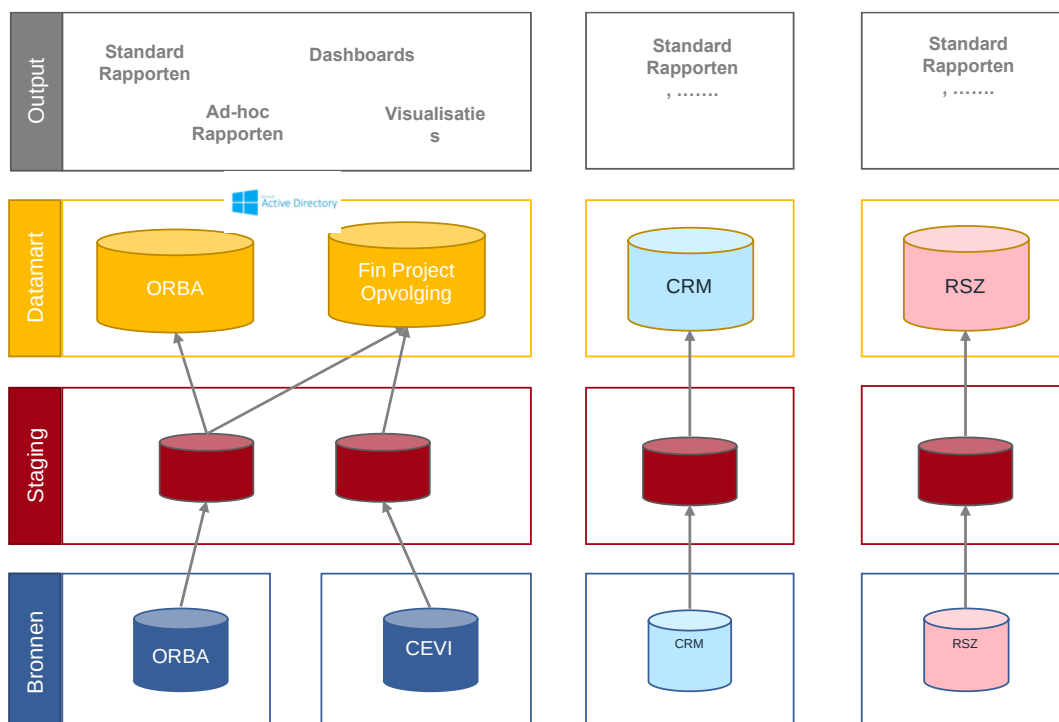
In alle andere gevallen is 2b de betere lange termijn oplossing

Binnen 2a kunnen we een SILO aanmaken voor 'ORBA + CEVI' die door alle steden die met CEVI werken kan gebruikt worden. Voor aanpassingen binnen deze SILO volstaat het om een akkoord te hebben met de steden die gebruik maken van deze SILO

Verder maken we ook een SILO aan voor 'ORBA+ Mercurius' die door alle steden die met Mercurius werken kan gebruikt worden. Voor aanpassingen binnen deze SILO volstaat het om een akkoord te hebben met de steden die gebruik maken van deze SILO

Elke stad kan verder eigen SILO's toevoegen zonder rekening te moeten houden met wat er in de andere SILO's eerder werd gebouwd.

De volgende figuur geeft dit schematisch weer:



Figuur 8 – Voorstelling uitbreidbaarheid scenario 2a

4.3.2. Draagvlak voor verdere gezamenlijke uitbreiding van de ORBA / financiële data oplossing

De vraag naar een draagvlak voor een verdere gezamenlijke uitbreiding wil eigenlijk zeggen of de verschillende centrumsteden eenzelfde eindresultaat willen hebben wat betreft de rapporten en de analyse mogelijkheden. Dit wil niet noodzakelijk zeggen dat men dezelfde bron applicatie/software dient te gebruiken.

Bijvoorbeeld: 5 centrumsteden willen kunnen rapporteren op hun personeelsgegevens volgens leeftijd, geslacht, graad, statuut maar hebben niet dezelfde personeelsapplicatie of bron hiervoor. Dan kan je in de oplossing 2b op een eenvoudige manier 1 rapport maken voor de 5 steden en onderaan in de structuur 5 verschillende bronnen het data warehouse met deze data te voeden.

Als nadien een 6de stad nood heeft aan dit rapport, dan kan zonder probleem hun bron toegevoegd worden en kunnen zij ook dat deel van het data warehouse en de bijhorende rapportering en analyse omgeving gebruiken mits betaling van hun deel van de investering (zelfde principe als in ORBA vandaag)

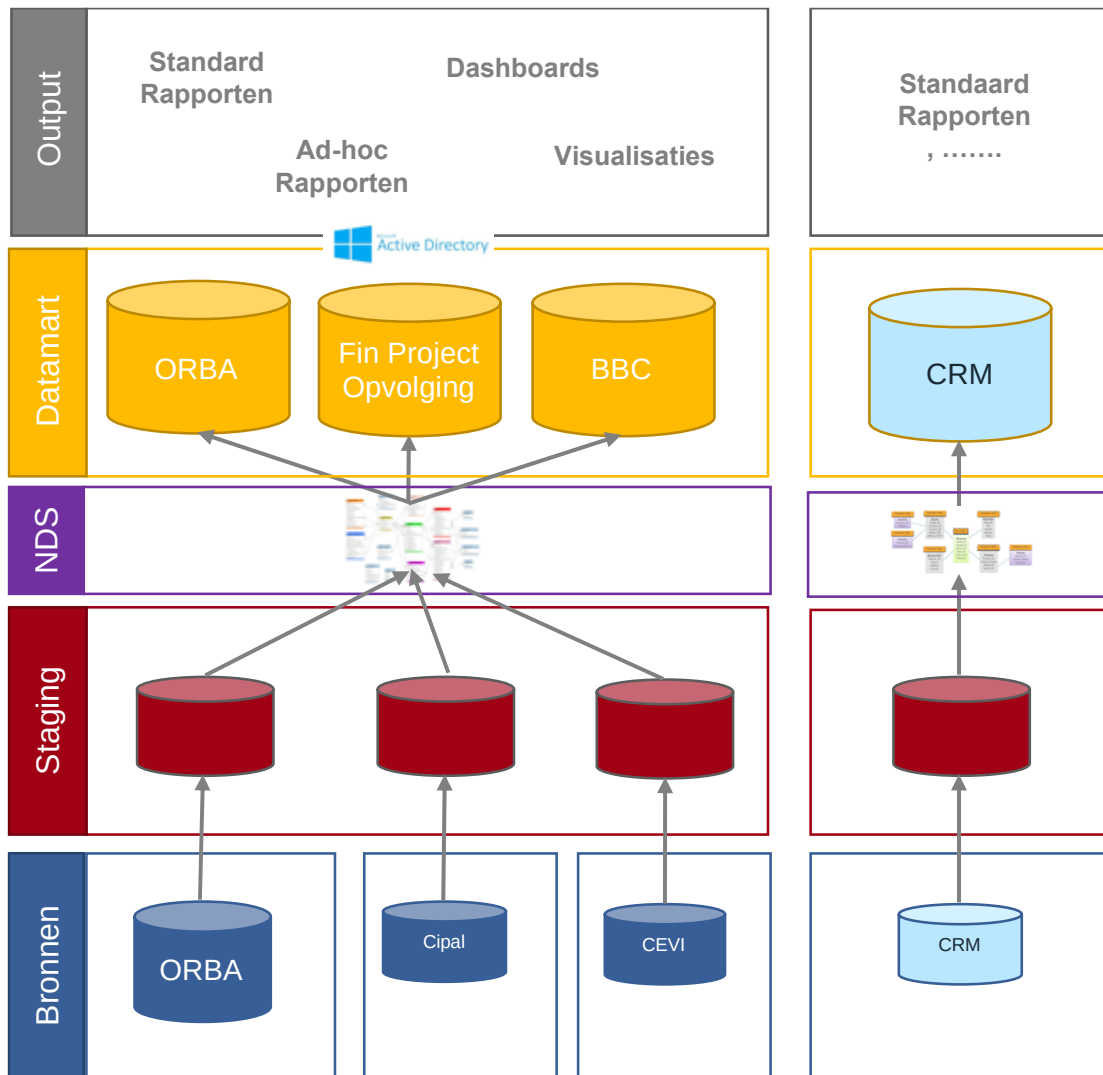
Hetzelfde principe is voorzien voor de koppeling 'ORBA + Financieel' pakket in de 2b oplossing, waar op basis van verschillende bronnen (New Horizon, Mercurius, etc.) dezelfde rapporterings en analyse omgeving wordt opgeleverd.

4.3.3. Mogelijkheden tot uitbreiding van de Scenario 2b oplossing

Een mogelijke uitbreiding van scenario 2b kan gezamenlijk of individueel per stad gebeuren.

Individuele uitbreiding zonder integratie

Onderstaande figuur geeft schematisch weer hoe een bepaalde stad een uitbreiding kan toevoegen aan zijn data warehouse waarbij er geen integratie nodig is met de gemeenschappelijke delen. In dit geval wordt een stuk van de oplossing in dezelfde omgeving naast de gemeenschappelijke oplossing bijgebouwd.



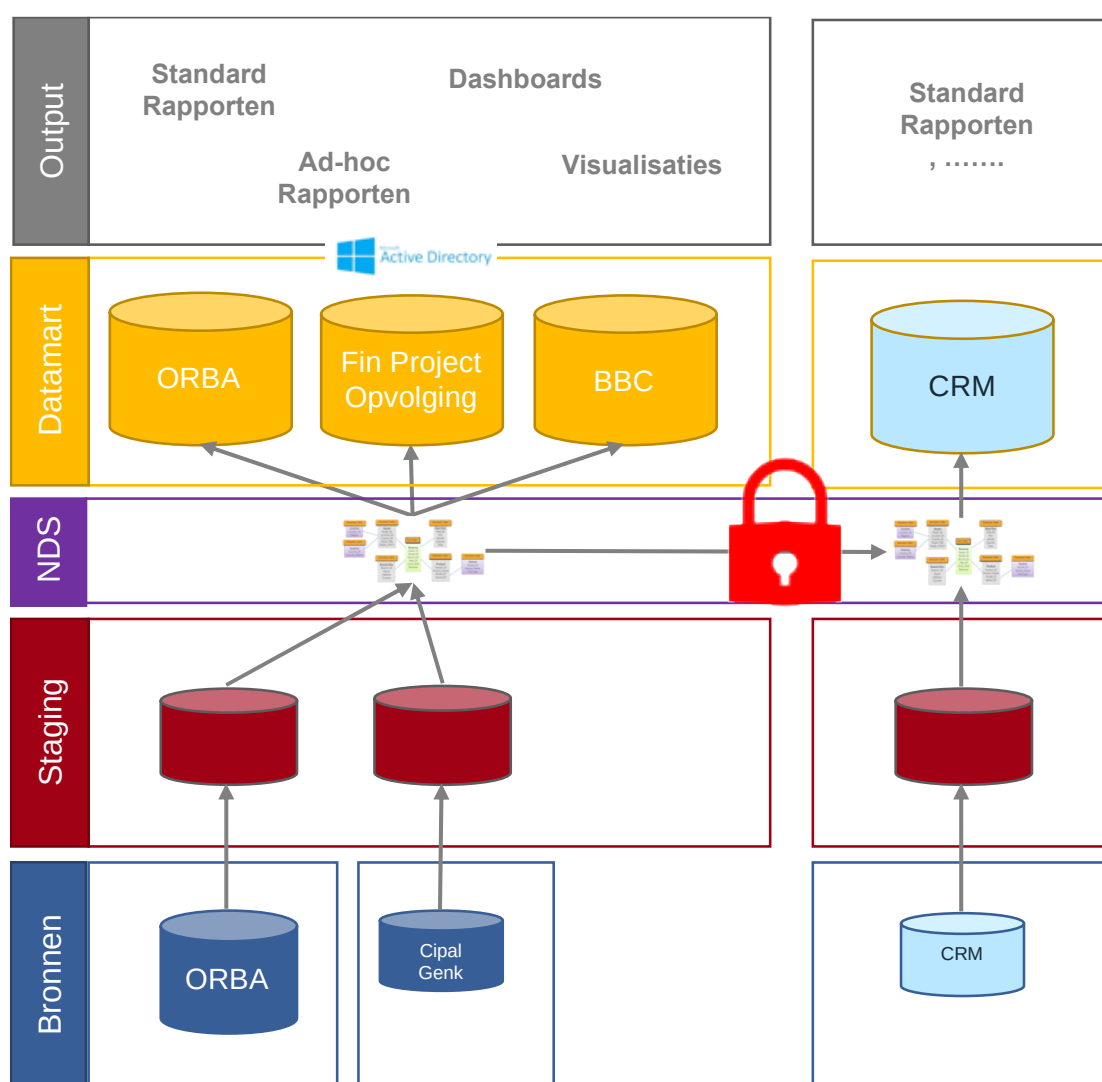
Figuur 9 -Voorstelling individuele uitbreiding scenario 2b zonder integratie

Individuele uitbreiding met een gemeenschappelijk luik

Het kan gebeuren dat deze individuele uitbreiding een gemeenschappelijk deel heeft (bijv. Gemeenschappelijke dimensies zoals tijd, personeel, organisatie, etc.) met de reeds bestaande gezamenlijke oplossing. In dit geval zijn een soort van 'rules of engagement' nodig.

- De gezamenlijke oplossing bepaald dan hoe de structuur er zal uitzien van de aangeleverde data, deze kan niet veranderd worden.
- Indien centrumstad die de info nodig heeft een wijziging wil aanbrengen aan het gezamenlijke deel, dan zal er
 - Toestemming nodig zijn van alle centrumsteden
 - En zal de beheerder van de 'ORBA + Financieel' pakket de verandering moeten doorvoeren

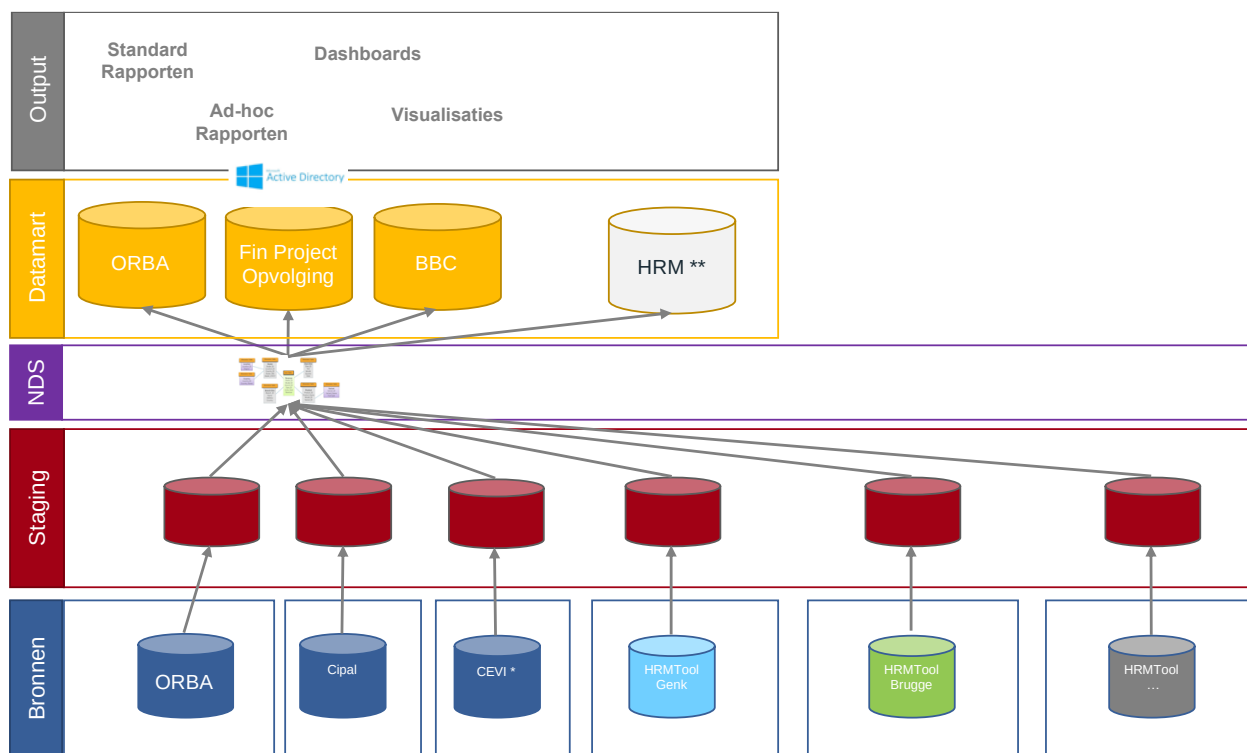
Onderstaande figuur stelt deze situatie schematisch voor:



Figuur 10 – Voorstelling individuele uitbreiding scenario 2 b met integratie

Gezamenlijke uitbreiding

Bij verdere gezamenlijke uitbreidingen van het gemeenschappelijk data warehouse is de kans groot dat de verschillende betrokken centrumsteden niet dezelfde bron hanteren voor deze data. Onderstaande figuur laat zien dat deze bronnen via de staging area en de integratie laag kunnen geïntegreerd worden om een gezamenlijke analyse en rapporterings omgeving voor alle centrumsteden te voeden.



Figuur 11 – Voorstelling gezamenlijke uitbreiding scenario 2b

4.4. Integratie van de oplossing met een bestaand data warehouse

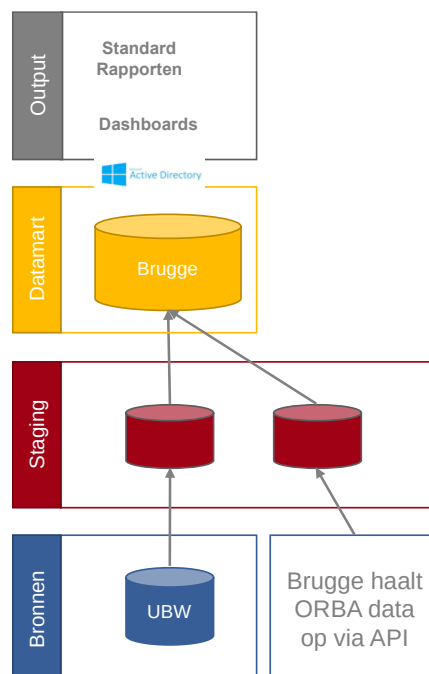
Bepaalde centrumsteden hebben reeds een bestaand en werkend data warehouse. Indien er door de andere centrumsteden beslist wordt om gezamenlijk een oplossing voor ORBA en financiële data te bouwen hebben zij 3 opties.

4.4.1. Optie 1:

De centrum stad die reeds een data warehouse heeft beslist om ORBA data / financiële data in het eigen data warehouse via eigen ontwikkeling te integreren.

- Dit kan een valabele optie zijn indien bijv. Enkel nog ORBA data dient geïntegreerd te worden, gezien de beperkte inspanning
- Snellere timing mogelijk dan via gedeelde ontwikkeling
- Geen gedeelde ontwikkelingskosten maar men kan eventueel wel ervaringen met de andere centrumsteden delen.

In onderstaande figuur wordt deze optie 1 schematisch voorgesteld.



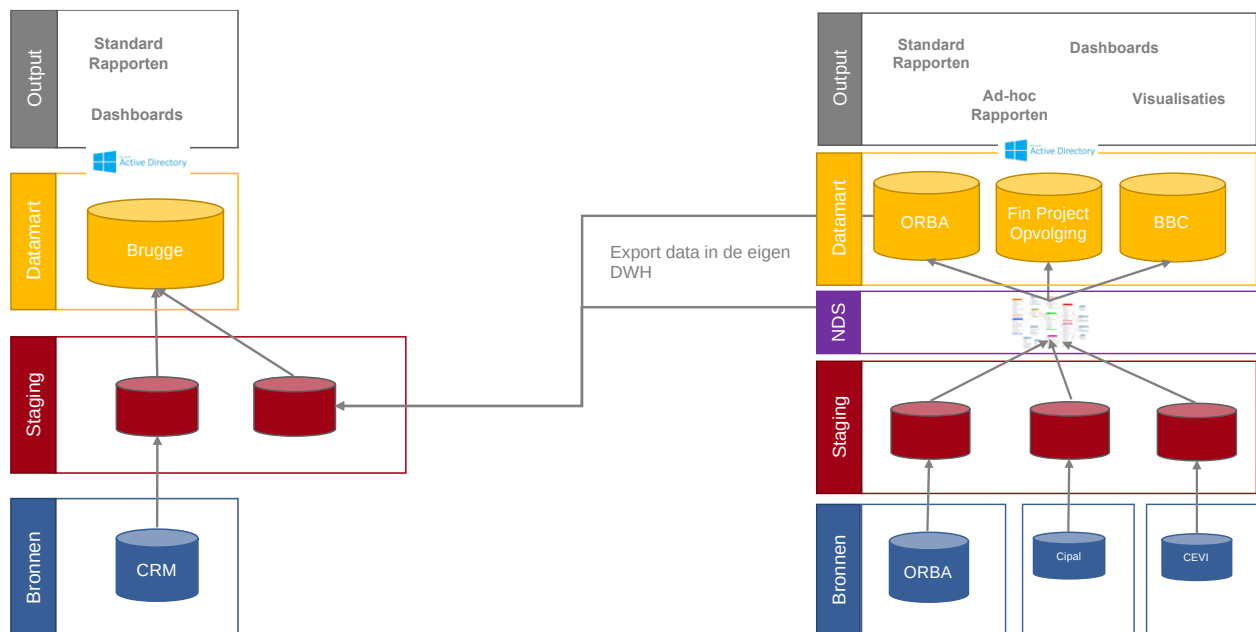
Figuur 12 – Overzicht optie 1 bestaand data warehouse

4.4.2. Optie 2:

De centrum stad die reeds een data warehouse heeft verkiest om in te stappen in de gedeelde ORBA / financiële data en integreert deze oplossing verder met zijn eigen data warehouse.

- Delen van kosten
- Integratie met de data uit het eigen data warehouse via het opladen van een extract uit de gedeelde oplossing en bridge tabellen

In onderstaande figuur wordt deze optie 2 schematisch voorgesteld.



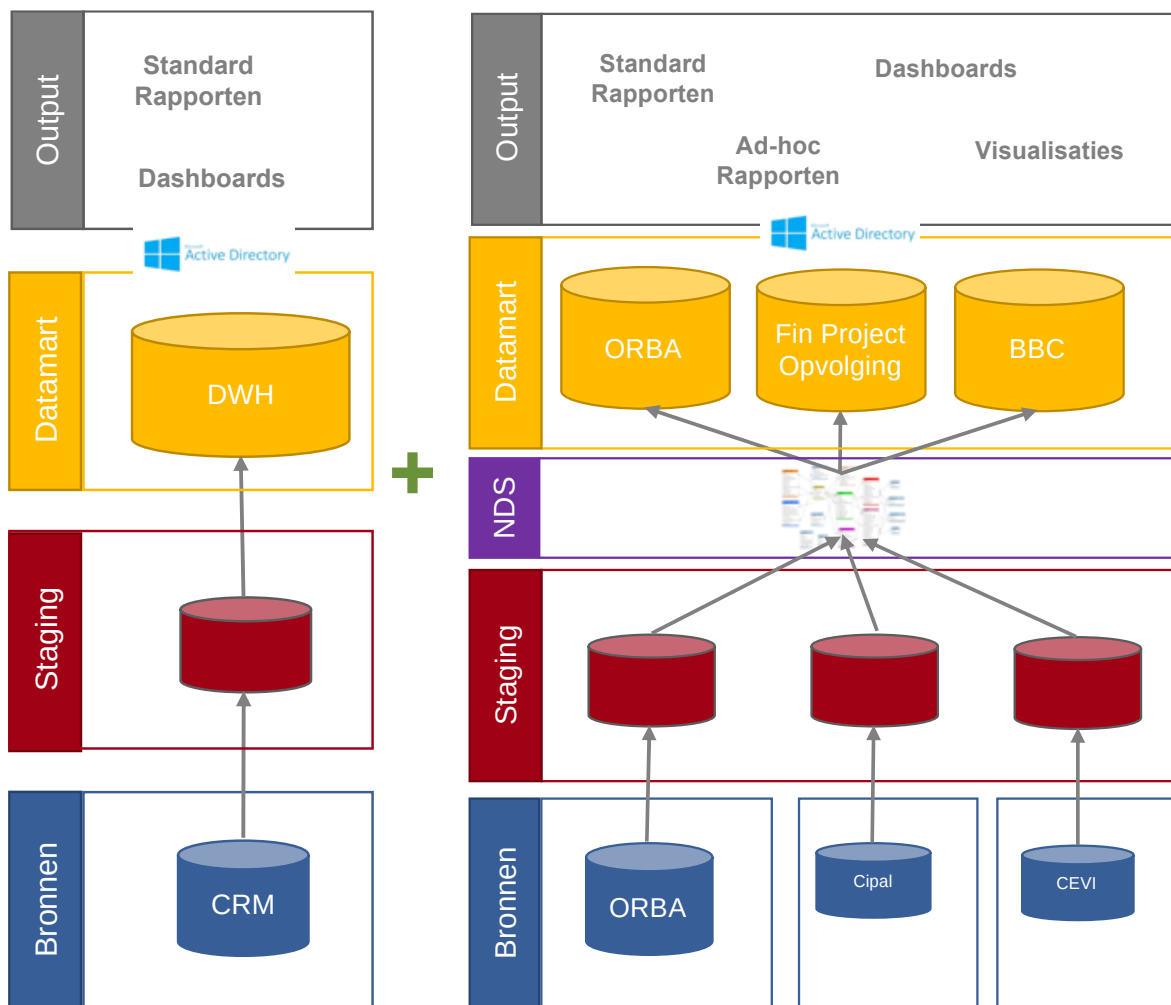
Figuur 13 – Overzicht optie 2 bestaand data warehouse

4.4.3. Optie 3:

De centrum stad die reeds een data warehouse heeft verkiest om in te stappen in de gedeelde ORBA / financiële data en integreert deze oplossing niet met zijn eigen data warehouse maar gebruikt ze als een aparte oplossing.

- Delen van kosten
- Dit is een valable optie indien men geen integratie nodig is met eigen data warehouse omgeving
- ORBA-financiële data oplossing wordt als een “stand-alone” oplossing beschouwd

In onderstaande figuur wordt deze optie 3 schematisch voorgesteld.



Figuur 14 – Overzicht optie 3 bestaand data warehouse

4.5. Bronextracties

Voor de realisatie van het Data warehouse ORBA / financiële data dient data te worden ge-extraheerd uit de ORBA/PPT en de financiële pakketten zoals Mercurius CIPAL, New Horizon CEVI,

Voor elk van deze bronsystemen dient de interface bepaald te worden.

- Deze kan zowel volgens het Push of Pull principe werken. Een push mechanisme heeft het voordeel dat de leverancier merkt dat interface niet meer werkt, maar in beide gevallen is het belangrijk dat de leverancier de verantwoordelijkheid neemt om de interface bij verdere releases mee te onderhouden en compatibiliteit garandeert.
- Interface kan van verschillende types zijn database query, .csv via FTP server, API, Een database query is enkel geschikt indien Bron en Data warehouse in dezelfde infrastructuur draaien.
- Alle afspraken met de leverancier dienen vast gelegd te worden in een SLA (inhoud, layout, frequentie, communicatie proces en timing bij wijzigingen, incident management proces, etc.)
- De data kan in een Full load of incrementeel geladen worden. Een increment bevat de nieuwe records, gewijzigde records en gedelete records. De volumes laten indien nodig wel toe om op niveau van het data warehouse de incrementen te bepalen.
- Er wordt vanuit gegaan dat de data dagelijks geladen wordt.
- De Financiële opvolging van de projecten vereist dat de financiële informatie (budget, vastleggingen, bestedingen, etc.) op niveau van projecten wordt bijgehouden (met n-n relatie tussen actie en project). Hiervoor dienen de financiële pakketten te worden aangepast. Daarnaast zal ook een change management project dienen te worden opgestart om budgettering op project niveau toe te laten.

4.6. Outbound API

4.6.1. Gekozen technologie

In al de scenario's die vervat zitten in dit haalbaarheidsonderzoek is communicatie met ORBA noodzakelijk. Er zal dus telkens een API moeten worden gebouwd, enkel zal het doel en dus ook de functionaliteiten ervan verschillen.

Om een goede technologie keuze te kunnen maken is het belangrijk om eerst vast te stellen wat juist de verwachtingen zijn van de API en hoe deze zal gebruikt worden.

Nadat deze verwachtingen zijn vastgelegd kunnen we deze vergelijken met de sterktes en zwaktes van de verschillende technologieën om zo de juiste keuze te maken.

- Verwachtingen van de API
 - Aangezien scenario 1 door ons wordt afgeraden en er nog niet eenduidig kan gedefinieerd worden wat de verwachtingen zouden zijn van de API, is het huidige en momenteel enige doel van de API het voeden van de datawarehouse. Een datawarehouse is tussen 2 data loads geïnteresseerd in de data die intussen is gewijzigd. Meer specifiek gegevens die zijn bijgekomen, gewijzigd en verwijderd. Om deze wijzigingen te detecteren zijn er aantal technische detectie mechanismen die daarvoor typische gebruikt worden :

- Verwijderde gegevens → bijhouden van een logische delete vlag per record
- Nieuwe gegevens → bijhouden van het tijdstip waarop het record is aangemaakt
- Gewijzigde gegevens → bijhouden van het tijdstip waarop het record is gewijzigd
- Het datamodel en de applicatie moeten deze dus mechanismen ingebouwd hebben.
- Indien deze mechanismen niet consequent bestaan in de applicatie en datamodel, kan er ook in de staging omgeving van de datawarehouse een vergelijking gemaakt worden tussen 2 versies van het datamodel die op een ander tijdstip gemaakt zijn. Zo kunnen deze wijzigingen ook gedetecteerd worden.
- Bij analyse van de ORBA/PPT databases is gebleken dat deze technische detectie mechanismen niet consequent aanwezig zijn. Het vergelijk van de database zal dus bijgevolg in de staging omgeving van de datawarehouse omgeving moeten gebeuren. Om dit te kunnen doen moet de gehele database, of toch de entiteiten die nodig zijn voor de rapportering, telkens gesynct kunnen worden door middel van de API.
- De endpoints van de API moeten dus voor elke entiteit een volledige download kunnen voorzien van de data, eventueel met paginering om de hoeveelheid data per request te beperken.
- REST vs GraphQL
 - In de afgelopen jaren zijn er technologieën gelanceerd die beloofden dat ze efficiënter en gebruiksvriendelijker zouden zijn dan hun voorgangers.
 - REST is steeds populairder geworden en dat ging ten koste van een andere, op dat moment minstens zo populaire technologie: SOAP. Waarom? Omdat REST enkel voordelen biedt ten opzichte van SOAP. Het is in het algemeen iets gebruiksvriendelijker en gebruikt minder data dan SOAP, omdat er geen XML meer bij komt kijken.
 - Maar er is inmiddels een nieuwere optie die aan populariteit wint: het in 2012 door Facebook ontwikkelde GraphQL. In 2015 werd de technologie open source. GraphQL omschrijft zichzelf als een 'query-taal voor API's'. Het heeft niet alleen netwerkresources nodig, maar ook minder front-end-resources, wat zorgt voor een betere totaalprestatie.
 - Zonder al te veel technisch de diepte in te duiken, is het belangrijk om nogmaals aan te halen dat men via GraphQL verschillende resources kan raadplegen in één enkele request; er is dus maar één resource waar men tegenaan programmeert. Bij REST API's zijn resources gescheiden opgezet en zal men doorgaans meerdere requests doen om de benodigde data op te halen.
 - Op de datawarehouse workshop, die doorging op 29 november 2019, hebben we reeds een vergelijk gemaakt tussen een REST en een GraphQL API. De insteek was toen om ORBA/PPT data ter beschikking te stellen zonder reeds te weten hoe en welke gegevens opvraagbaar moesten zijn. Rekening houdend met de voordelen die een GraphQL API bieden was er toen meer de neiging om een GraphQL API voor te stellen.
 - We zullen even deze voordelen opsommen en kijken of ze nog steeds gelden met wat we nu weten uit de analyse van de databehoeftes :
 - Beschrijving van de endpoints dan datastructuur: REST is niets meer dan een architectuurstijl, terwijl GraphQL een specificatie heeft: meer dan tweehonderd pagina's met een gedetailleerde beschrijving over hoe het schema moet ontworpen worden. REST is meer wildwest - de enige specificatie die je hebt komt van de server. Het GraphQL voordeel speelt nog steeds, maar er bestaan frameworks die automatisch een REST API beschrijven (bv Swagger), en zo deze tekortkoming van REST tegen over GraphQL compenseren.
 - Underfetching : Stel, we willen een overzicht van alle orders van augustus 2018 die zijn geplaatst door klanten tussen de 15^{de} en de 25^{ste} dag van de maand. We vragen hier dus

verschillende data op: over orders, producten en klanten. Dat zorgt voor diverse calls naar de backend voor elk object. Dit wordt wel underfetching genoemd. Je krijgt niet in één keer alle data die je nodig hebt, en je moet meerdere calls doen.

Aangezien we de API in deze toepassing moeten ontwerpen zodat alle data van alle entiteiten kan opgevraagd worden, en de relaties op de staging omgeving van de datawarehouse kunnen gelegd worden, gaat dit voordeel niet echt meer op.

- Een ander typische issue met REST is juist het tegenovergestelde: overfetching. Misschien hebben we alleen een orderlijst met de klantnamen nodig maar krijgen we ook alle blogcontent. Wanneer een app via een mobiel netwerk vragen moet afhandelen, wil je overfetching natuurlijk niet. Dit voordeel geldt ook niet meer daar we juist wel alle gegevens van alle entiteiten willen ophalen.
- Tijdens de workshop van 29 november 2019 werden ‘Underfetching’ en ‘overfetching’ als nadelig aanzien. Hierdoor stelden we toen voor om GraphQL te gebruiken.
- Echter in de loop van de studie is gebleken dat ‘Underfetching’ en ‘overfetching’ geen nadeel waren.
- GraphQL is een oplossing die gebruikt wordt indien men adhoc opvragingen wenst te doen. De studie heeft uitgewezen dat er een offload nodig is van alle data, en geen nood aan adhoc opvragingen, waardoor de REST API als conclusie van de studie de betere oplossing is.
- JSON vs XML
 - Er zijn redenen waarom een API de gegevens JSON dan wel XML formaat zou afleveren. Een vergelijk tussen de twee kan bv. hier gevonden worden: https://www.w3schools.com/js/js_json_xml.asp
 - Nu, het is vrij eenvoudig om het formaat van de gegevens afhankelijk te maken van het formaat dat de gebruiker van de API het liefst wenst. Dit kan via de ‘Accept’ header van de request naar de API. Aangezien deze mogelijkheid dus bestaat, is het niet nodig om hiervoor een keuze te maken.

4.6.2. Backward compatibiliteit

Een API is backward compatible als een programma dat gebruik maakt van de API op dezelfde manier blijft werken, zonder aanpassingen, tegen nieuwe versies van de API.

Er moet een verschil gemaakt worden tussen de API backward compatibel te maken tussen grote en kleine wijzigingen.

Een nieuwe versie van een API maken is zeer zeldzaam. Een nieuwe versie houdt meestal grote globale wijzigingen in met wijzigingen in modellen en business logica. Om deze wijzigingen mogelijk te maken wordt er doorgaans een versienummer in de URI opgenomen.

Bvb: <https://some.domain/api/v1/customers>

Bij een nieuwe versie van de API wordt dan dat versie nummer verhoogd.

Er zijn ook aandachtspunten die kleine wijzigingen mogelijk maken in bestaande API's zonder een gehele nieuwe versie te moeten deployen:

Testen

- Het is belangrijk een set van geautomatiseerde testen te hebben die falen wanneer er compatibiliteitsproblemen geïntroduceerd worden in een nieuwe versie.

Voeg alleen parameters toe

- Verwijder nooit bestaande optionele of verplichte parameters die zijn doorgegeven aan of geretourneerd door de API.
- Als men een typefout naar de productie-API-versie heeft gepusht, verwijder dit veld dan niet! Voeg een nieuw veld toe met het juiste benaming en zorg ervoor dat de API de waarde uit het juiste veld gebruikt. Markeer het veld met een typefout als verouderd en verwijder het bij een nieuwe release uit de API-versie.

Maak nooit optionele parameters verplicht

- Parameters die optioneel zijn moeten niet meegegeven worden door die client. Door die verplicht te maken moeten ze dus wel meegegeven worden door de client. Dit kan bestaande bevragingen van de API breken.

Voeg altijd extra HTTP-responscode toe die wordt geretourneerd door de API

- Deze is vergelijkbaar met punt over parameters, maar een beetje anders.
- Als clients alleen wachten op HTTP-respons 200 of 503 van clients, kan een nieuwe respons 501 leiden tot onverwacht gedrag van clients.

Verwijder of wijzig nooit het bestaande gedrag van de HTTP-responscode

- Als u een bepaalde HTTP-response gebruikt om een uitzondering te identificeren (er gebeurt iets) en u hebt clients die deze methode en antwoord gebruiken om te identificeren dat er iets mis is gegaan, en dan plots diezelfde HTTP-response wordt gebruikt om succes aan te geven, wordt de backward compatibiliteit verbroken.

4.6.3. Authenticatie & autorisatie

Aangezien de nood aan Open Data momenteel niet bestaat, en in het geval die zou nodig zijn deze rapportering beter vanuit de datawarehouse kan aangeboden worden, houden we hiermee geen rekening in het voorstel voor het beveiligen van de API.

Authenticatie vs Autorisatie

Het onderscheid kennen tussen authenticatie en autorisatie is belangrijk om te begrijpen hoe RESTful API's werken en waarom toegang wordt geaccepteerd of geweigerd:

- Authenticatie is de verificatie van de inloggegevens. Dit proces bestaat uit het verzenden van de inloggegevens van de RAS-client naar de RAS-server in platte tekst of gecodeerd met behulp van een authenticatieprotocol.
- Autorisatie is de verificatie ofdat de toegang is toegestaan. Autorisatie vindt plaats na succesvolle authenticatie. Met andere woorden: authenticatie betekent dat u bent wie u bent en autorisatie vraagt of u toegang heeft tot een bepaalde bron.

Aangezien het hier momenteel gaat over een API waarbij ofwel alle, ofwel geen data moet opvraagbaar zijn is men in dit geval ook geautoriseerd als er succesvol geauthentiseerd wordt.

Om een API goed te beveiligen moeten we rekening houden met een aantal aandachtspunten:

- Hou het simpel – Beveilig een systeem net zo veilig als het nodig is. Telkens een oplossing complexer gemaakt wordt dan nodig, is er een kans dat er “gat” ontstaat in de beveiliging.
- Gebruik altijd HTTPS – Door altijd SSL te gebruiken, kunnen inloggegevens zelfs bestaan uit een random gegenereerd toegangstoken dat naar de server wordt gestuurd in de header. Het is relatief simpel op te zetten, en het versimpeld in grote mate de manier waarop de beveiliging kan geïmplementeerd worden.
- Gebruik een ge-hash-ed paswoord – Sla nooit een paswoord op in platte tekst. Zelfs als een site wordt gehackt en de hash in de handen komen van de hacker kan hij deze niet zomaar gebruiken om aan te loggen aan het systeem.
- Geeft nooit informatie bloot in de URL – Gebruikersnamen, wachtwoorden, sessietokens en API-sleutels mogen niet in de URL voorkomen, omdat dit kan worden weggeschreven in de logs van een webserver, waardoor ze gemakkelijk kunnen worden misbruikt. Bv.
[//https://api.domain.com/user-management/users/{id}/someAction?apiKey=abcd123456789 //](https://api.domain.com/user-management/users/{id}/someAction?apiKey=abcd123456789)
 Nooit doen!!, de bovenstaande URL onthult de API-sleutel. Gebruik deze vorm van beveiliging dus nooit.
- Overweeg OAuth – Hoewel basic authentication goed genoeg is voor de meeste API's en indien correct geïmplementeerd het ook veilig is, kan misschien ook OAuth overwogen worden. Het OAuth 2.0-autorisatiekader stelt een applicatie van een derde partij in staat om een in de tijd beperkte toegang tot een HTTP-service te verkrijgen.
 - Hiermee rekening houdend, en wetende dat de Identity Server die momenteel reeds in gebruik is om ORBA en PPT te authentifieren overweg kan met OAuth2 is het de meest logische keuze om dit protocol te gebruiken om de API te beveiligen.
 - Hierbij zal het gedeelde geheime sleutel op een gehashte manier opgeslagen worden, en zal elke stad via de Admin applicatie een eigen geheime sleutel kunnen instellen.
 - Dat de communicatie tussen server en client over SSL verloopt is uiteraard een evidentie.

4.6.4. Overzicht van de API endpoints

ORBA					
Tabel Naam	Audit	Log del?	In API?	DWH Antw?	API Endpoint
BusinessUnits	N	N	Y	Y	API/ORBA/BusinessUnits
Calculations	N	N	N	N	
DefaultProjects	N	Y	Y	Y	API/ORBA/DefaultProjects
DraftDefaultProjects	N	N	N	N	
DraftIndicatorGoals	N	N	N	N	
DraftIndicatorProcesses	N	N	N	N	
DraftIndicators	N	Y	N	N	

DraftProjectGoals	N	N	N	N	
DraftProjects	N	Y	N	N	
EmailLogs	N	N	N	N	
Emails	N	N	N	N	
Employees	Y	Y	Y	Y	API/ORBA/Employees
Events	N	N	N	N	
GoalResponsibles	N	N	Y	Y	api/orba/GoalResponsibles
Goals	N	N	Y	Y	api/orba/Goals
Groupings	N	N	Y	Y	api/orba/Groupings
IndicatorChangeRequests	N	N	N	N	
IndicatorGoals	N	N	Y	Y	api/orba/IndicatorGoals
IndicatorGroupings	N	N	Y	Y	api/orba/IndicatorGroupings
IndicatorIssues	N	N	Y	Y	api/orba/IndicatorIssues
IndicatorLogs	N	N	N	N	
IndicatorProcesses	N	N	Y	Y	api/orba/IndicatorProcesses
IndicatorProjects	N	N	Y	Y	api/orba/IndicatorProjects
IndicatorReleases	N	N	Y	N	
IndicatorReporting	Y	Y	Y	Y	api/orba/IndicatorReporting
IndicatorReportingException	N	N	Y	Y	api/orba/IndicatorReportingException
Indicators	N	Y	Y	Y	api/orba/Indicators
IssueGoals	N	N	Y	Y	api/orba/IssueGoals
IssueGroupings	N	N	Y	Y	api/orba/IssueGroupings
IssueLogs	N	N	N	N	
IssueReporting	N	N	Y	Y	api/orba/IssueReporting
Issues	N	Y	Y	Y	api/orba/Issues
Notifications	N	N	N	N	
Periods	N	N	Y	Y	api/orba/Periods
ProcessAnalyses	N	Y	Y	Y	api/orba/ProcessAnalyses
ProcessAnalysisAttachment	N	N	N	N	
ProcessAnalysisGoals	N	N	Y	Y	api/orba/ProcessAnalysisGoals
ProcessAnalysisGroupings	N	N	Y	Y	api/orba/ProcessAnalysisGroupings

ProcessAnalysisLogs	N	N	Y	Y	api/orba/ProcessAnalysisLogs
ProcessAnalysisProcesses	N	N	Y	Y	api/orba/ProcessAnalysisProcesses
ProcessAnalysisReporting	N	N	Y	Y	api/orba/ProcessAnalysisReporting
Processes	N	N	Y	Y	api/orba/Processes
ProjectAttachment	N	N	N	N	
ProjectGoals	N	N	Y	Y	api/orba/ProjectGoals
ProjectGroupings	N	N	Y	Y	api/orba/ProjectGroupings
ProjectIssues	N	N	Y	Y	api/orba/ProjectIssues
ProjectLogs	N	N	Y	Y	api/orba/ProjectLogs
ProjectProcessAnalyses	N	N	Y	Y	api/orba/ProjectProcessAnalyses
ProjectProcesses	N	N	Y	Y	api/orba/ProjectProcesses
ProjectReporting	N	N	Y	Y	api/orba/ProjectReporting
Projects	N	Y	Y	Y	api/orba/Projects
ProjectTypes	N	N	Y	Y	api/orba/ProjectTypes
ReleaseChangeRequests	N	N	Y	Y	api/orba/ReleaseChangeRequests
ReleaseGoals	N	N	Y	Y	api/orba/ReleaseGoals
ReleaseGroupings	N	N	Y	Y	api/orba/ReleaseGroupings
ReleaseIssues	N	N	Y	Y	api/orba/ReleaseIssues
ReleaseLogs	N	N	N	N	
ReleaseReporting	N	N	Y	Y	api/orba/ReleaseReporting
Releases	N	N	Y	Y	api/orba/Releases
ReportingDates	N	N	Y	Y	api/orba/ReportingDates
SearchPreferences	N	N	N	N	
TargetsFormulas	N	N	N	N	
Visits	N	N	N	N	

Tabel 11 – Overzicht ORBA endpoints

PPT					
CalendarDeviation	N	N	Y		api/ppt/CalendarDeviation
CellStyle	N	N	N		
Company	N	N	N		
Planning	Y	N	Y		api/ppt/Planning
PlanningAction	N	N	Y		api/ppt/PlanningAction
PlanningColumnData	N	N	N		
PlanningTask	N	N	Y		api/ppt/PlanningTask
PlanningTaskLink	N	N	Y		api/ppt/PlanningTaskLink
ProjectType	N	N	N		
TaskCalendar	Y	N	Y		api/ppt/TaskCalendar
TaskColumn	N	N	N		
Template	N	N	N		
TemplateColumnData	N	N	N		
TemplateTask	N	N	N		
TemplateTaskLink	N	N	N		

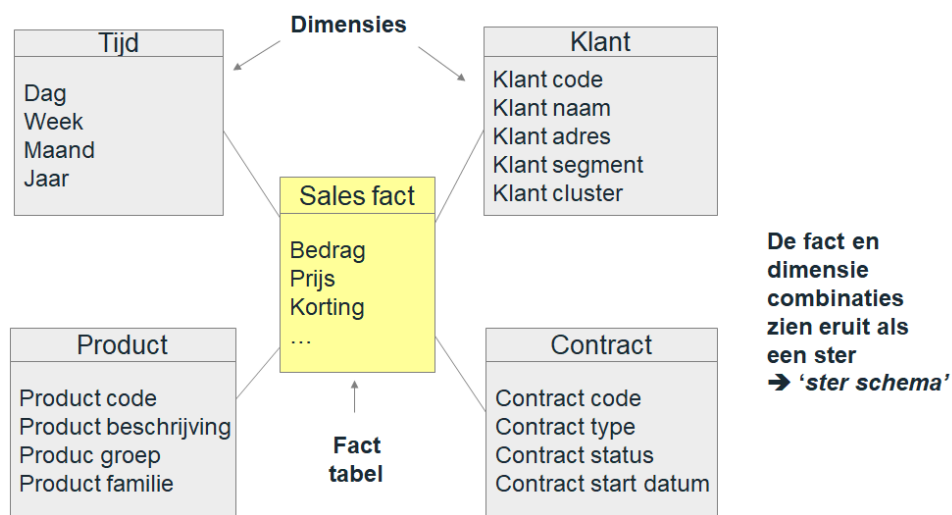
Tabel 12 – Overzicht PPT Endpoints

4.7. Data model

De referentie architectuur voorziet in 2 lagen om het data warehouse op te bouwen.

Eenzijds een wat men ook noemt, Data warehouse “publication layer”, die de data marts omvat en typisch volgens een ster model worden ontwikkeld. In ster model worden de meetwaarden in de fact tabellen bijgehouden en de contextuele informatie of analyse assen in wat men noemt dimensie tabellen.

Onderstaande figuur geeft een voorbeeld van een dergelijk ster model.



Figuur 15 – Voorbeeld data model volgens ster model

Anderzijds voorziet de referentie architectuur ook een Data Warehouse “integration layer” en wordt die voorzien als een genormaliseerd data model.

Een alternatieve methode die veelal wordt gebruikt bij data warehouses die snel veranderden business modellen dienen te supporteren is het Data vault data model. Data vault wordt beschreven door Dan Linstedt (<https://danlinstedt.com/>) die de data modelleert in 3 type tabellen Hub tabellen (die de keys bijhouden), Link tabellen (die de relaties tussen de hubs bijhouden) en Satelliet tabellen waarin de descriptieve informatie over de keys in de hubs worden bijgehouden.

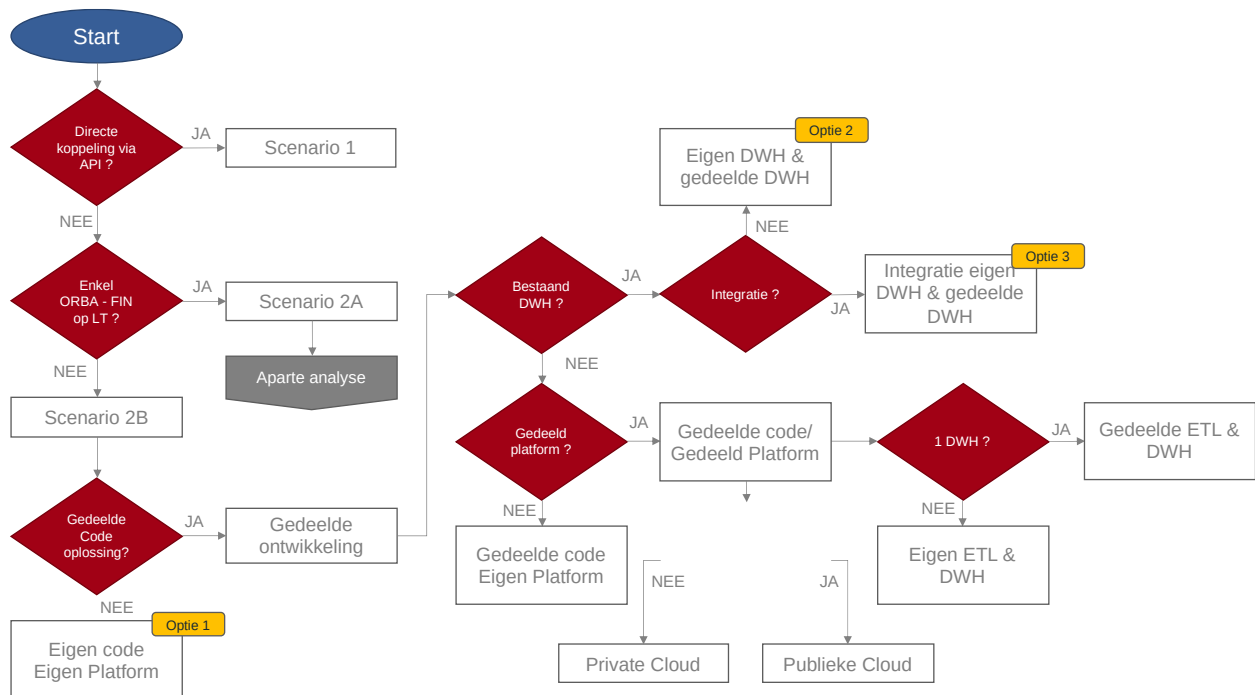
In onderstaande figuur wordt een voorbeeld van een data model weergegeven.



Tenslotte werd ook de toepasbaarheid van OSLO tijdens dit haalbaarheidsonderzoek bekeken. OSLO is echter niet echt data model maar een geheel van semantische definities is eerder van toepassing op de brondata. Voor het type data in de betrokken bronsystemen ORBA en financiële pakketten zijn echter binnen OSLO nog geen standaarden gedefinieerd.

4.8. Beslissingsboom

Naar aanleiding van de conclusies van dit haalbaarheidsonderzoek dienen een aantal beslissingen genomen te worden. Onderstaande figuren trachten dit verloop schematisch weer te geven.



Figuur 17 – Beslissingsboom

5. FINANCIËLE OVERZICHTSTABEL

Naar aanleiding van de uitgevoerde haalbaarheidsstudie werd ook een eerste highlevel inschatting gemaakt van de gerelateerde kosten aan de scenario's 2A en 2B. Voor 2B werden ook de 2 varianten uit de studie m.b.t. de extra financiële data en de extra financiële rapporten uitgewerkt.

Om echter ook een éénduidige vergelijking te kunnen maken voor hetzelfde bereik hebben we ook variant 2B berekend op basis van enkel de financiële data gerelateerd aan ORBA zoals ook in 2A voorzien is. Bij deze variant werd ook uitgegaan van de 3 zelfde financiële bronnen die geïntegreerd worden, met name CIPAL, CEVI en SAP.

Als platform werd steeds een Microsoft SQL 2019 gebaseerde omgeving voorzien per centrumstad met volgende karakteristieken (Variant Gedeelde code/Eigen infrastructuur)

- Om de kosten per stad zo laag mogelijk te houden hebben we één enkele server voorzien voor de ganse Datawarehouse omgeving. Dit is echter geen “best practice” opzet.
- Voor het datawarehouse systeem werd volgende configuratie voorzien :
 - Windows 2016 Server met 4 vCPU en 80 RAM
 - Een Tier 2 Disk van 50 GB en een Tier 1 Disk voor de Database van 200 GB, inclusief 30 dagen backup
 - De volledige server en onderliggende infrastructuur is “managed”. Voor de MS SQL beperkt het management zich wel tot het patchen van de database. Er is geen verder onderhoud meegerekend.
- Tot slot werden ook 10 Power BI Pro licenties voorzien per stad voor de gebruikers van het platform.

Noteer, mocht er geopteerd worden voor om te werken met een eigen instantie (Instance) op een gedeeld platform (Gedeelde code/Gedeeld platform) dan zullen er zich sowieso een paar schaalvoordelen naar voor komen.

- Hoewel de opzet iets complexer is dan voor 1 stad, zal deze opzet niet 8 keer moeten overgedaan worden.
- Als het platform gedeeld wordt dan moet de architectuur zeker geoptimaliseerd worden, maar deze architectuur zal echter minder middelen (resources) verbruiken dan de 8 systemen die nu “naast” elkaar staan te draaien.
- Ook het beheer van het gedeelde platform zal iets moeilijker zijn dan voor één stad, maar opnieuw zal het wel niet 8 keer naast elkaar moeten gebeuren, wat nu wel het geval is, wat opnieuw een kostenbesparing met zich mee zal brengen.

Tabel 13 – Overzicht kosten per scenario variant

	Scenario 2A*	Scenario 2B - Zonder extra financiële input & rapportering	Scenario 2B - Met extra financiële input maar zonder financiële rapportering	Scenario 2B - Met financiële rapportering
ORBA/FIN Data	✓	✓	✓	✓
Integratie laag		✓	✓	✓
Extra FIN Data			✓	✓
Extra FIN Rapporten				✓
Stap 1 - Bron koppeling	42.471,00 €	42.471,00 €	48.133,80 €	59.459,40 €
Stap 2 - Transformatie & Integratie	99.099,00 €	78.952,50 €	138.847,50 €	138.847,50 €
Stap 3 - Datamart	26.136,00 €	26.136,00 €	26.136,00 €	41.382,00 €
Stap 4 - Self-Service II	13.068,00 €	13.068,00 €	13.068,00 €	20.691,00 €
Stap 5 - Rapporten	68.675,06 €	68.675,06 €	68.675,06 €	96.921,00 €
Totaal budget Incl. BTW	249.449,06 €	229.302,56 €	294.860,36 €	357.300,90 €
Budget incl BTW/Stad**	31.181,13 €	28.662,82 €	36.857,55 €	44.662,61 €
Opzet Microsoft DWH Omgeving/Stad	3.825,60 €	3.825,60 €	3.825,60 €	3.825,60 €
Totaal eenmalige opzet/Stad (Incl. BTW)	35.006,73 €	32.488,42 €	40.683,14 €	48.488,21 €
Wederkerende kosten Omgeving/Stad (Incl BTW)	30.147,44 €	30.147,44 €	30.147,44 €	30.147,44 €
* Totaal voor 3 bronnen (CEVI, CIPAL, SAP)				
** Verdeeld over 8 centrumsteden				

Volgende parameters werden gebruikt voor de bovenstaande berekeningen

- Drie financiële bronnen werden in acht genomen, zijnde CEVI, CIPAL en SAP (Gent). UBW van Brugge werd niet mee opgenomen vanwege de eigen oplossing in Brugge.
- Budget incl. btw/centrumstad : 8 centrumsteden werden weerhouden. Ook hier werd Brugge niet mee in rekening genomen.

De totale eenmalige kosten dienen in het eerste contractjaar voorzien te worden. De wederkerende kosten zullen beginnen lopen vanaf het ogenblik dat de systemen opgezet worden voor de verschillende steden. Noteer dat dit vrij in het begin van het project zal zijn.

De wederkerende kosten dienen voor een minimale periode van 4 jaar gebudgetteerd te worden, tenzij er voor een Private Cloud oplossing zou gekozen worden.

Het gebruik van een Publieke Cloud is voor deze omgeving mogelijks ook een meer dan waardevol alternatief. Dit vereist echter een aparte studie om hier een budgettaire inschatting van te kunnen maken.

Tot slot lichten we ook nog graag toe dat het mogelijk is om de opzet kosten uit te spreiden op bijvoorbeeld de looptijd van 6 jaar. Rekening houdend met een financieringskost voor deze gespreide betaling geeft dit het resultaat vermeld in onderstaande tabel.

Tabel 14 – Afschrijving van de opzet kosten op 6 jaar

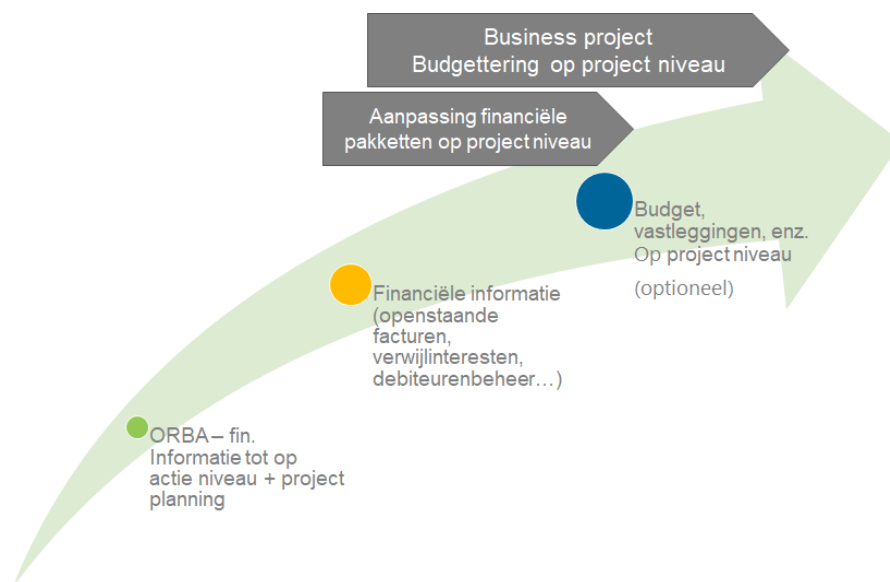
	Scenario 2A*	Scenario 2B - Zonder extra financiële input & rapportering	Scenario 2B - Met extra financiële input maar zonder financiële rapportering	Scenario 2B - Met financiële rapportering
ORBA/FIN Data	✓	✓	✓	✓
Integratie laag		✓	✓	✓
Extra FIN Data			✓	✓
Extra FIN Rapporten				✓
Jaarlijkse kost/stad (Incl. BTW, 6 jaar legislatuur)***	36.961,63 €	36.471,44 €	38.066,57 €	39.585,86 €
*** eenmalige kosten afgeschreven op 6 jaar, met financieringskost				

6. IMPLEMENTATIE PLAN

Om het data warehouse met de ORBA en financiële data te bouwen stellen we voor om dit gradueel op te bouwen in iteraties. Op deze manier kan men snel toegevoegde waarde realiseren en ervaringen opdoen op een relatief beperkte scope, die meegenomen worden bij een verdere iteratie.

Onderstaande figuur geeft de volgorde waarin we voorstellen om de iteraties te implementeren.

- Eerst, de combinatie ORBA data, financiële informatie tot op actie niveau
- Daarna de bijkomende financiële data
- Daarna op het moment dat ook de financiële systemen zijn aangepast, de financiële opvolging op project niveau



Figuur 18 – Implementatie roadmap

Een data warehouse implementatie dient niet beschouwd te worden als een éénmalig project, maar eerder als een journey. Indien succesvol zullen bijkomende uitbreidingen gevraagd worden en ook binnen het domein van ORBA - financiële gegevens zullen aanpassingen nodig zijn en aan bijkomende noden te voldaan.

OVERZICHT TABELLEN

Tabel 1 – Overzicht AS IS situatie in de centrumsteden.....	14
Tabel 2 – Overzicht behoeften met betrekking tot de strategische project opvolging	16
Tabel 3 – Overzicht behoeften met betrekking tot de wettelijke beleidsopvolging.....	16
tabel 4 Overzicht behoeften met betrekking tot andere data.....	17
Tabel 5 – Overzicht behoeften qua type rapportering	17
Tabel 6 – Overzicht behoeften met betrekking tot de opvolgingsniveau per stad.....	18
Tabel 7 – Overzicht behoeften met betrekking tot het type gebruikers en ondersteuning	19
Tabel 8 – Overzicht van de behoeften met betrekking tot de BI infrastructuur.....	19
Tabel 9 – Overzicht van de behoeften met betrekking tot veiligheid.....	19
Tabel 10 – Overzicht van de behoeften met betrekking tot de BI technologie	20
Tabel 11 – Overzicht ORBA endpoints	43
Tabel 12 – Overzicht PPT Endpoints.....	44
Tabel 13 – Overzicht kosten per scenario variant.....	49
Tabel 14 – Afschrijving van de opzet kosten op 6 jaar	50

OVERZICHT FIGUREN

Figuur 1 – Details opvolgingsniveau's	4
Figuur 2 – Opzet bronsystemen - ORBA.....	15
Figuur 3 – Opzet bronsystemen – financiële systemen	15
Figuur 4 – Voorstelling scenario 2a	23
Figuur 5 – Referentie Architectuur (B. Inmon).....	24
Figuur 6 - Voorstelling scenario 2b.....	24
Figuur 7 – Schaalvoordeel t.o.v. individualisatie graad.....	27
Figuur 8 – Voorstelling uitbreidbaarheid scenario 2a	29
Figuur 9 -Voorstelling individuele uitbreiding scenario 2b zonder integratie.....	31
Figuur 10 – Voorstelling individuele uitbreiding scenario 2 b met integratie.....	32
Figuur 11 – Voorstelling gezamenlijke uitbreiding scenario 2b	33
Figuur 12 – Overzicht optie 1 bestaand data warehouse	34
Figuur 13 – Overzicht optie 2 bestaand data warehouse	35
Figuur 14 – Overzicht optie 3 bestaand data warehouse	36
Figuur 16 – Voorbeeld data model volgens ster model	45
Figuur 17 – Voorbeeld data model volgens Data Vault methodologie	46
Figuur 18 – Beslissingsboom	47
Figuur 19 – Implementatie roadmap	51