Business Processes Modelling

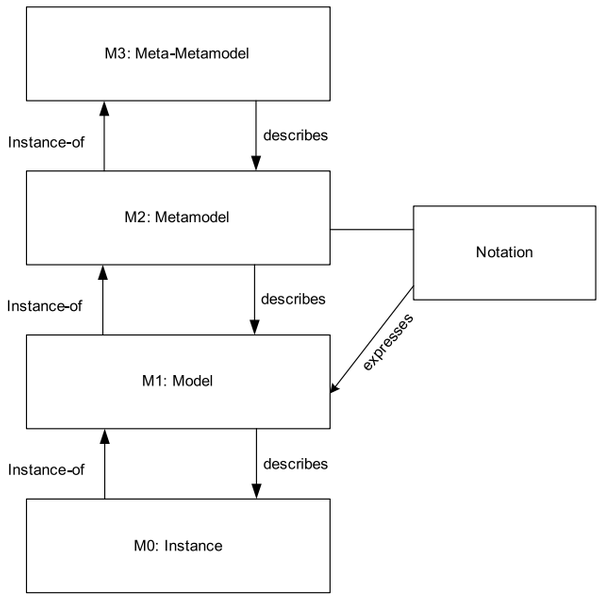
MPB (6 cfu, 295AA)

Roberto Bruni

http://www.di.unipi.it/~bruni

03 - Models and Abstraction

Object



Overview of the conceptual models and abstraction mechanisms in business process modeling

Ch.3.1--3.3 of Business Process Management: Concepts, Languages, Architectures

Model

A model is a simplified representation of reality

"Essentially all models are wrong,
but some are useful"
(George P. Box)

Abstraction

To derive general rules and concepts
from specific examples of some phenomenon,
by selecting only the aspects which are relevant
for a particular purpose

A way to cope with complexity

Guiding principle

Separation of Concerns (SoC)
(to separate a system into distinct features that overlap in functionality as little as possible)

E. W. Dijkstra Archive

the manuscripts of

Edsger W. Dijkstra

1930–2002

Search transcriptions: Search Advanced search.

(photo ©2002 *Hamilton Richards*)

Edsger Wybe Dijkstra was one of the most influential members of computing science's founding generation. Among the domains in which his scientific contributions are fundamental are

- algorithm design
- programming languages
- program design
- operating systems
- distributed processing
- formal specification and verification
- design of mathematical arguments

In addition, Dijkstra was intensely interested in teaching, and in the relationships between academic computing science and the software industry.

http://www.cs.utexas.edu/users/EWD/

On the role of scientific thought (EWD447)

Let me try to explain to you, what to my taste is ***characteristic for all intelligent thinking****.*

It is, that one is willing to study in depth an aspect of one's subject matter ***in isolation for the sake of its own consistency****, all the time knowing that one is occupying oneself only with one of the aspects.*

On the role of scientific thought (EWD447)

...
*We know that a program must be **correct** and we can study it from that viewpoint only;*
*we also know that it should be **efficient** and we can study its efficiency on another day, so to speak.*
*In another mood we may ask ourselves whether, and if so: why, the program is **desirable**.*

But nothing is gained —on the contrary!— by tackling these various aspects simultaneously.

On the role of scientific thought (EWD447)

...
It is what I sometimes have called ***"the separation of concerns"****, which, even if not perfectly possible, is yet the only available technique for effective ordering of one's thoughts, that I know of.*

...
it does not mean ignoring the other aspects, it is just doing justice to the fact that
from this aspect's point of view, the other is irrelevant.

On the role of scientific thought (EWD447)

Business data processing systems are sufficiently complicated to require such a separation of concerns and the suggestion that in that part of the computing world "scientific thought is a non-applicable luxury" puts the cart before the horse: the mess they are in has been caused by ***too much unscientific thought****....*

SoC: an example

HyperText Markup Language (HTML):
organization of webpage content

Cascading Style Sheets (CSS):
definition of content presentation style

JavaScript (JS):
user interactions

Abstractions

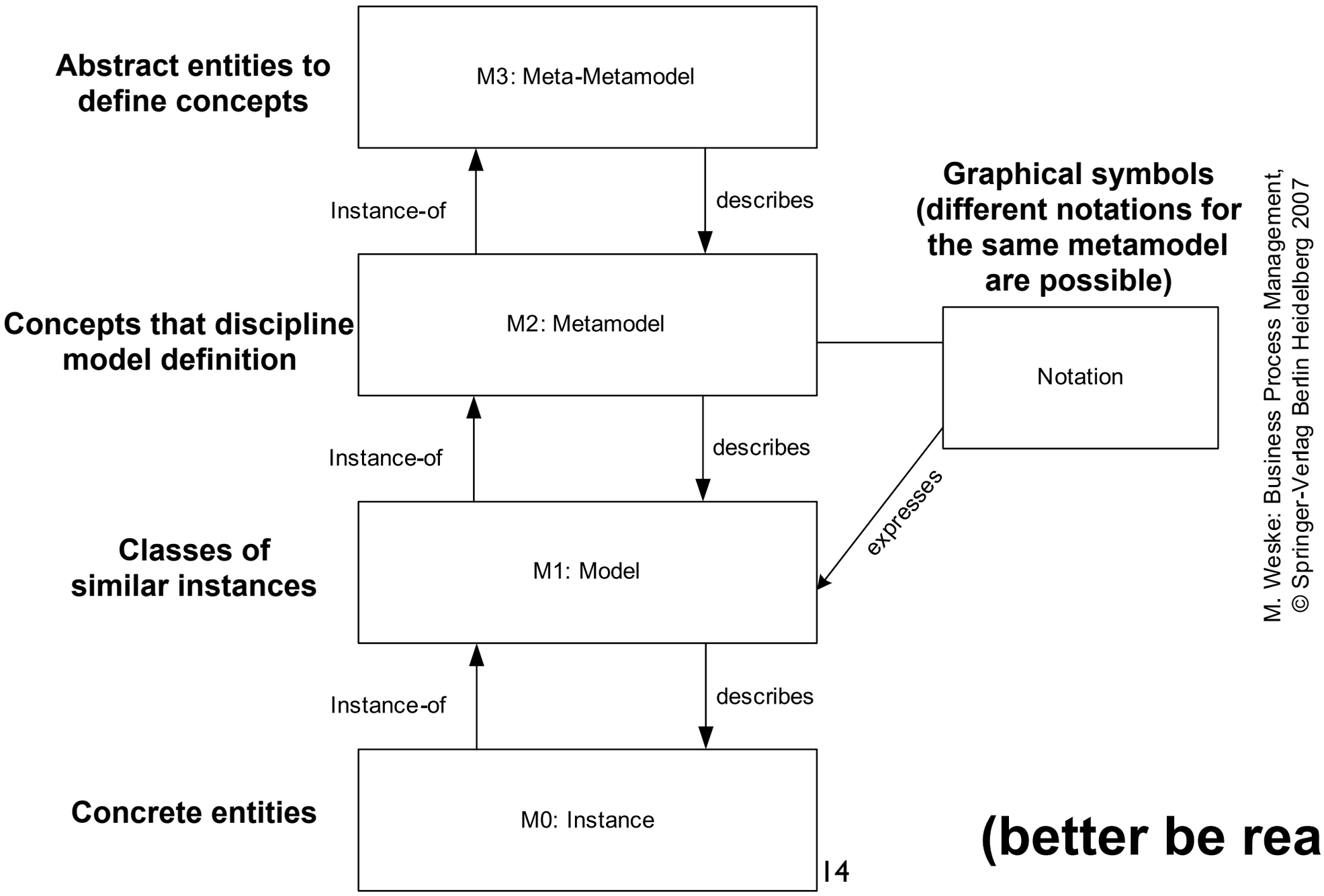
Horizontal: separation at different modeling levels

Aggregation: separation at different granularity levels

Vertical: separation at different subdomains

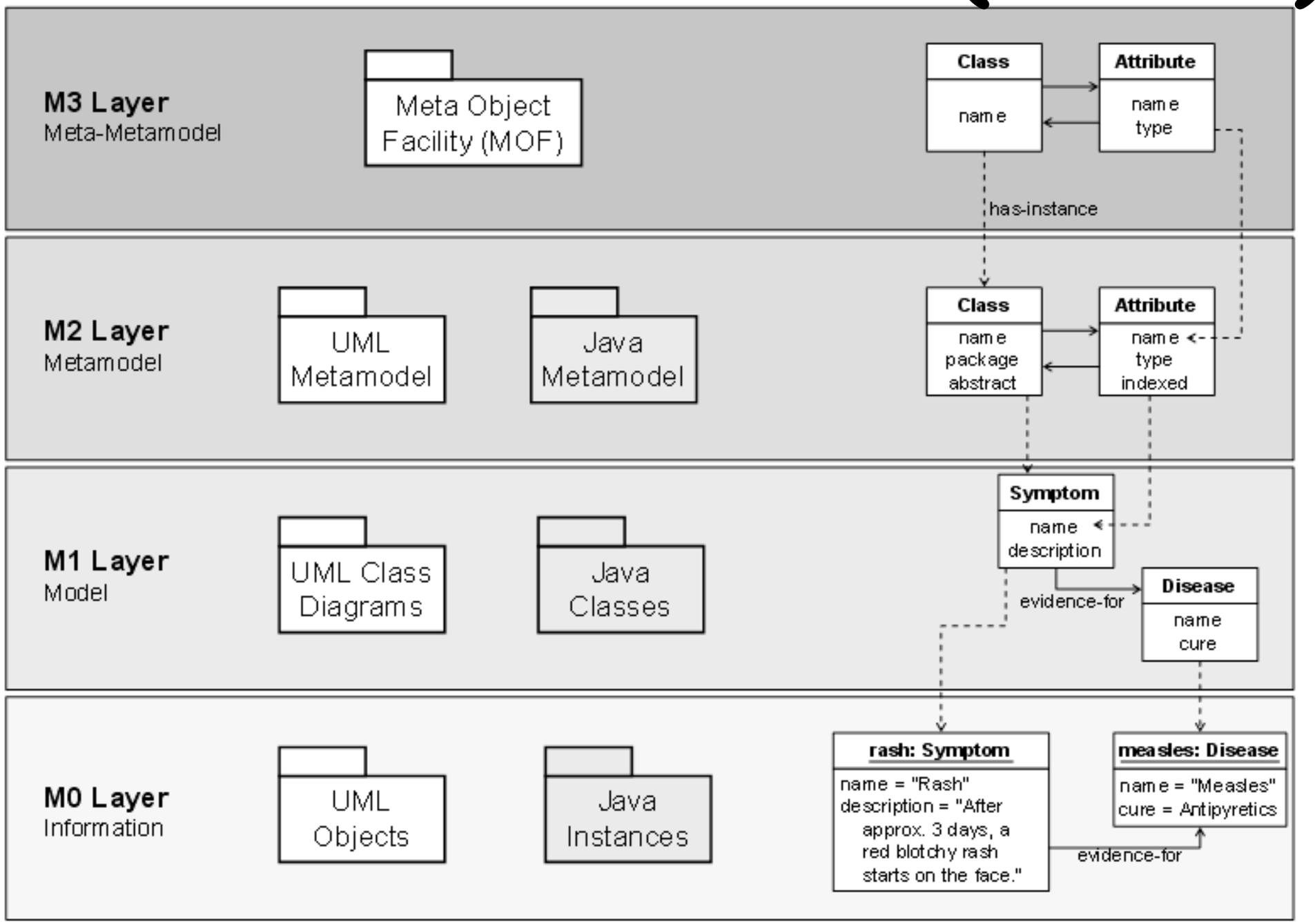
Horizontal Abstraction

Horizontal abstraction (modeling levels)

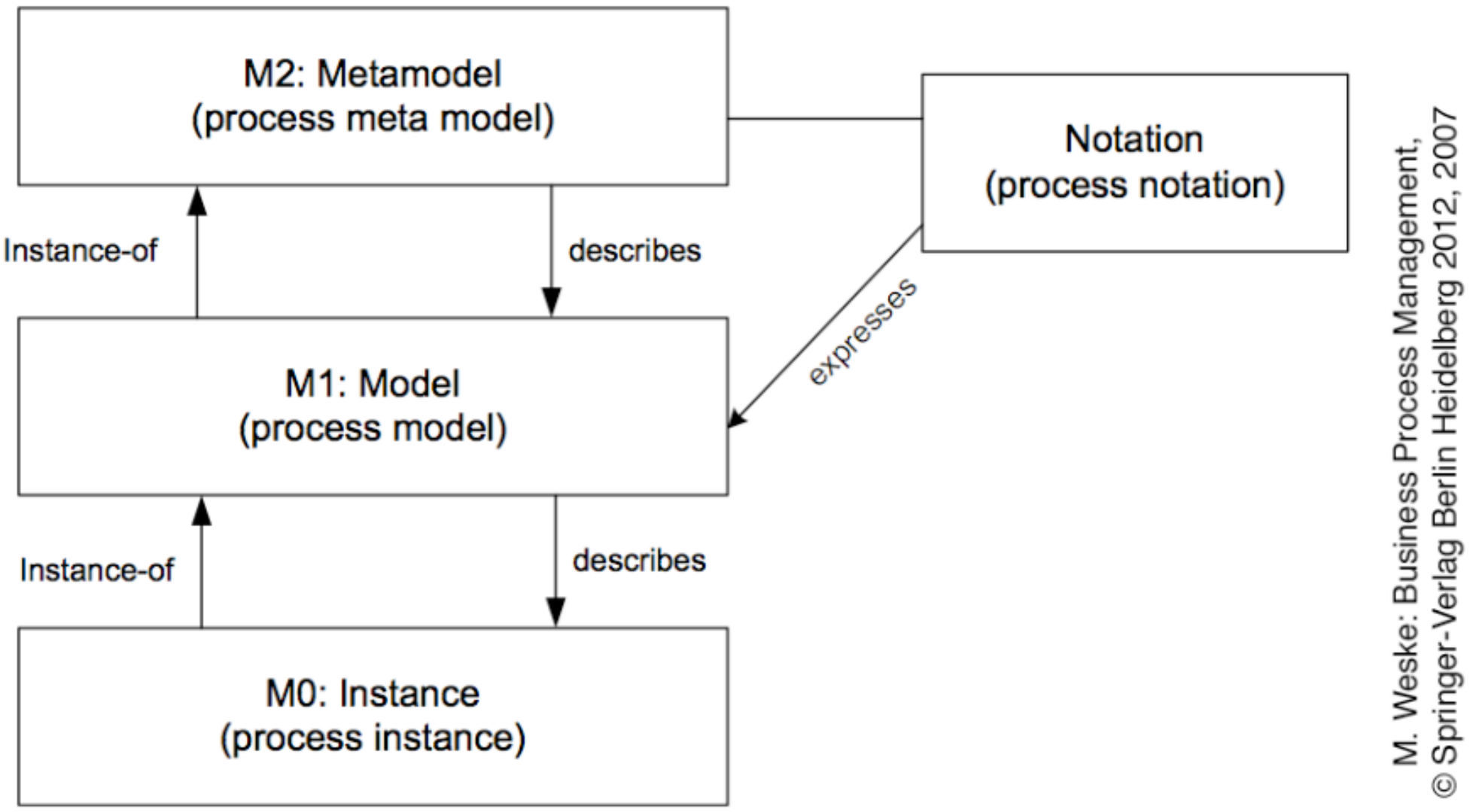


(better be read bottom-up)

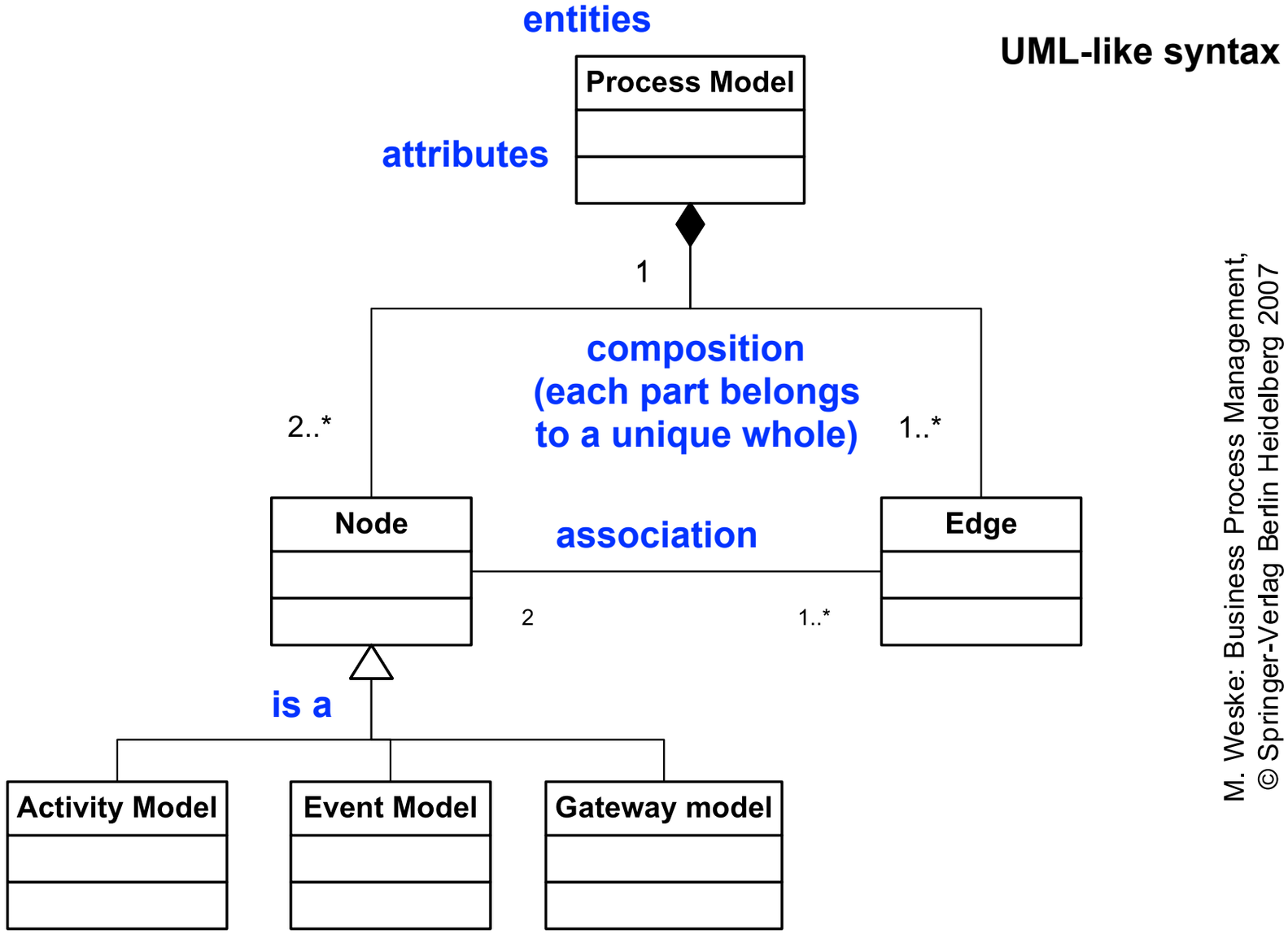
An example: MOF metamodel (OMG)



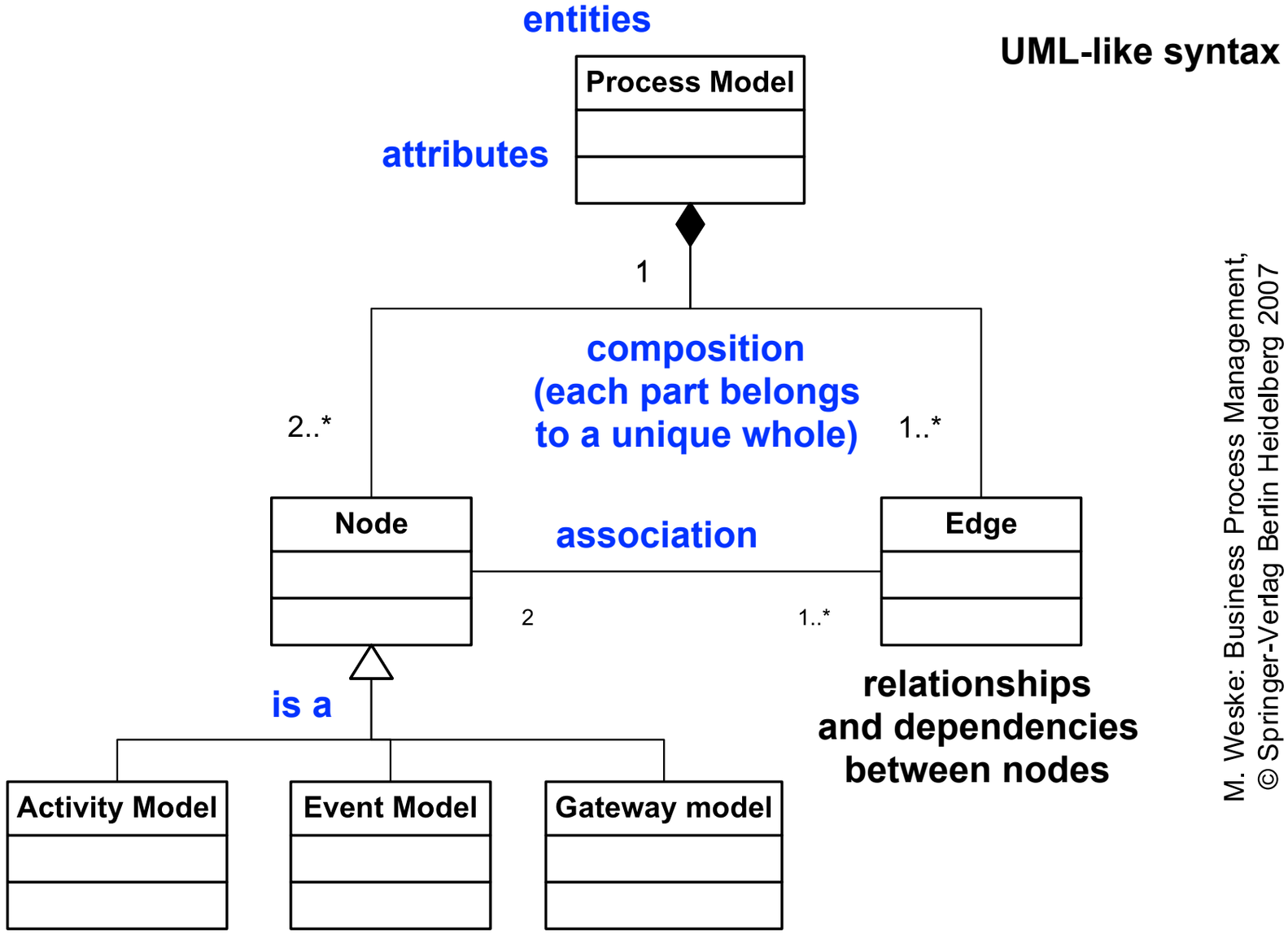
Process models and process instances



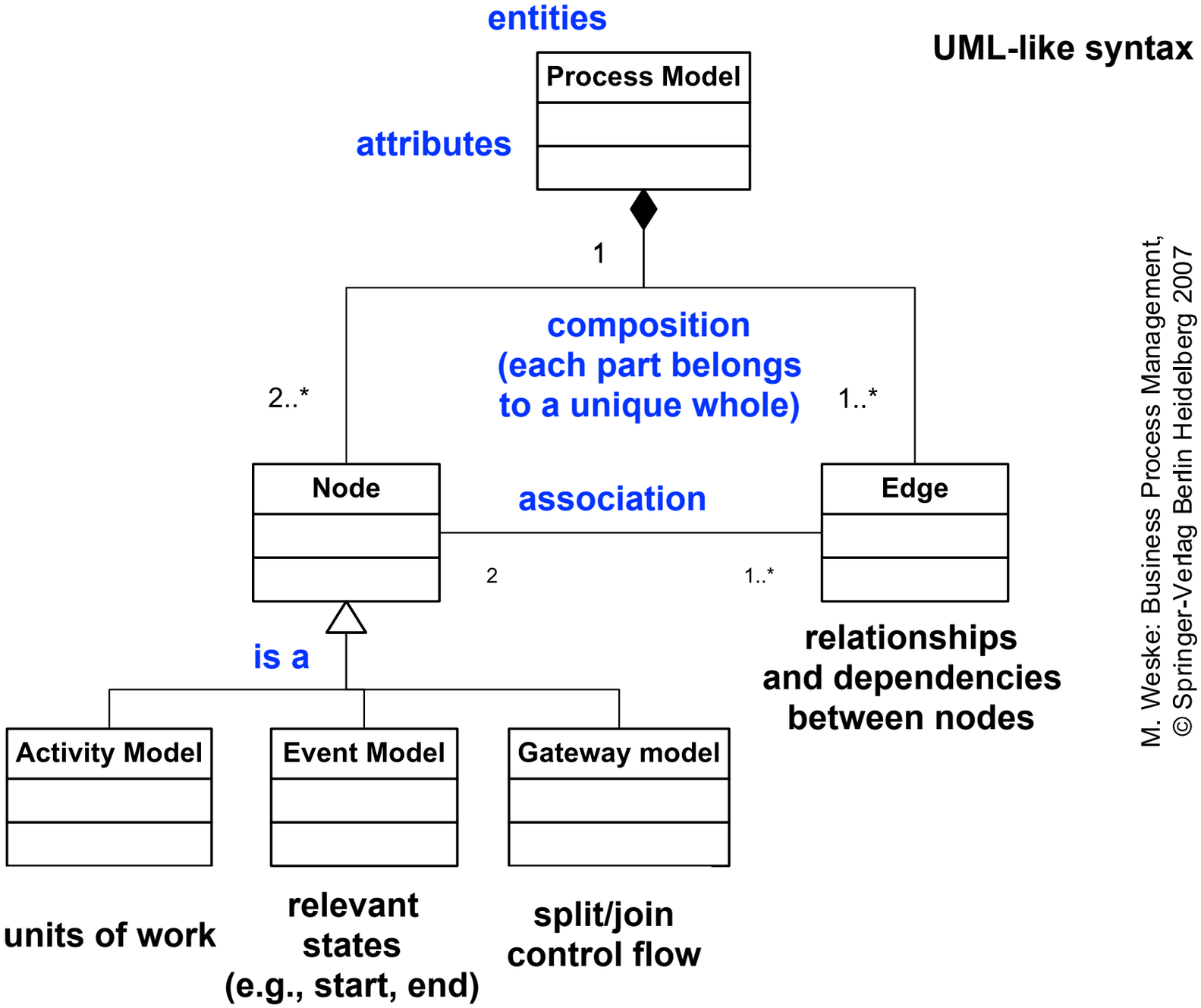
A process metamodel (level M2)



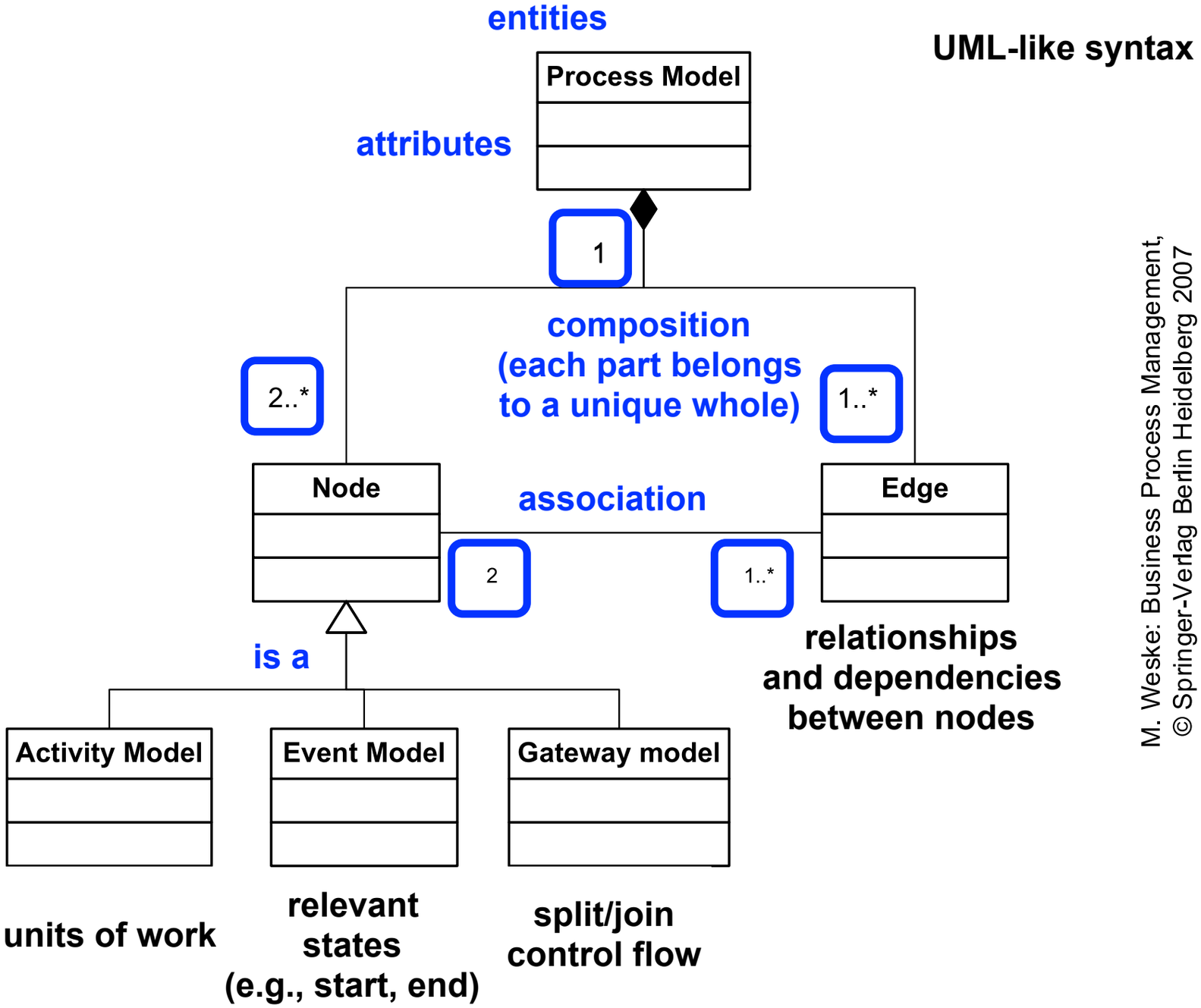
A process metamodel (level M2)



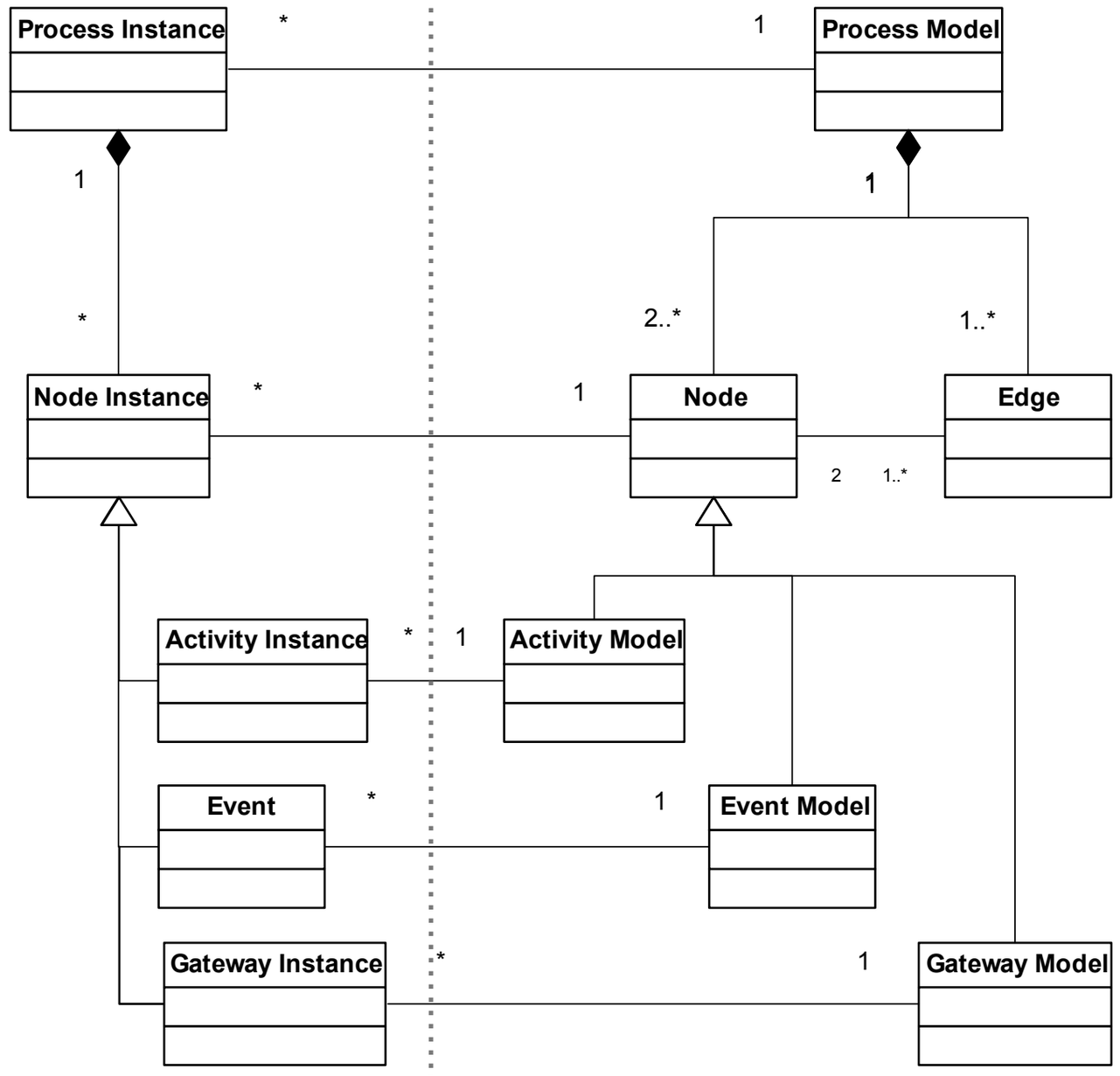
A process metamodel (level M2)



A process metamodel (level M2)



Process models and process instances



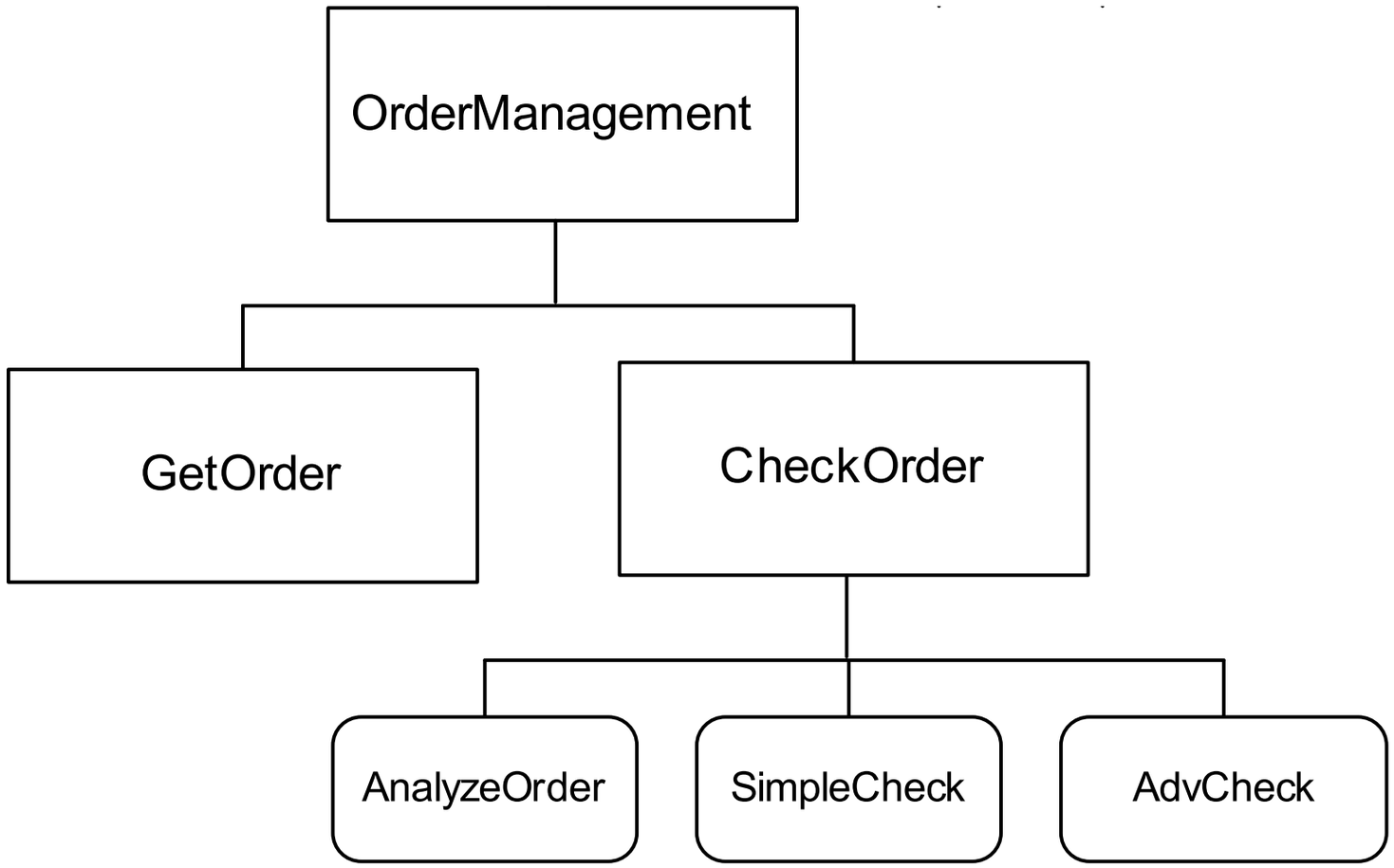
Aggregation Abstraction

Aggregation abstraction

Multiple elements of a lower level of granularity can be grouped and represented by a single artifact at the higher level of granularity

Different from horizontal abstraction, where all entities lie at the same level of granularity

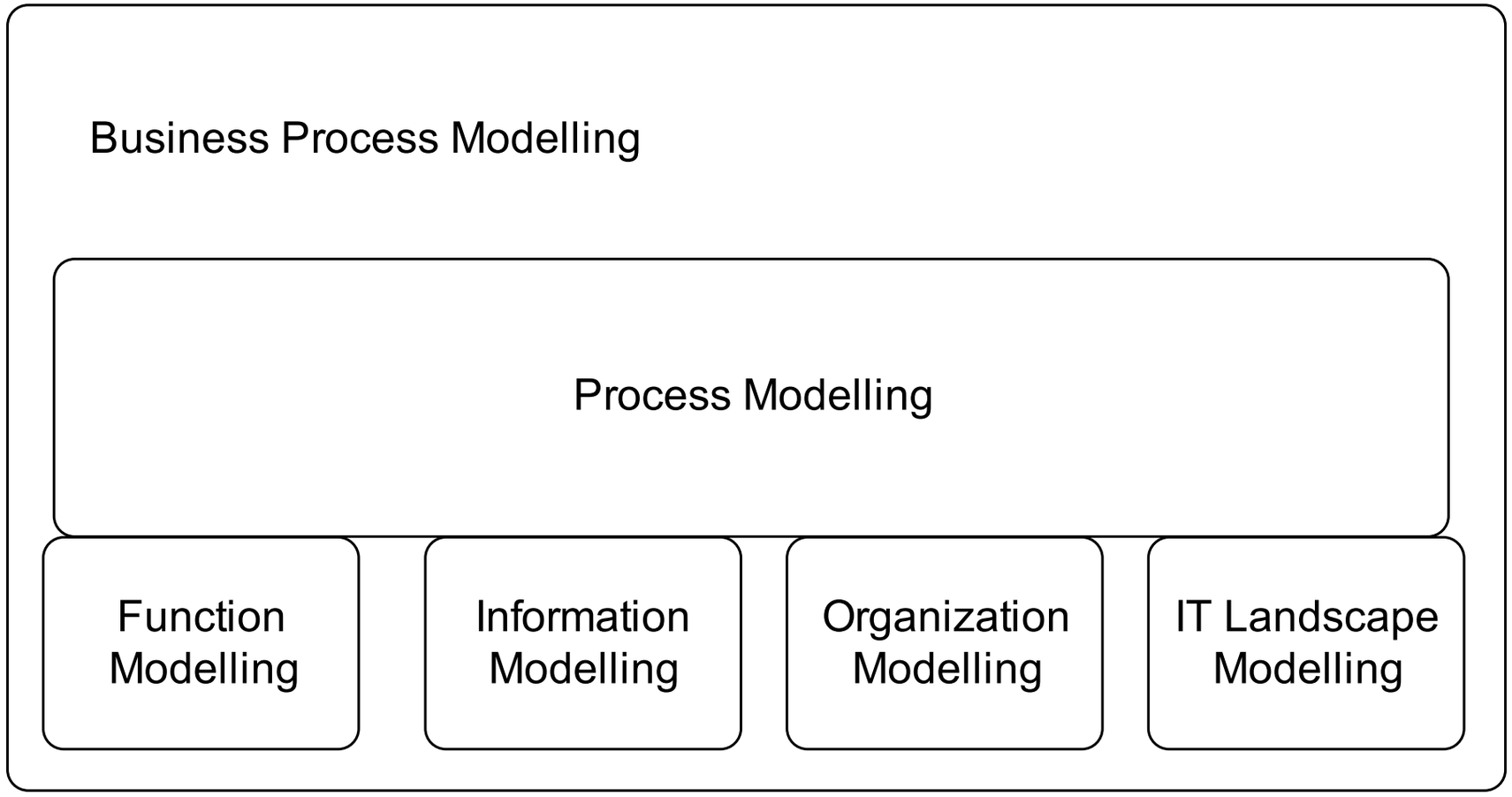
A sample aggregation



Vertical Abstraction

Vertical abstraction (domain separation)

BPM includes multiple modelling domains, integrated by Process Modelling

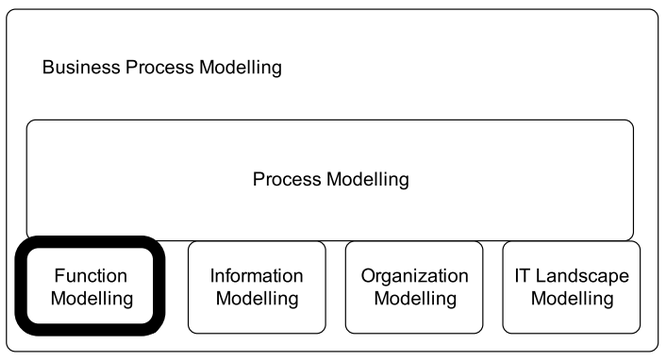


Function models

Units of work enacted by processes
(at different levels of granularity)

Informal description, textual documents
(coarse-grain business level)

Formal description, function specifications
(fine-grain software layer)



Value Chains

Value chains are a way to organize the work that a company conducts to achieve its business goal

Value chains were developed by Michael Porter to organize high-level business functions and to relate them to each other

Value systems

Companies have goals to fulfill

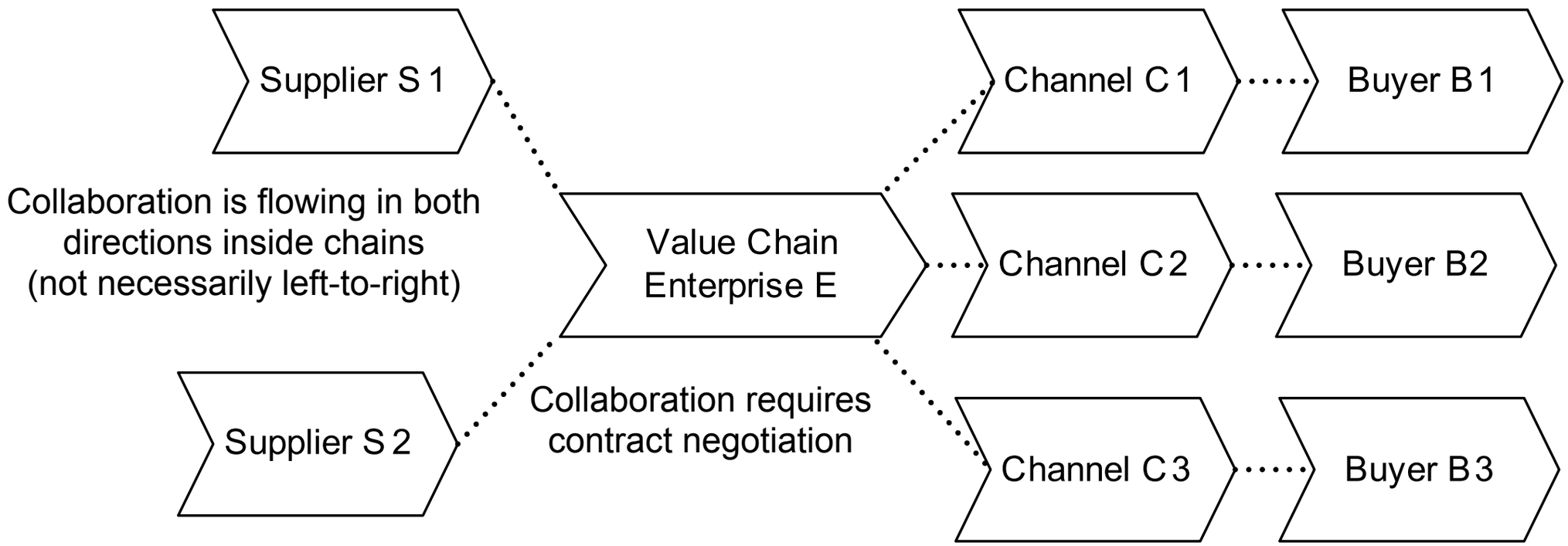
To reach their goals,
companies can cooperate with each other

The value chains of cooperating companies
become linked/related to each other:
they form a **value system**

Value systems can provide an immediate
understanding of ``how a company operates''

Value systems

Informal, high-level business functions decomposition
produce a
Value system
made of
Value chains
centred at the enterprise E under consideration



Ecology of value chains

``*gaining and sustaining competitive advantage depends on understanding not only a firm's value chain but how the firm fits in the overall value system*''

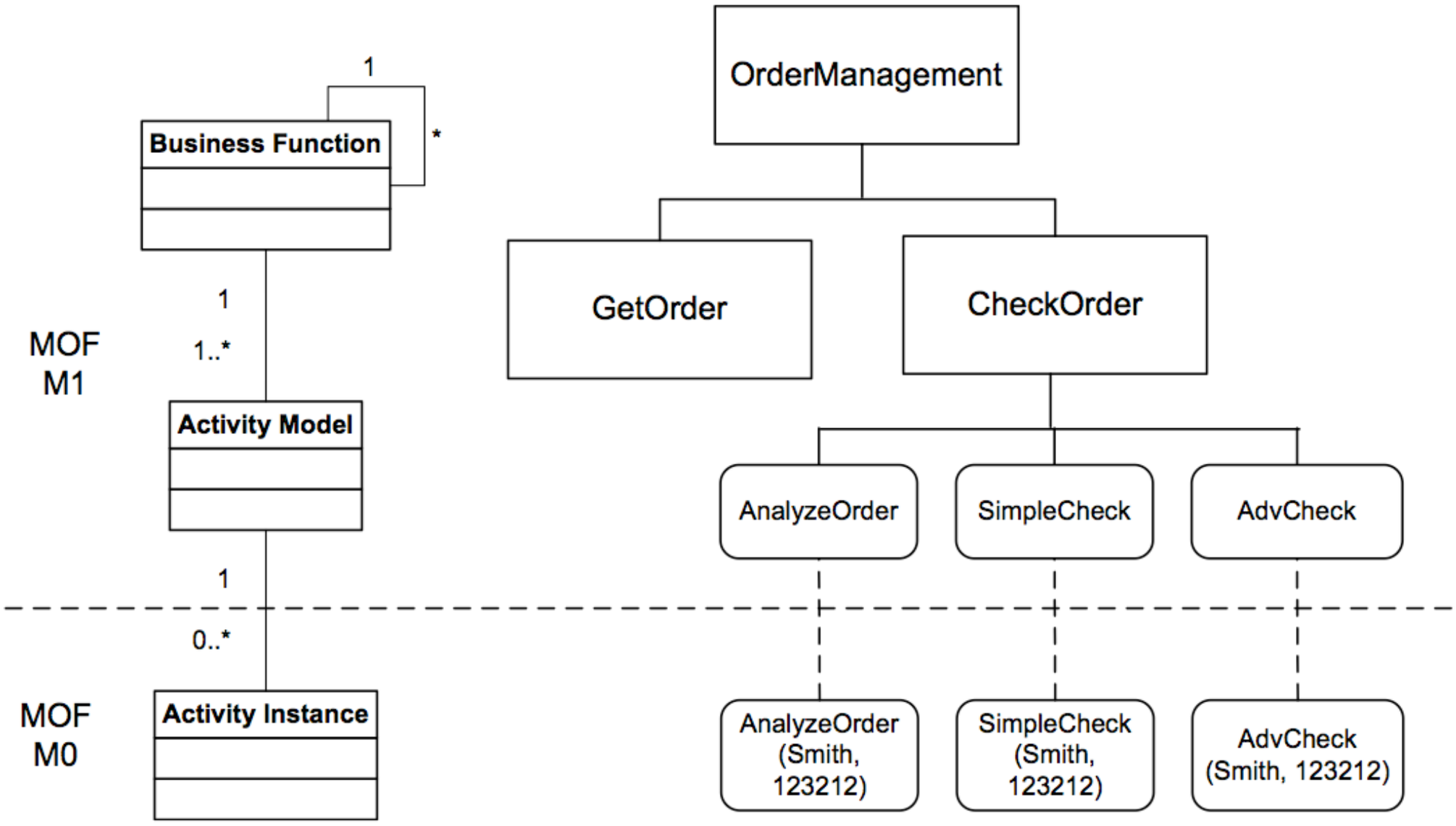
- Porter

High-level business functions

The value chain of a company has a rich internal structure, consisting of a set of coarse-grained business functions
(e.g. Order management, Human resources)

High-level business functions can be decomposed into finer-grained functions
(this is called **functional decomposition**)
(e.g. from ``Order management'' to ``storing'' and ``checking'' orders)

Activity models and activity instances



Value chains and processes

Porter was not able to identify the role of processes within value chains

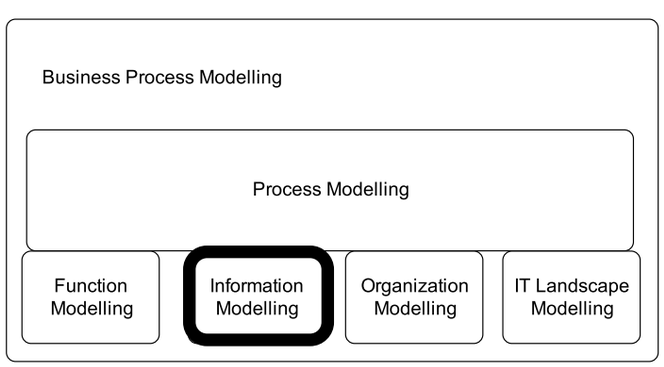
However, process-orientation can fit very well with value-chains and functional decomposition (see Lecture 4)

Key factor:
the granularity of business processes must be in line with the particular goals associated with the supported business function

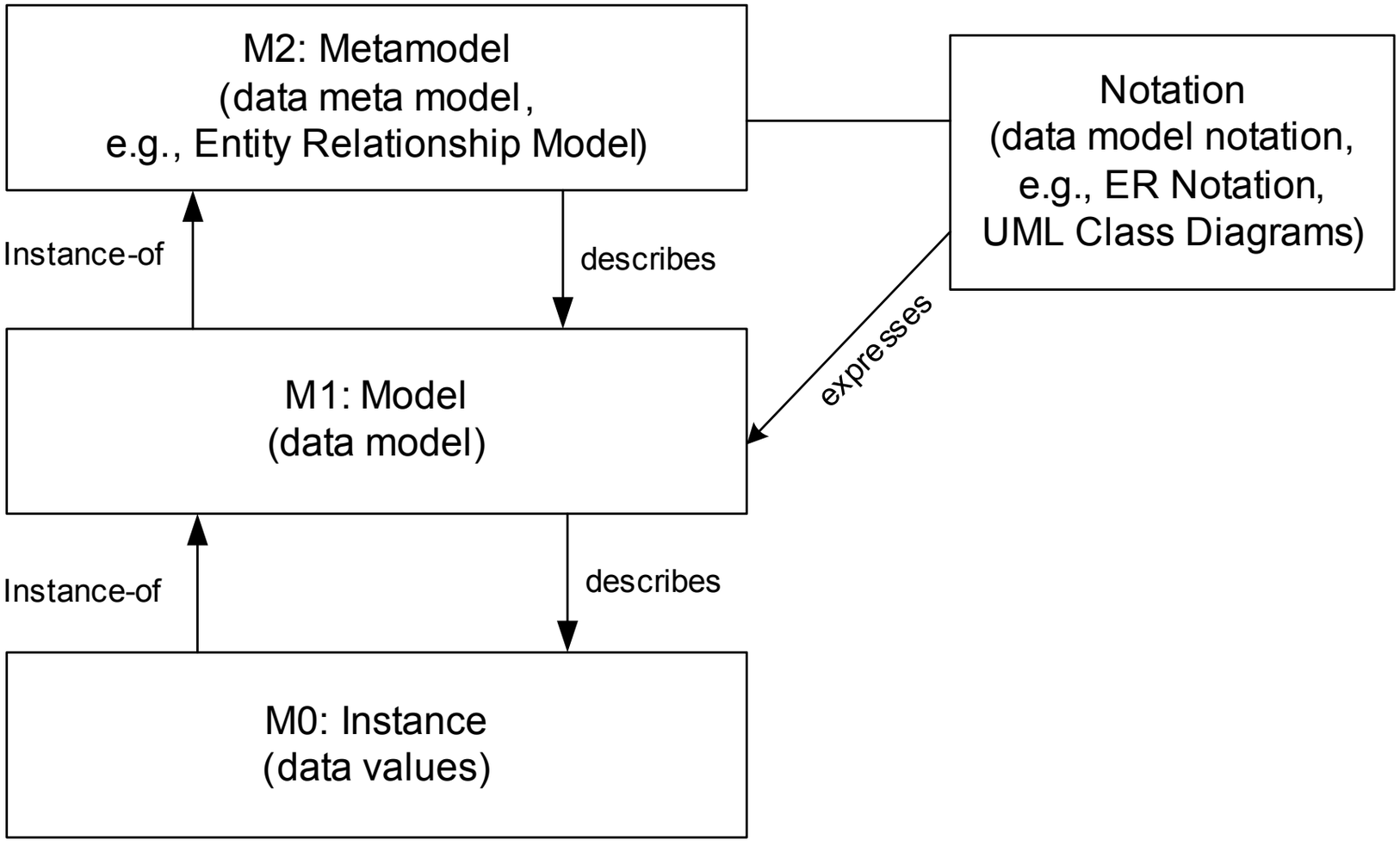
Information models

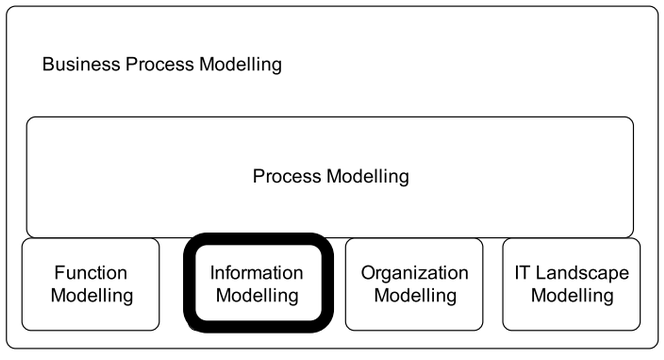
Data representation is crucial:
all decisions made during a business process
depends on data values

Data dependencies between activities
are also important
(ensure data-availability, reduce waiting time)



Data models



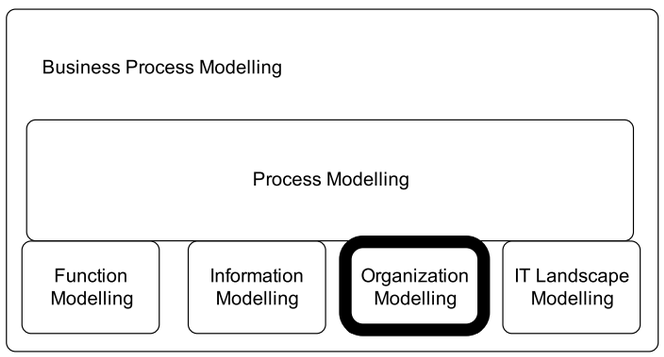


Organizational models

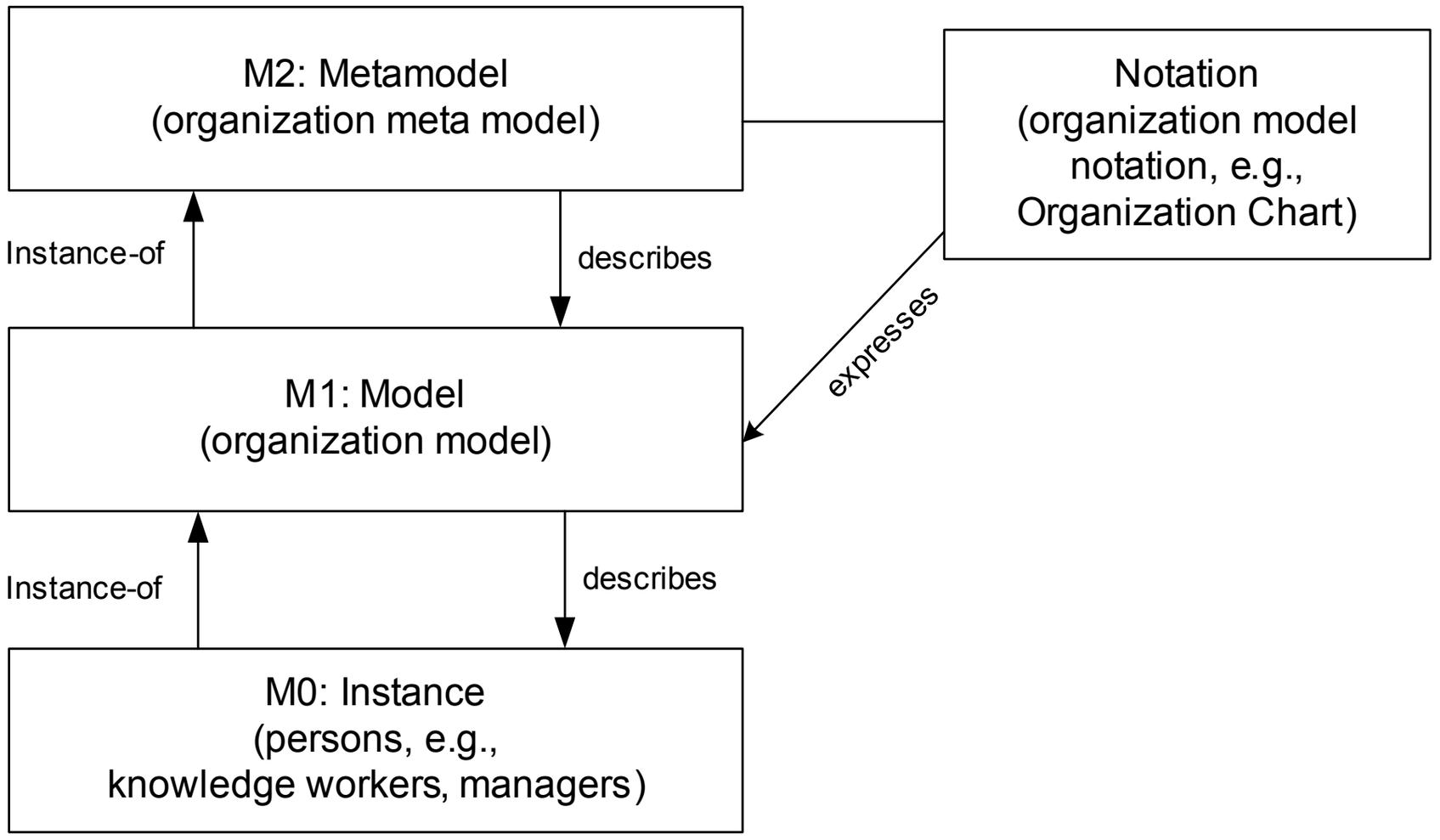
Organizational structure must be represented

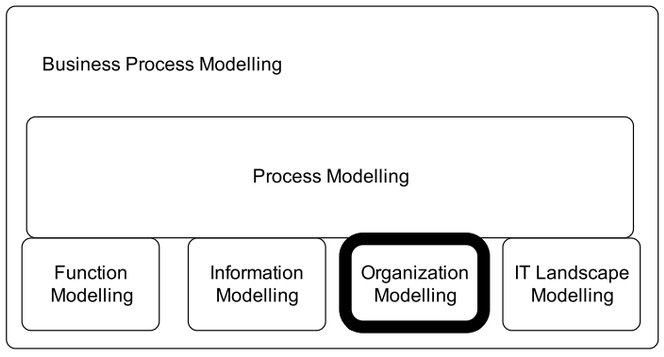
Activities must be associated

to specific roles or departments

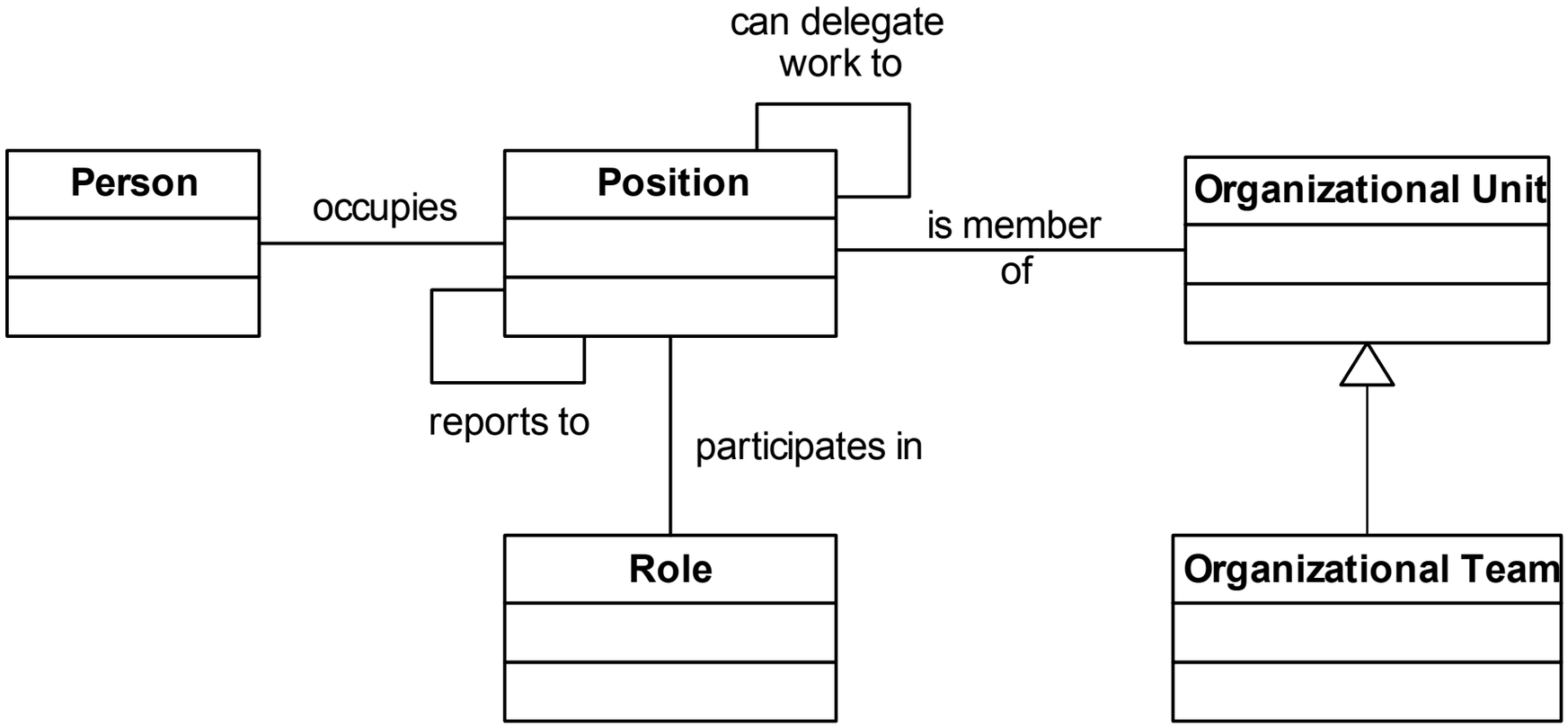


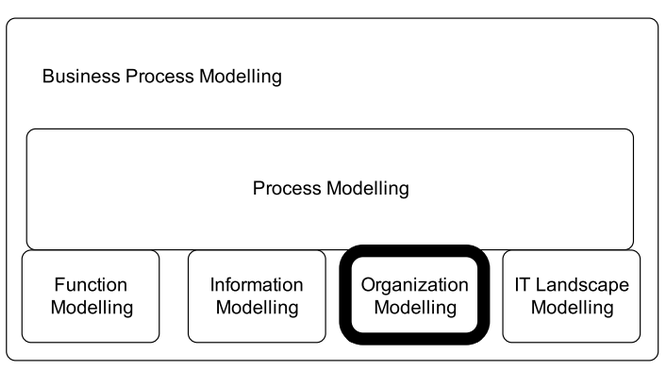
Organizational models





An organizational metamodel

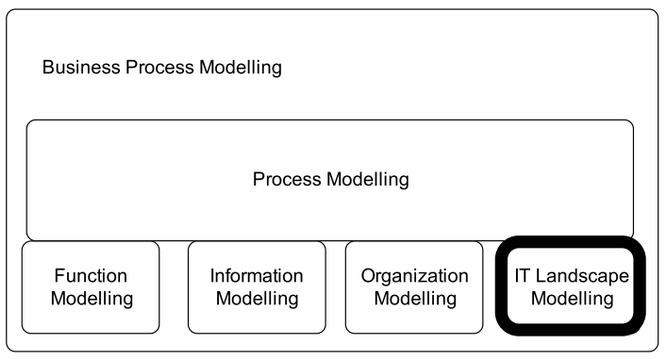




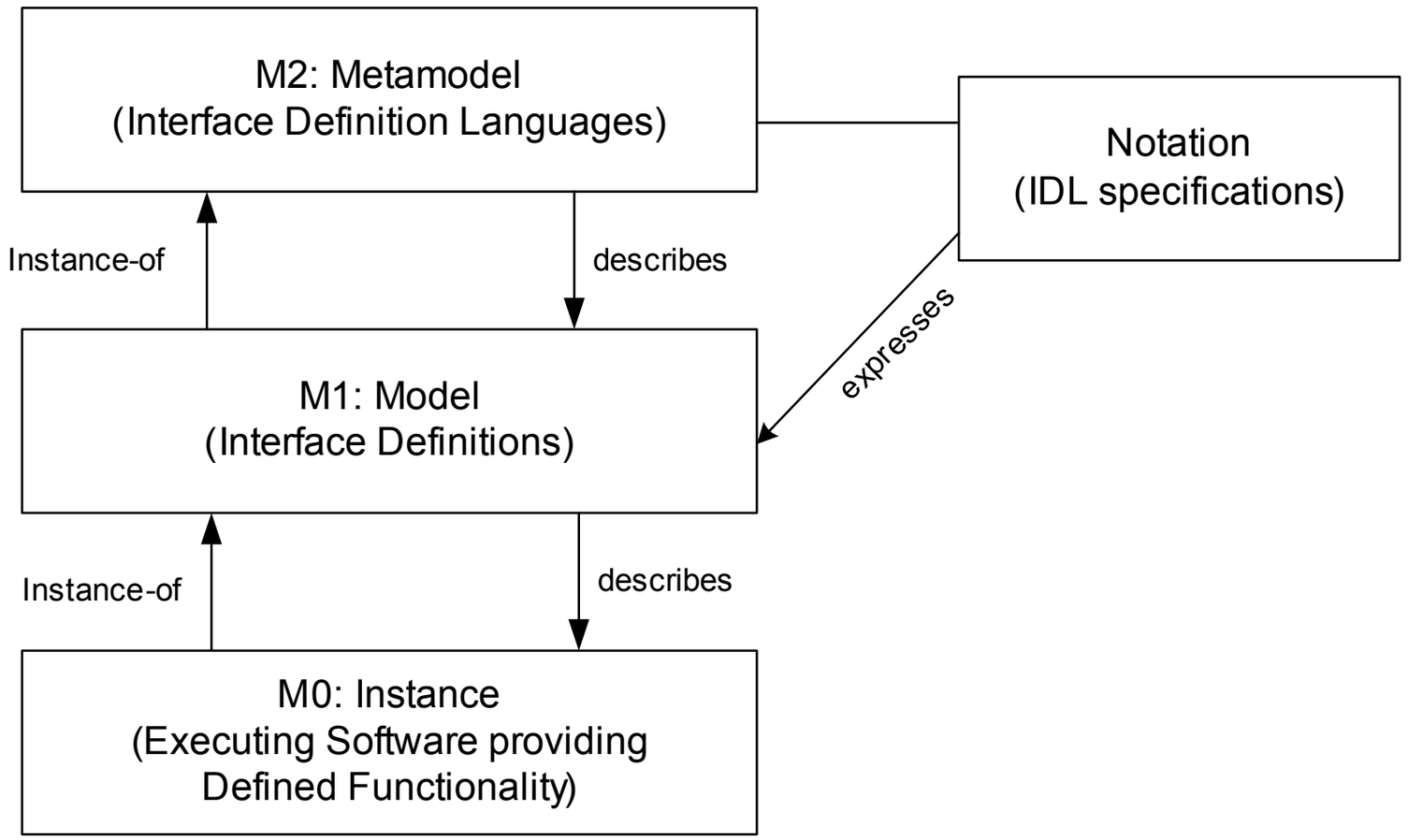
IT landscape

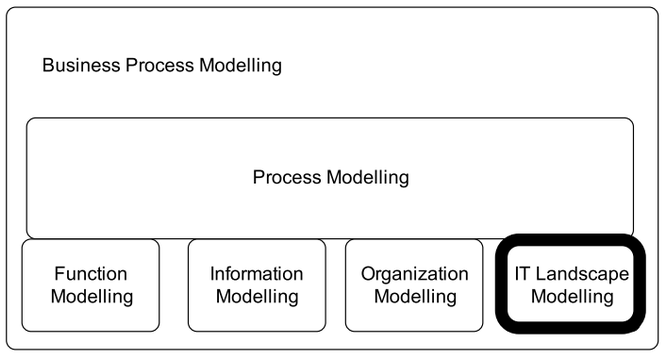
Many activities in a business process are supported by information systems

Information systems and programming interfaces needs to be represented because they provide functionalities



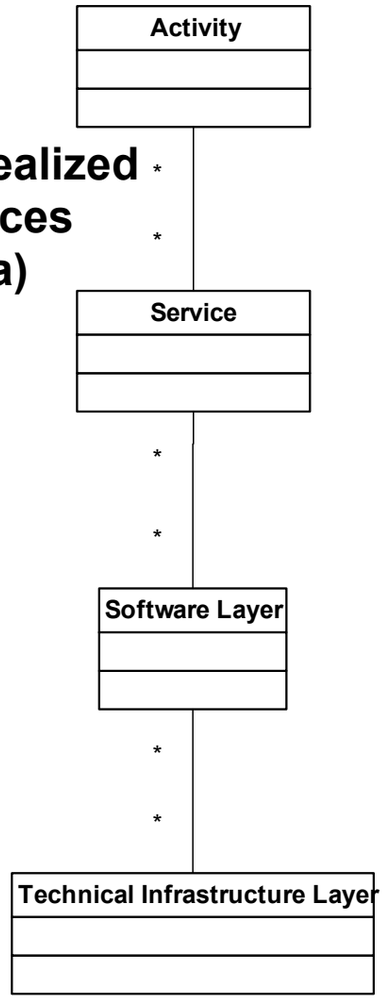
Interface Definition Languages

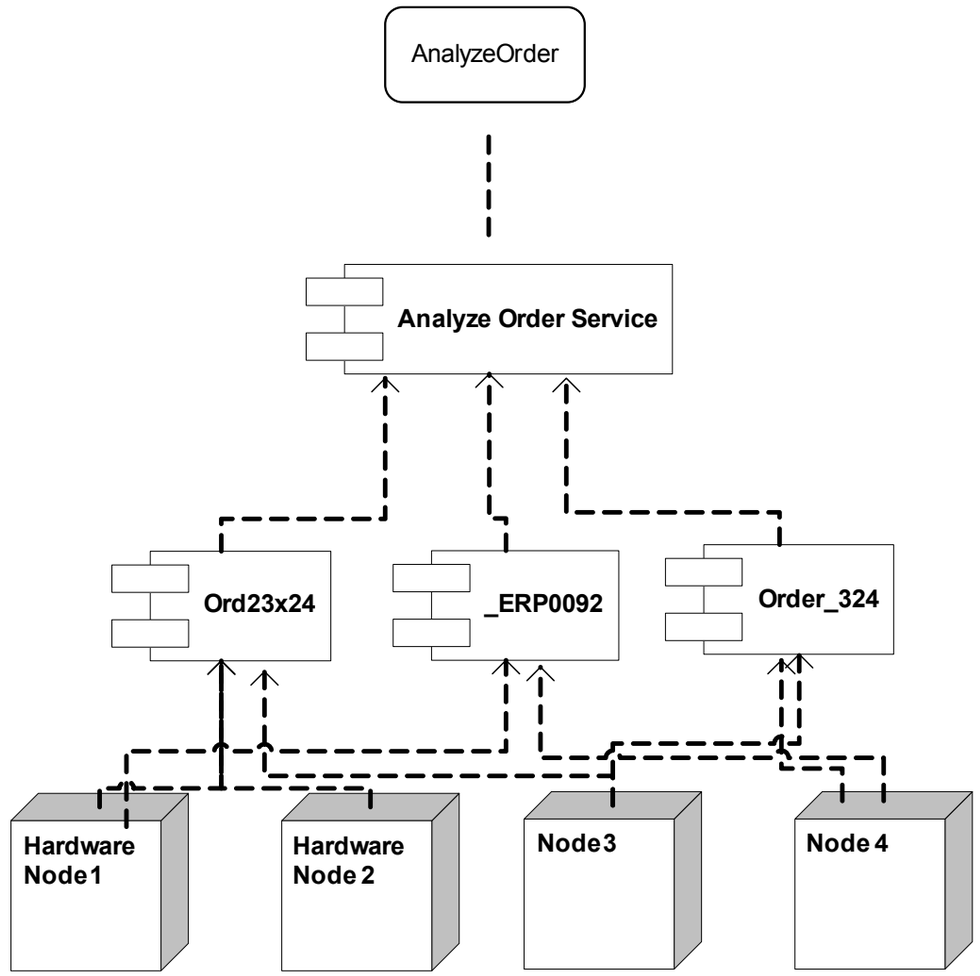


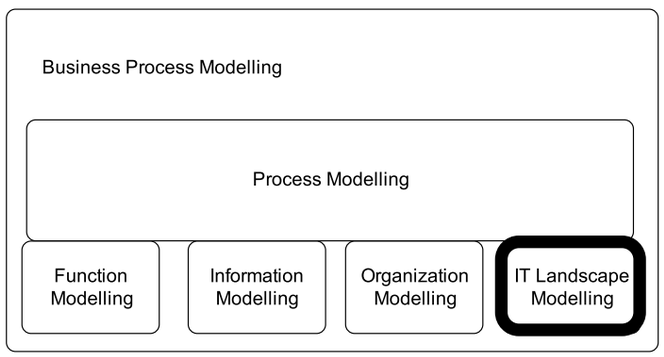


Service enabling

An activity can be realized by multiple services (and vice versa)





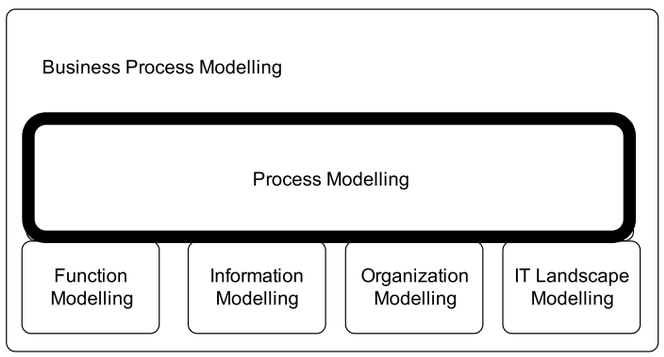


Process models

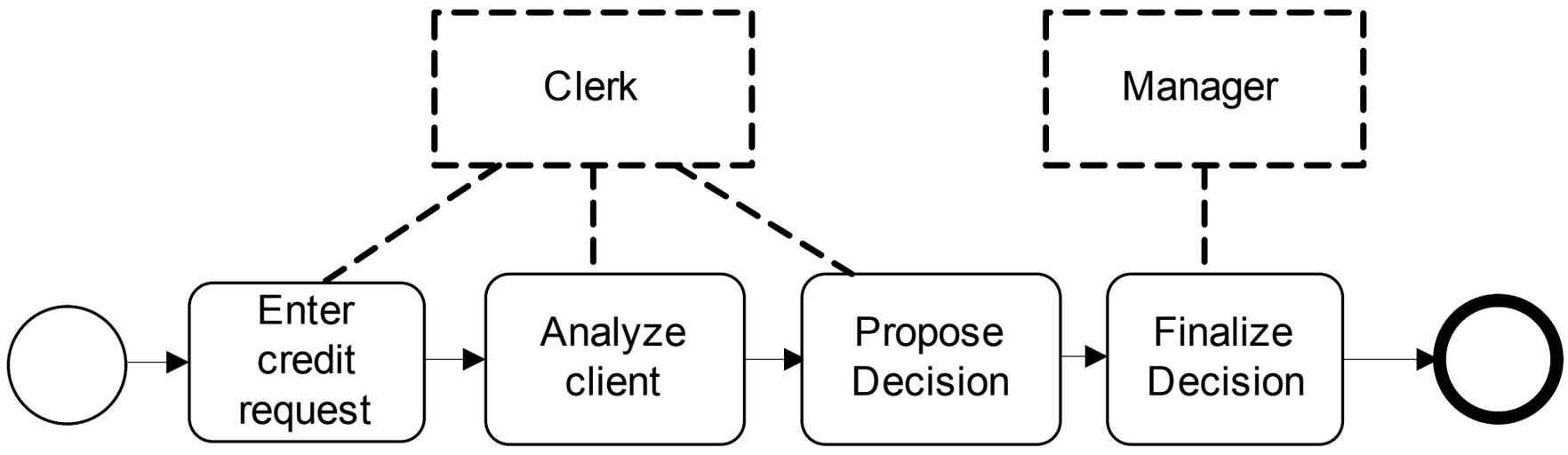
Define the glue between the subdomains

Relate functions and execution constraints

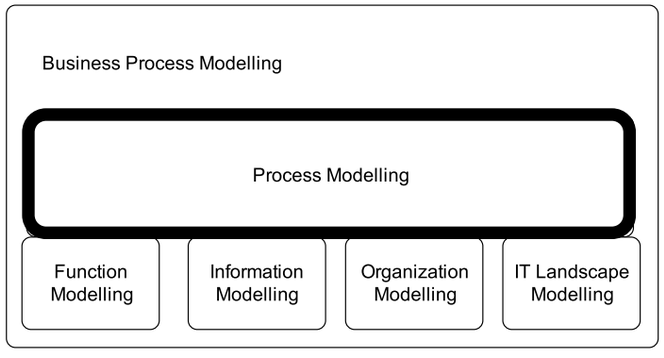
Relate data values with process instances
(e.g. the process of a credit approval may depend on the requested amount)



A process model with role information



M. Weske: Business Process Management,
© Springer-Verlag Berlin Heidelberg 2007



A process instance with workers information