Business-Oriented Data Modelling Masterclass –
Balancing Engagement, Agility, and Complexity

Presented by
Adept Events and Clariteq Systems Consulting

Alec Sharp
Consultant
Clariteq Systems Consulting Ltd.
West Vancouver, BC, Canada
asharp@clariteq.com
www.clariteq.com

Instructor (2) background…

Bram Laumans, IR kwadraat – *bram.laumans@ir2.nl*

LinkedIn: Bram Laumans

- 28 years experience as database developer, consultant & teacher
- Married to Dénise
- 2 daughters, 2 dogs & 2 cats.

Consultancy & Development

- Rabobank
- SPA - Schoonderbeek en Partners Advies
- XCESS expertise center b.v.
- Reisiger Verhuur
- Koninklijke Marine
- Schuitema / C1000
- Direktbank (Fortis Bank)
- Alfam (Fortis Bank)
- Foodstep
- Lekkerland
- Nippur
- Gemeente Venlo
- CZ Zorgverzekeringen
- FEI (Thermo Fisher)

Teaching

- SVB - Sociale Verzekeringsbank
- Rabobank – Data & Literacy Program
- HAN - Hogeschool Arnhem Nijmegen
- RAVU – Regional Ambulance service

Day 2

Data vault

- Data modelling technique for Data warehouses
- Invented / documented by Hans Hultgren & Dan Linsted

Modeling the Agile
Data Warehouse
with Data Vault

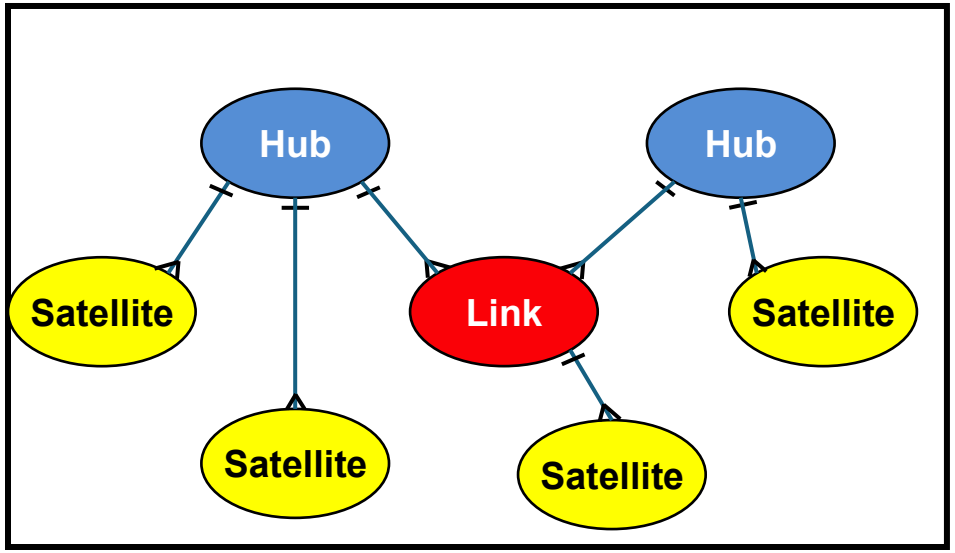
Hans Hultgren, 2012

Data modelling – to what end?

System type → Aspect	Operational system	Data warehouse	Reporting system aka Data mart
Purpose	Store the present truth, run a business	Keep track of “what happened when”	Fixed reports and easy querying for analysis
Redundancy	Usually not	Some, as little as possible	Yes, on purpose
Modelling technique	Entity Relationship (3NF)	Data vault or Anchor modelling	Dimensional
Querying complexity	Medium	Hard to Horrible	Easy
Main issue	Do not run analyses on production systems for fear of slowing down the operational activities (the SELECT * FROM VeryLargeTable – effect)	Thorough understanding of the model needed and even then many joins needed to get an actionable result	A lot of work is needed if you want to add information that is not available within the model
Main advantage	When well designed: lean and mean system that only stores what is necessary	Flexible structure, easy to add new data sources and integrate information	Simple structure, fast for querying

Why Data vault?

- More readily absorb changes (agile)
- Respond well to new subject areas (start small, add new stuff later)
- Manage historical time slices of data (historization)
- Provide full traceability back to source feeds (auditability)
- Grow and adapt with minimal impact (lower TCO)
- Integrate, align & reconcile data (enterprise integration)
- Track, manage and report on exceptions (feedback)



Why not?

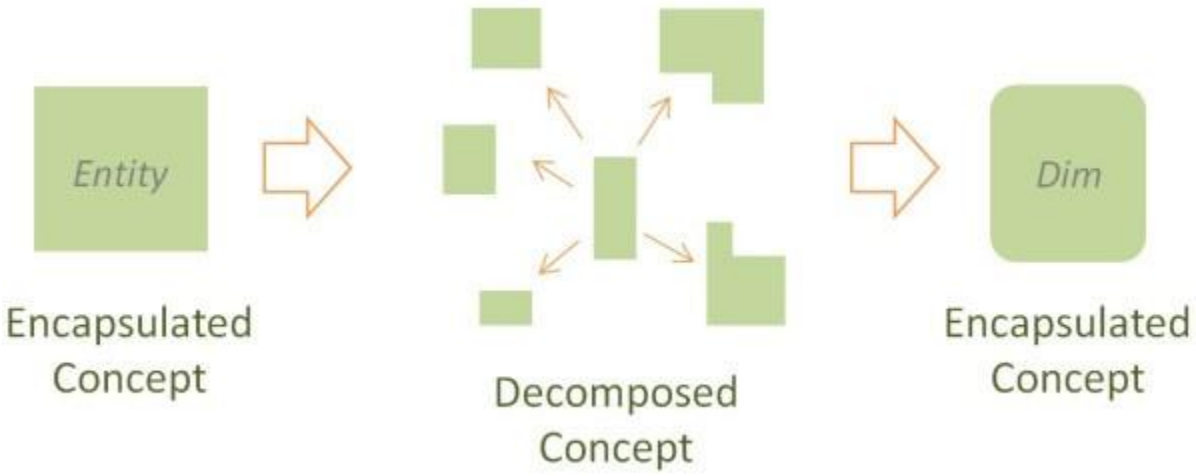
- Different way of thinking
- Hard to query ad hoc for mere mortals

Data vault key concepts

- *Hub = business key*
- *Link = relationship*
- *Satellite = context*

Unified decomposition

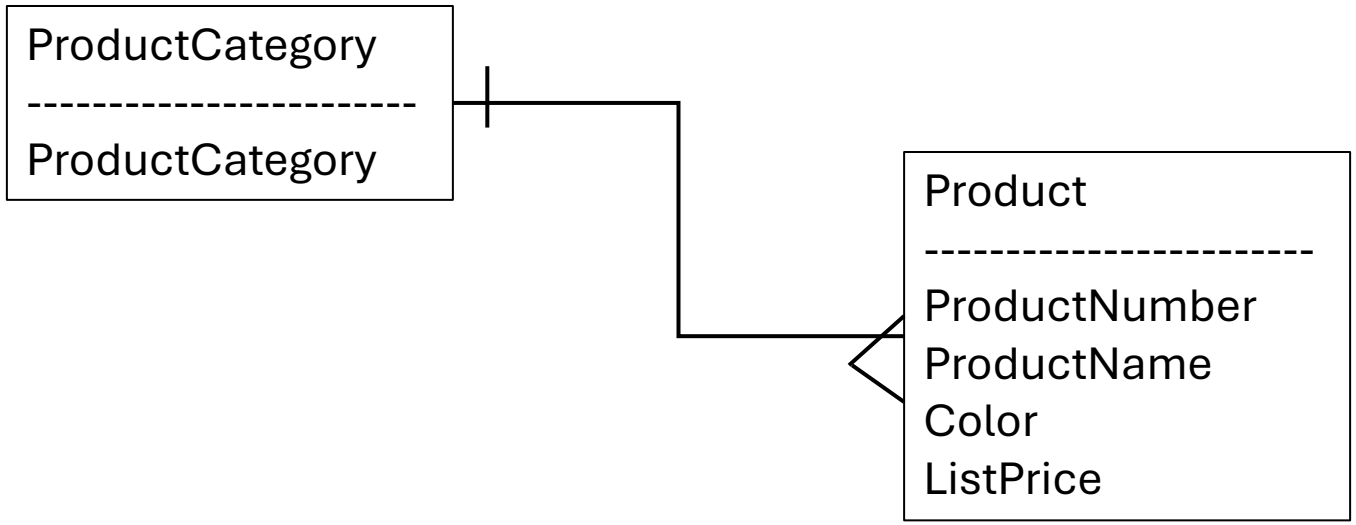
- Breaking up entities into component parts
- Change component independently
  - Flexibility, adaptability, agility
  - Capture from different sources
  - Integrate and load to DM (dims & facts)
- Decomposition by
  - Change frequency
  - Source system



Operational system | Data warehouse | Data mart

Example: AdventureWorks - Bikes

We have
1) a reference entity “ProductCategory”
2) an entity called “Product”



Assertions:
- *Each* Product is categorized by *one* ProductCategory
- *Each* ProductCategory categorizes *one or more* products

Example: AdventureWorks - bikes

What does the data look like?

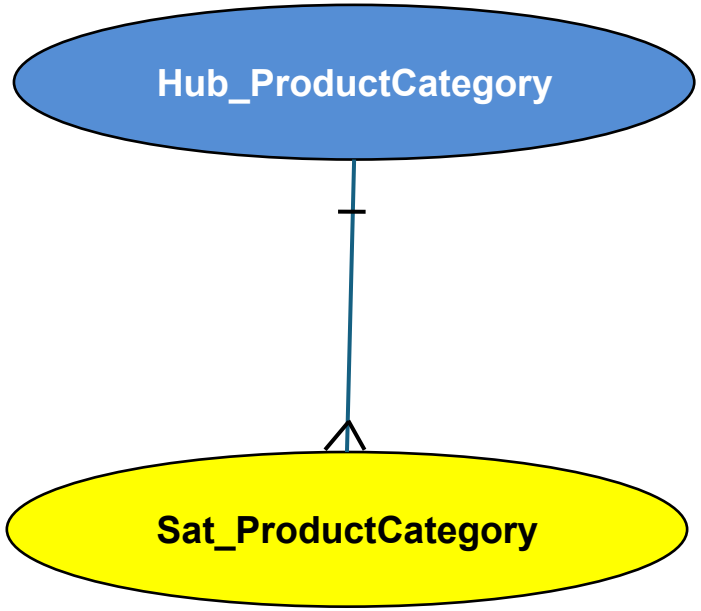
ProductCategoryID	ProductCategory
1	Mountain Bikes
2	Road Bikes
3	Touring Bikes

ProductID	ProductCategoryID	ProductNumber	ProductName	Color	DaysToManufacture	ReorderPoint	StandardCost	ListPrice
800	2	BK-R64Y-44	Road-550-W Yellow, 44	Yellow	4	75	713.08	1120.49
801	2	BK-R64Y-48	Road-550-W Yellow, 48	Yellow	4	75	713.08	1120.49
975	2	BK-R79Y-44	Road-350-W Yellow, 44	Yellow	4	75	1082.51	1700.99
976	2	BK-R79Y-48	Road-350-W Yellow, 48	Yellow	4	75	1082.51	1700.99
961	3	BK-T18Y-44	Touring-3000 Yellow, 44	Yellow	4	75	461.44	742.35
963	3	BK-T18Y-54	Touring-3000 Yellow, 54	Yellow	4	75	461.44	742.35
956	3	BK-T79Y-54	Touring-1000 Yellow, 54	Yellow	4	75	1481.94	2384.07

We have 97 different bikes, in colors yellow, red, black, blue and silver. The sample shows only 7 yellow bikes.

ProductCategory → Data vault

Attribute	Role	Change behaviour	Data vault target	Object
ProductCategoryID	Technical key	Never	Sat	Sat_ProductCategory
ProductCategory	Business key	Hardly ever	Hub	Hub_ProductCategory



ProductCategoryID	ProductCategory
1	Mountain Bikes
2	Road Bikes
3	Touring Bikes

Product → Data vault

Attribute	Role	Change
ProductID	Technical key	
ProductCategoryID	Foreign key --> ProductCategory	
ProductNumber	Business key	
ProductName	The name of this product	
Color	The color of this product	
DaysToManufacture	The number of days it takes to finish this product	
ReorderPoint	The number of items we have in stock below which we will reorder a new set (?) of this product	
StandardCost	The cost of making this product	
ListPrice	The retail price of this product	

Which attributes will change:

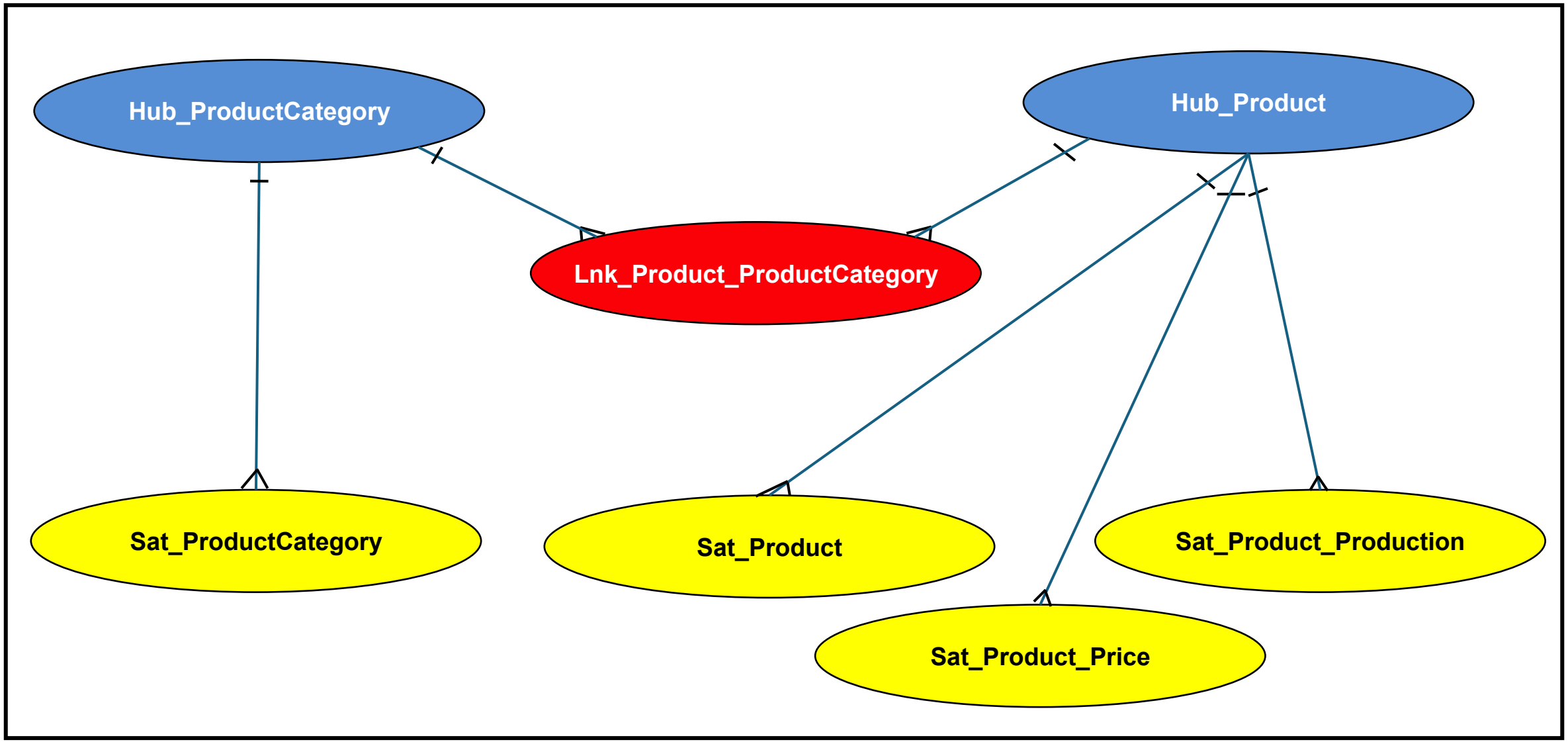
- Never
- Hardly ever
- Sometimes
- Often

ProductID	ProductCategoryID	ProductNumber	ProductName	Color	DaysToManufacture	ReorderPoint	StandardCost	ListPrice
800	2	BK-R64Y-44	Road-550-W Yellow, 44	Yellow	4	75	713.08	1120.49
801	2	BK-R64Y-48	Road-550-W Yellow, 48	Yellow	4	75	713.08	1120.49
975	2	BK-R79Y-44	Road-350-W Yellow, 44	Yellow	4	75	1082.51	1700.99
976	2	BK-R79Y-48	Road-350-W Yellow, 48	Yellow	4	75	1082.51	1700.99
961	3	BK-T18Y-44	Touring-3000 Yellow, 44	Yellow	4	75	461.44	742.35
963	3	BK-T18Y-54	Touring-3000 Yellow, 54	Yellow	4	75	461.44	742.35
956	3	BK-T79Y-54	Touring-1000 Yellow, 54	Yellow	4	75	1481.94	2384.07

Product → Data vault

Attribute	Role	Change behaviour	Data vault target	Object
ProductID	Technical key	Never	Sat	Sat_Product
ProductCategoryID	Foreign key --> ProductCategory	Hardly ever	Link	Lnk_Product_ProductCategory
ProductNumber	Business key	Hardly ever	Hub	Hub_Product
ProductName	The name of this product	Hardly ever	Sat	Sat_Product
Color	The color of this product	Hardly ever	Sat	Sat_Product
DaysToManufacture	The number of days it takes to finish this product	Might	Sat	Sat_Product_Production
ReorderPoint	The number of items we have in stock below which we will reorder a new set (?) of this product	Might, especially if this products popularity changes	Sat	Sat_Product_Production
StandardCost	The cost of making this product	Periodically, depending on the supply chain	Sat	Sat_Product_Price
ListPrice	The retail price of this product	Periodically, depending on cost and company policy	Sat	Sat_Product_Price

Data vault – ProductCategory and Product



Data vault “stamp fields”

“Stamp fields” are added to all records to keep track of “what happened when”.

- *Hubs and Links only record “when was I created”.*
- *Satellites also keep track of the start and end date of their validity.*

Stamp fields	**Hub**	**Lnk**	**Sat**	**Meaning**
LoadDate	Yes	Yes	Yes	The date and time at which this information was loaded into the Data Warehouse
ValidFrom	No	No	Yes	The date (and time) *as of which* this information is valid.
ValidTo	No	No	Yes	The last date (and time) up to which this information is valid (have a “tot” versus “tot en met” discussion). For the "current actual record" the value typically is "31 dec 9999“
SourceSystem	Yes	Yes	Yes	The name of the system that first offered this information
LoadBatchLogID	Yes	Yes	Yes	The technical key of the batch process that created this record.
LastUpdateBatchLogID	No	No	Yes	The technical key of the batch process that modified this record (usually when the ValidTo date was updated)

Data vault “to the extreme” = Anchor modelling

- *each* attribute gets its own satellite
- Only realistically achievable with automation
- No duplicates (why not?)

Why not do Anchor modelling?

- Different way of thinking
- Impossible to query ad hoc for mere mortals

Summary

- Know for which type of system you do data modelling: Operational, Data Warehouse or Data Mart
- Data vault can help you build your data warehouse incrementally
- Data vault may help you keep track of things (the Audit Department will love you)
- Work: you need to think about your satellites (unified decomposition)
  - By change frequency
  - By source system
- You need something “on top of” your data vault data warehouse for analysis (a dimensional data mart)

Day 3

Example Collective Labour Agreement

Employee

EmployeeID	Name	ContractStartDate	ContractEndDate
47312	John	1-4-2025	<NULL>

Salary

SalaryID	EmployeeID	Salary	CreateDate	EffectiveDate
100457	47312	1000	18-3-2025	1-4-2025
104744	47312	1025	28-6-2025	1-4-2025

ContractStartDate is April 1st 2025

SalaryPayment

SalaryPaymentID	EmployeeID	SalaryPaymentDate	SalaryMonth	Amount	Description
256147	47312	24-4-2025	apr-25	1000	Salary
260434	47312	23-5-2025	mei-25	1000	Salary
260435	47312	23-5-2025	mei-25	750	Holiday pay
264721	47312	24-6-2025	jun-25	1000	Salary
269008	47312	24-7-2025	jul-25	1025	Salary
269009	47312	24-7-2025	apr-25	25	Salary
269010	47312	24-7-2025	mei-25	25	Salary
269011	47312	24-7-2025	jun-25	25	Salary

25th of June 2025:
New Collective Labour Agreement (CAO):
Salary +2,5%
as of Jan 1st, 2025

Questions:
What salary did John receive

1. *in* April 2025?
2. *for* April 2025?

Example Collective Labour Agreement

What salary did John receive
1. *in* April 2025?
2. *for* April 2025?

New question: What salary did John receive
1. *in* April 2025?
2. *for* April 2025 with the knowledge of date X ?

The answer depends on when you ask the question:

1. *in* April 2025 John recieved a salary of 1000

2. The *for* question is complicated:
- Between March 18th and June 24th: 1000
- Between June 25th and July 23rd: *it should have been* 1025, but it is 1000
- After July 24th: 1025

18 25 24

	March	April	May	June	July	***Total***
Salary in HR system	1000	1000	1000	1000	1025	*4025*
Corrected on 28-6	0	1025	1025	1025	1025	*4100*
Paid	0	1000	1000	1000	1100	*4100*