# Reasons for data modelling

## Common data system complaints

Simple items of information are held in many places.

> When customer changes address, this involves manual updates to several different computer files.

There's information in there, but we cannot get it out.

> Last week, I bought something from a BT shop. The shop assistant had to type my name, address, postcode and telephone number into the computer. Surely BT already has this information on its database?

Strategic decisions are not supported.

> If I want to do WHAT IF analysis, I probably need to build my own spreadsheet - but where do I get the data to put in the spreadsheet?

Data structures inflexible, difficult to change.

> Direct debit example
>
> Year 00

## But in the perfect data system ...

Each item of information is stored in only one place.

Each item of information is updated promptly.

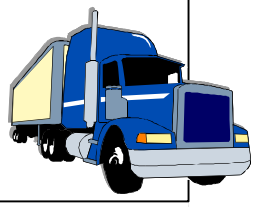
Simple and urgent changes to business rules and requirements can be implemented quickly and easily.

Management information can be obtained in a form that supports management activity

> operational decisions
>
> control, tracking and audit
>
> strategic decisions

## Three principles of articulation

- Clear structure
- Separation of parts
- Connection of parts



***we express information structure in the form of an information model***

## Using an information or data model

Design systems and databases

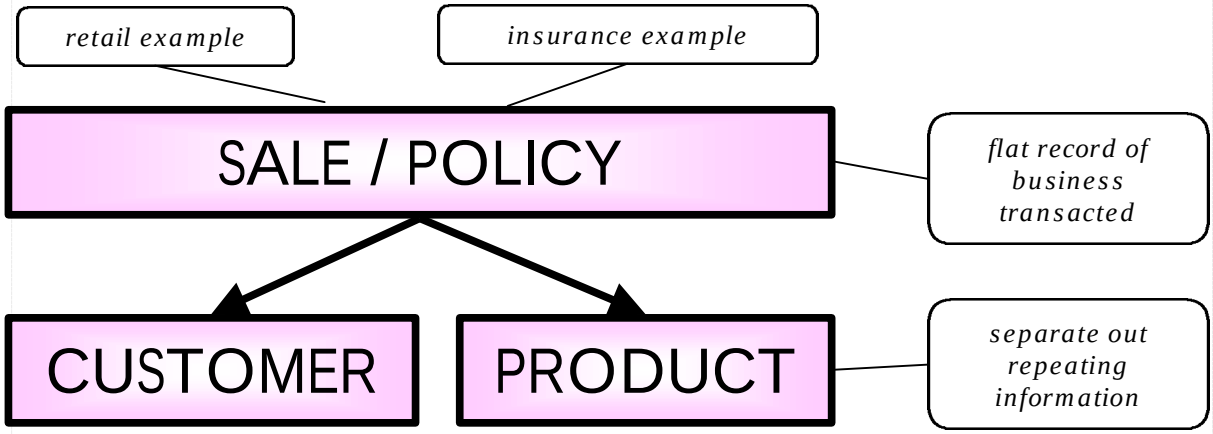
Evaluate and implement packages, components and external information services

Build bridges and interfaces between diverse systems and data stores

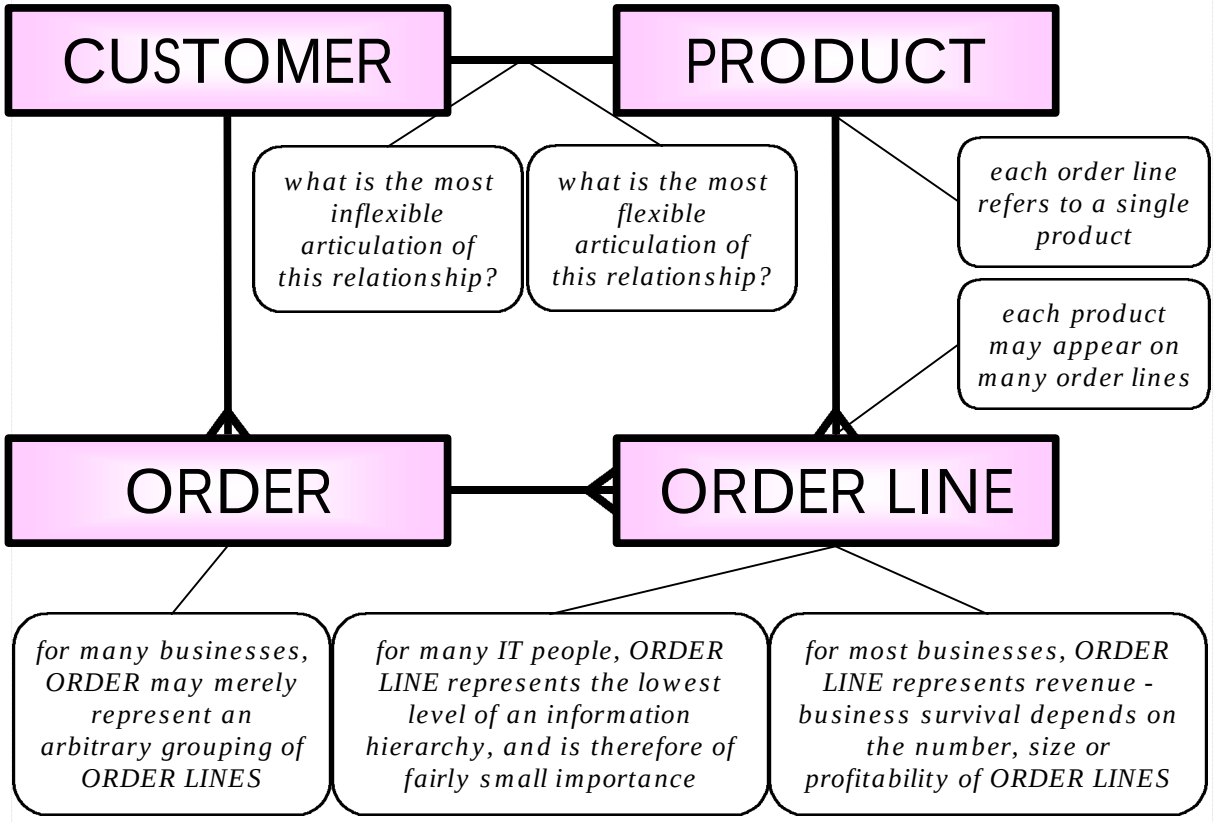
Provide management information

Plan information and systems development

# Articulating information



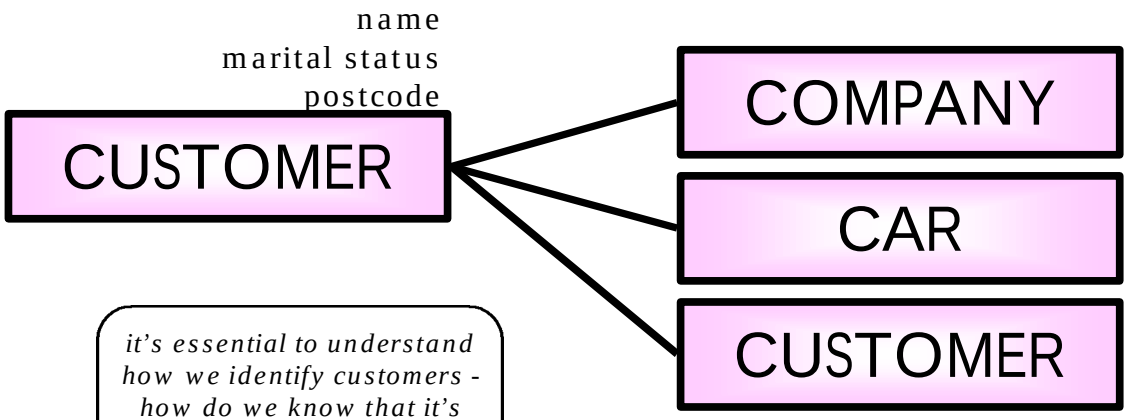

# Typical information model - articulation of SALE



# Detailing information

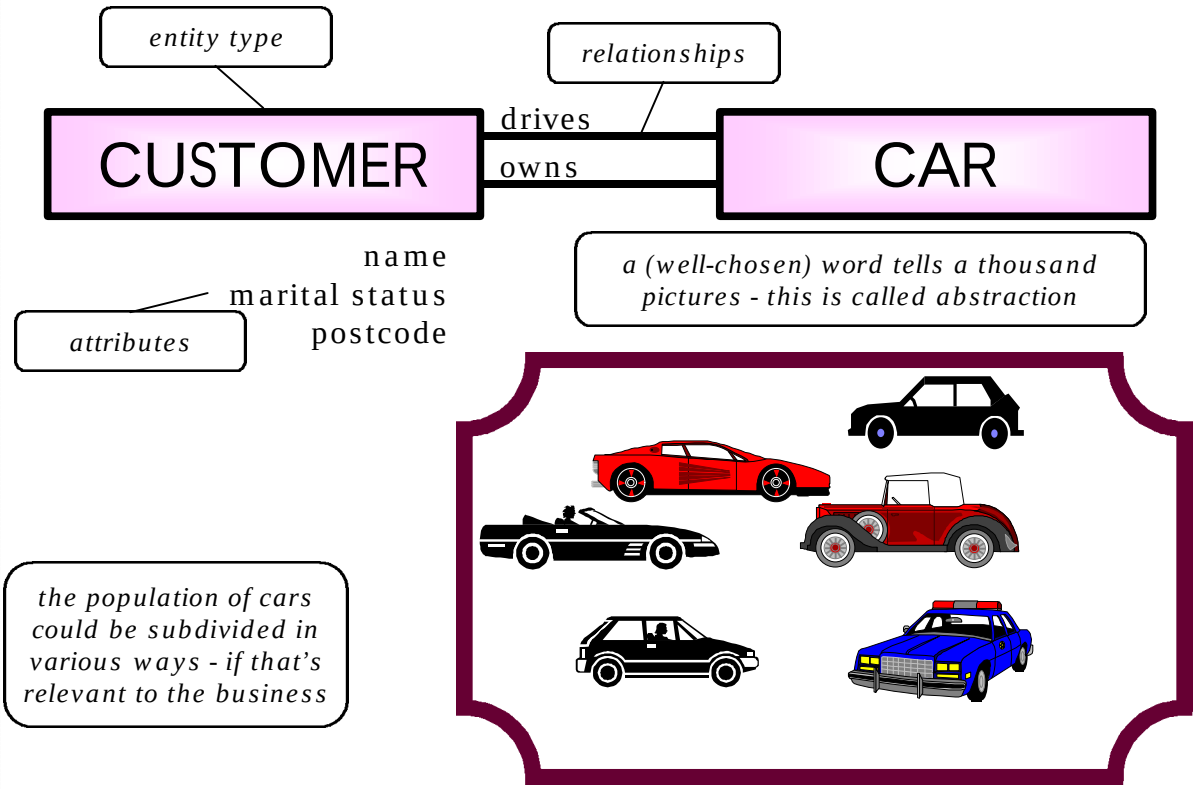
*some of the things we want to know about customers are direct attributes of customer*

*some of the things we want to know about customers are direct attributes of other things associated with the customer*




*it's essential to understand how we identify customers - how do we know that it's him again? - we call this the identifier or key*

# Completing the model



# Three notions of information

***1 "A by-product of a business transaction."***

- Each insurance policy has a different "customer".
- Each "customer" has a separate name and address.
- These "customers" may coincide - one real person may appear many times in our files.
- Difficult to consolidate data across products to produce such a unified view.
- Some business strategies demand a "unified" view of "customer".
- Therefore this view of information is inadequate - typically inhibits business strategy.

***2 "A fact about a real-world object."***

- "Customers" in our database represent "real" people in the "real world".
- Since there is only one "real" person, there should be only one database record.
- Similarly, each database record should represent a single entity or object in the "real world".
- Our knowledge of the real world is typically incomplete and imperfect.
- Customer identity may not be straightforward.
  - married couples
  - small businessman
  - proxies & agents

***3 "A difference that makes a difference."***

- Databases and systems should support any **distinctions** that business strategy demands.
- Databases and systems should support any **connections** that business strategy demands.
- Business rules and regulations demand **correctness** and **consistency**.

# Quality Indications

**Quality of model as end-product**

- Well-scoped
- Correct - i.e. accurately representing the intended enterprise, to the satisfaction of all interested parties. (This implies it is comprehensible to all interested parties.)
- Externally complete - i.e. completely describing the intended enterprise, covering all required functionality.
- Internally complete and consistent - i.e. with no missing cross-references, no unused objects, etc. (Some aspects of this can be automatically checked by an appropriate modelling tool.)
- Well-documented - i.e. with adequate and meaningful descriptions of all objects
- Non-redundant - i.e. only representing each fact/requirement once, without unjustified duplication or overlap between objects.
- Coordinated - i.e. consistent with architectures and policies, and with an appropriate level of consistency with other related models.
- Stable - i.e. capable of absorbing minor future changes to the intended enterprise without major changes to the model. (This characteristic is also known as resilience or robustness.)

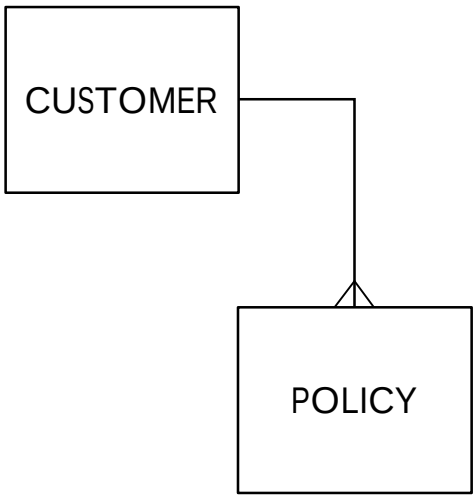
**Quality of modelling process**

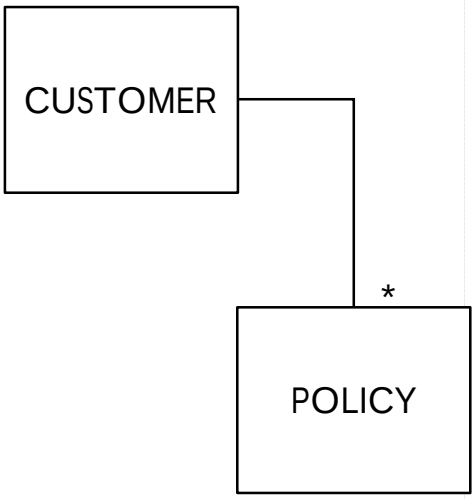
- Ease of agreement by users of the model
- Ease of agreement by users of the design implications of the model
- Minimum 'thrashing' - i.e. going round in circles before agreement can be reached
- Minimum discovery of additional requirements during design and subsequent phases
- Relatively few surprises during implementation
- Low maintenance costs of system (owing to changes in model)
- Maximum learning for participants and entire organization
- Efficient & effective - i.e. achieving a good result with a reasonable expenditure of time and energy

# Two competing notations/jargons

| Relational Modelling | Object Modelling |
|---|---|
| • Entity | • Class |
| • Relationship | • Object type |
| • Attribute | • Association |
| • Subtype | • Inheritance |

*UML = "unified modelling language"*

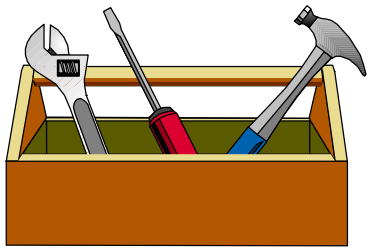



Many (although not all) of the concepts and principles hold equally for relational and object modelling.

# Further reading & other sources

- Richard Veryard, Information Modelling: Practical Guidance (Prentice Hall, 1992)
- Martin Fowler, Analysis Patterns: Reusable Object Patterns (Addison-Wesley, 1997)



- UML
  - http://www.rational.com/
- Cetus Links
  - http://www.cetus-links.org/
- Graphical Tools
  - http://www.methods-tools.com/tools/modeling.html