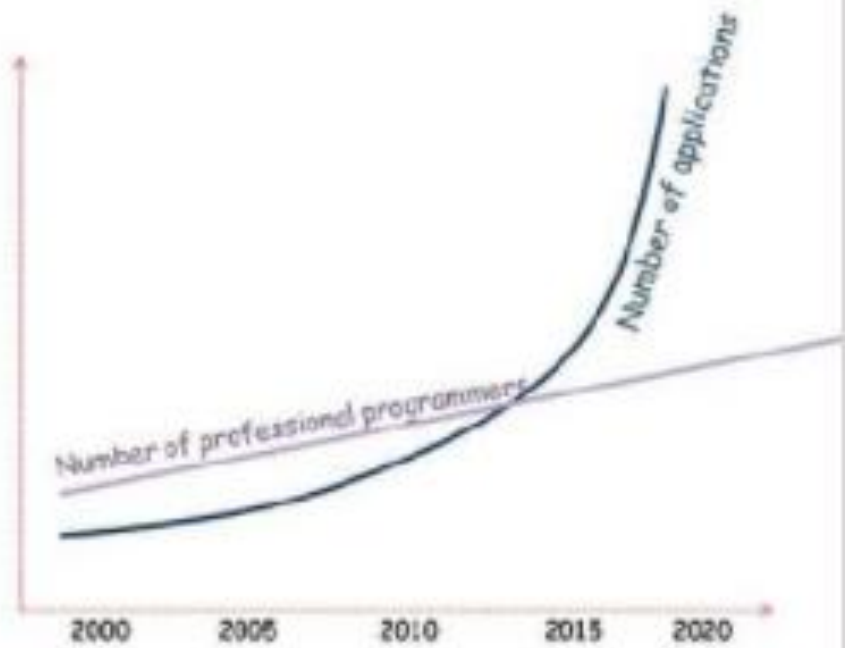
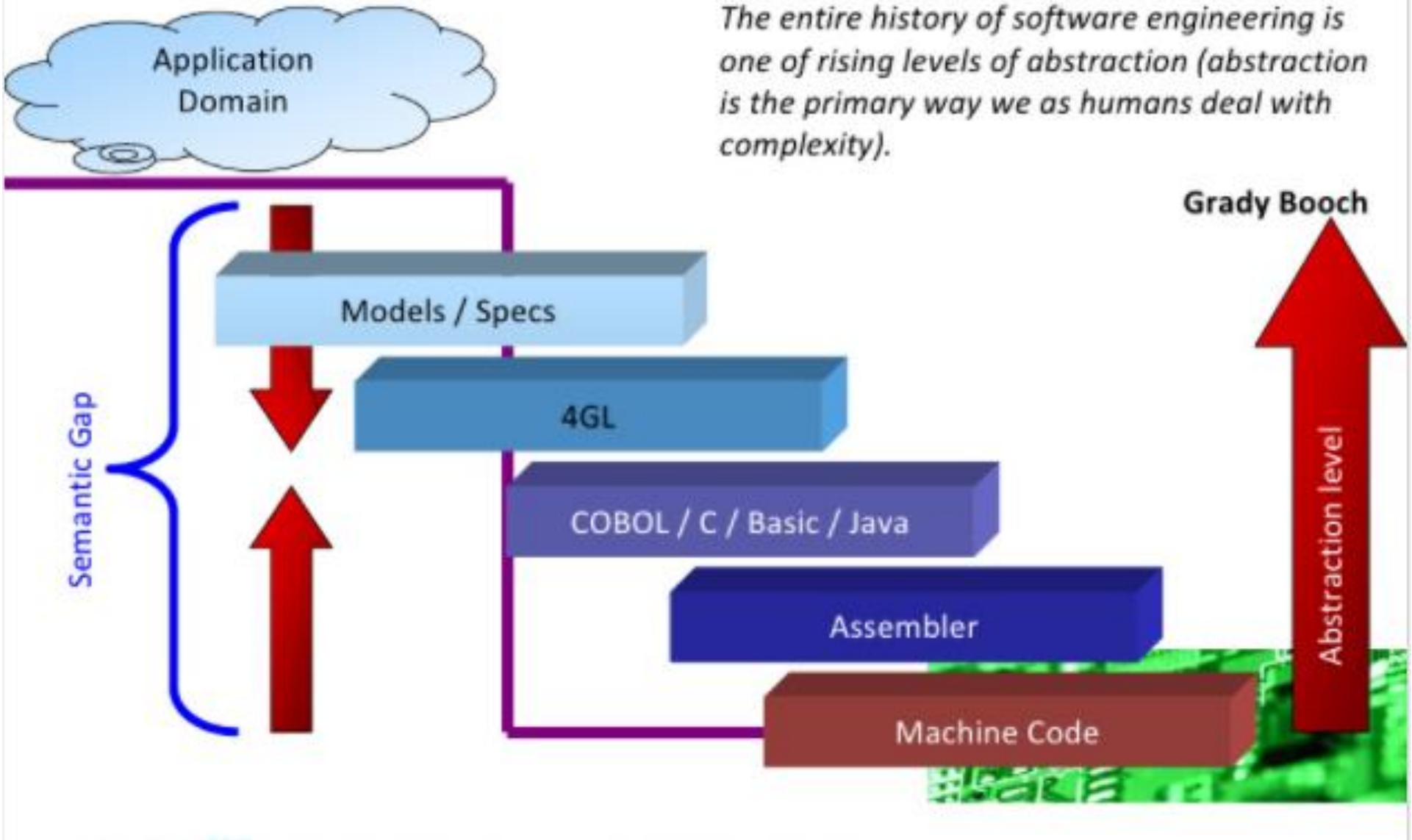
Professional programmers needed!

- The impossible equation (Jean Bézivin)
  1. Slow increase of professionals
     - Same trend in next 25 years
  2. Rapid increase of SW to be developed
  3. No way to cope with it
  4. End User Programming
     - Example: Visical, Excel
     - By DSL, abstraction
     - Custom languages



Abstraction Levels

The entire history of software engineering is one of rising levels of abstraction (abstraction is the primary way we as humans deal with complexity).

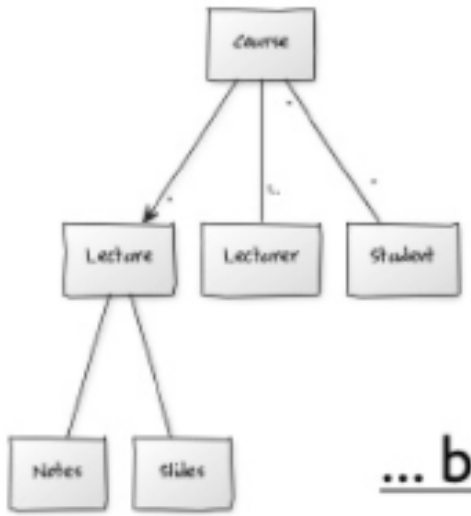


What is a model?

A model

- is a simplification of a system
  - ★ abstraction, description, specification, information
- can answer questions in place of actual system
  - ★ analysis, inference, predictions
- is used for a purpose
  - ★ understanding, planing, risk analysis, ...

A model can be a UML diagram ...



... but it can be any other representation ...

$$e = x \mid e + e \mid e - e \mid f(e,\ldots,e)$$

... that serves purpose of abstraction, analysis, etc.

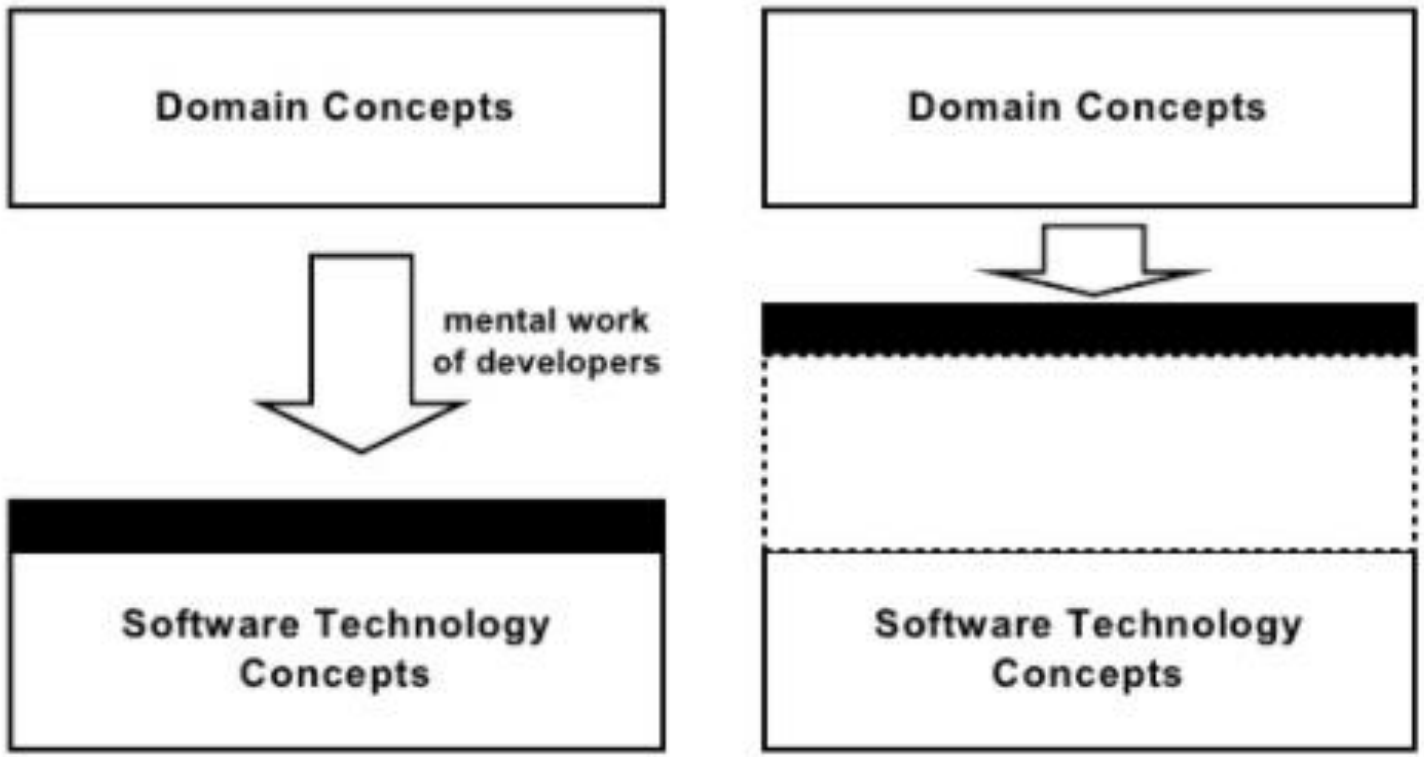
MDD

- *Model Driven Development*

Definition:

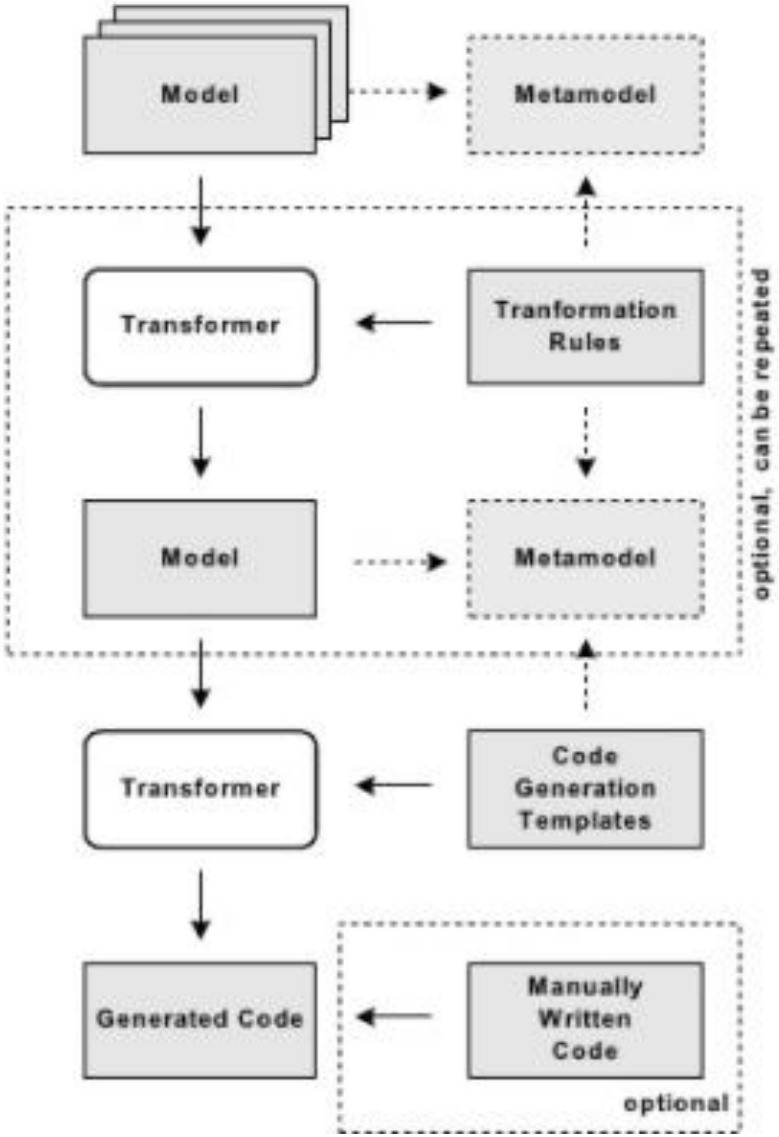
The usage of Models as the main artefacts to Drive the software Development.

- Model Driven Development is about making software development more **domain-related** as opposed to **computing related**. It is also about making software development in a certain domain **more efficient**.



How MDSD works

- Developer develops **model(s)** based on certain metamodel(s), expressed using a DSL.
- Using **code generation templates**, the model is transformed to executable code.
  - Alternative: Interpretation
- Optionally, the **generated code is merged** with manually written code.
- One or more **model-to-model transformation steps** may precede code generation.



Core Values

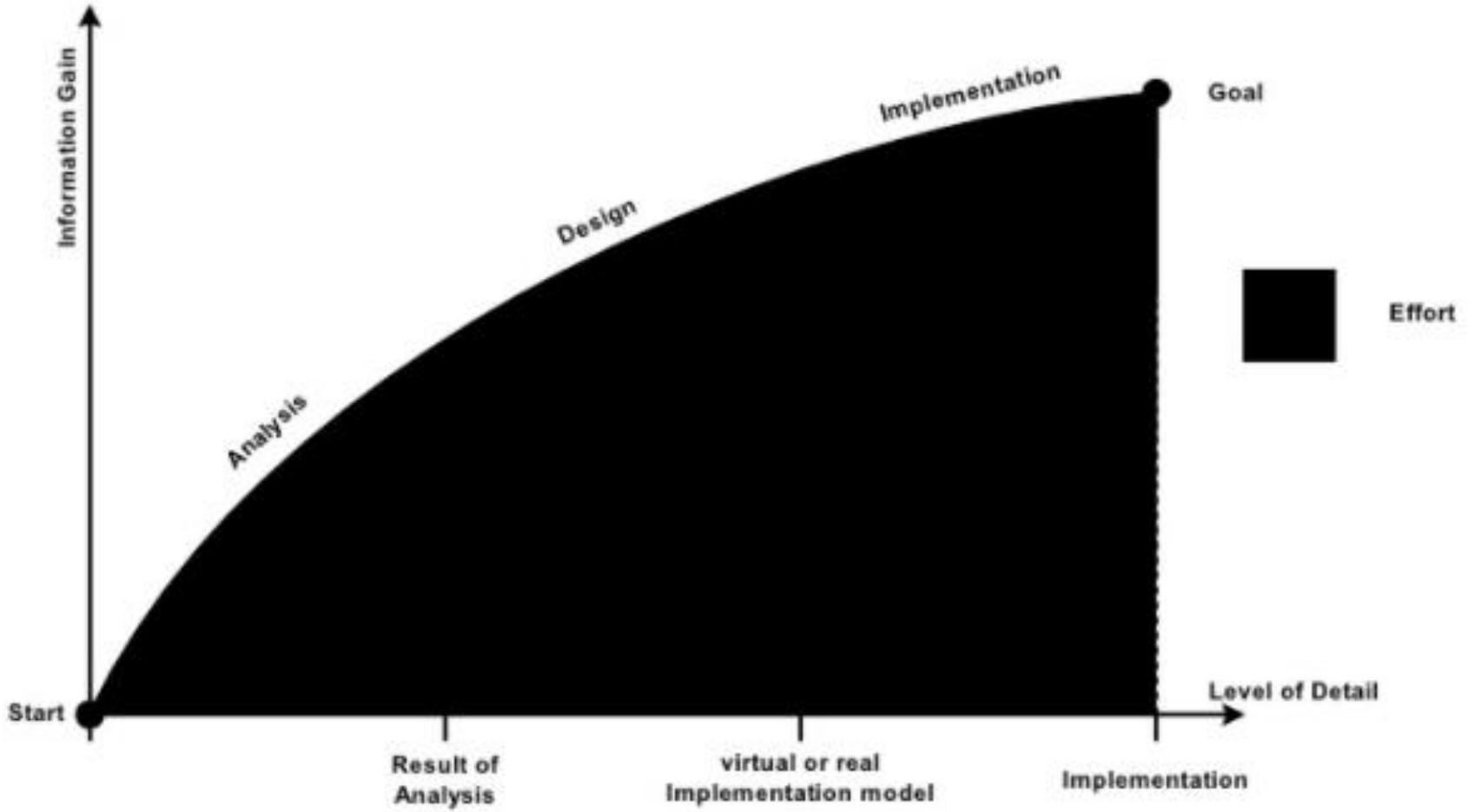
- We prefer to validate **software-under-construction** over validating software requirements
- We work with **domain-specific assets**, which can be anything from models, components, frameworks, generators, to languages and techniques.
- We strive to **automate software construction** from domain models; therefore we consciously distinguish between building software factories and building software applications
- We support the **emergence of supply chains for software development**, which implies domain-specific specialization and enables mass customization

Code Generation

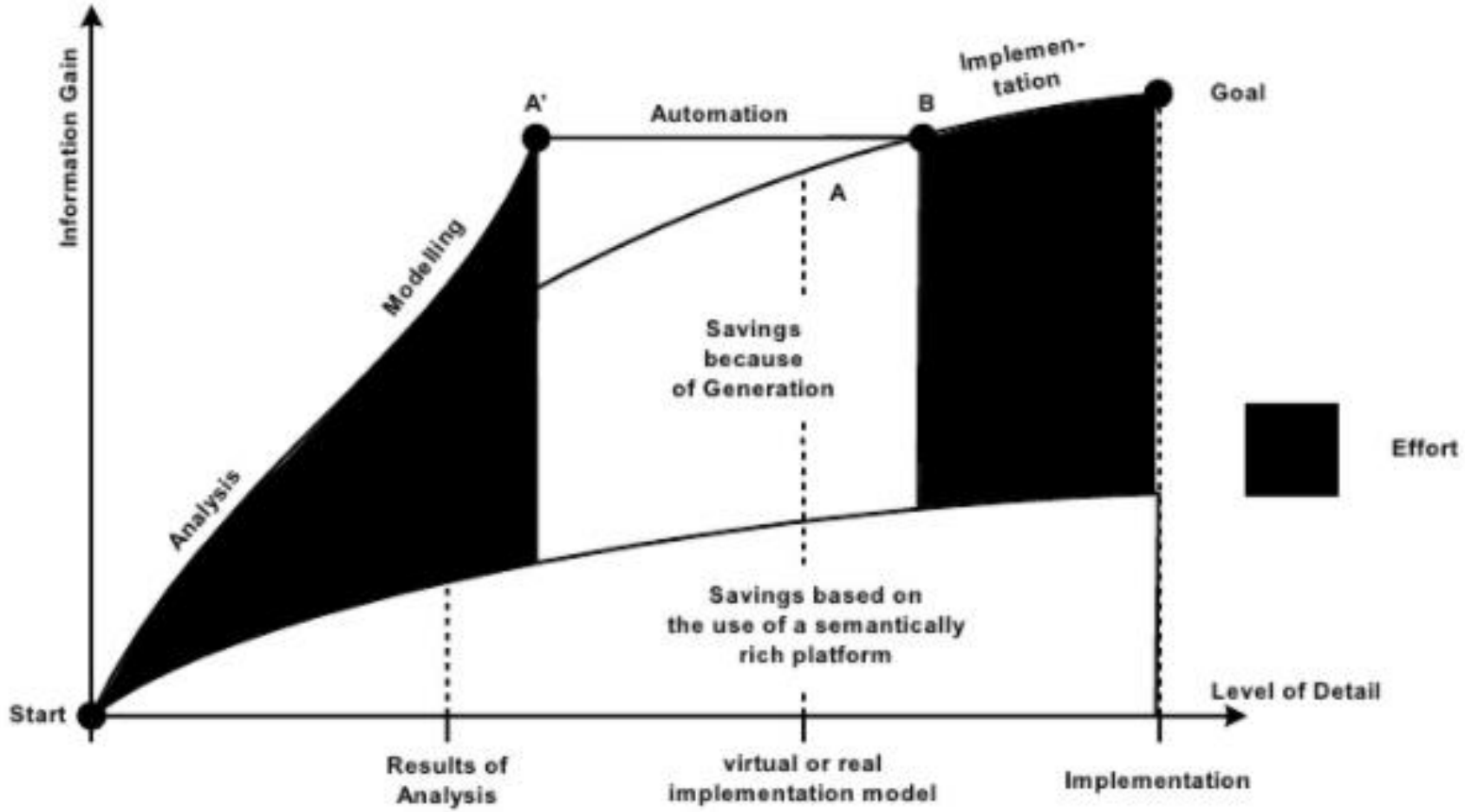
Definition:

The automated synthesis of SW assets like source code, documentation or models using models as input.

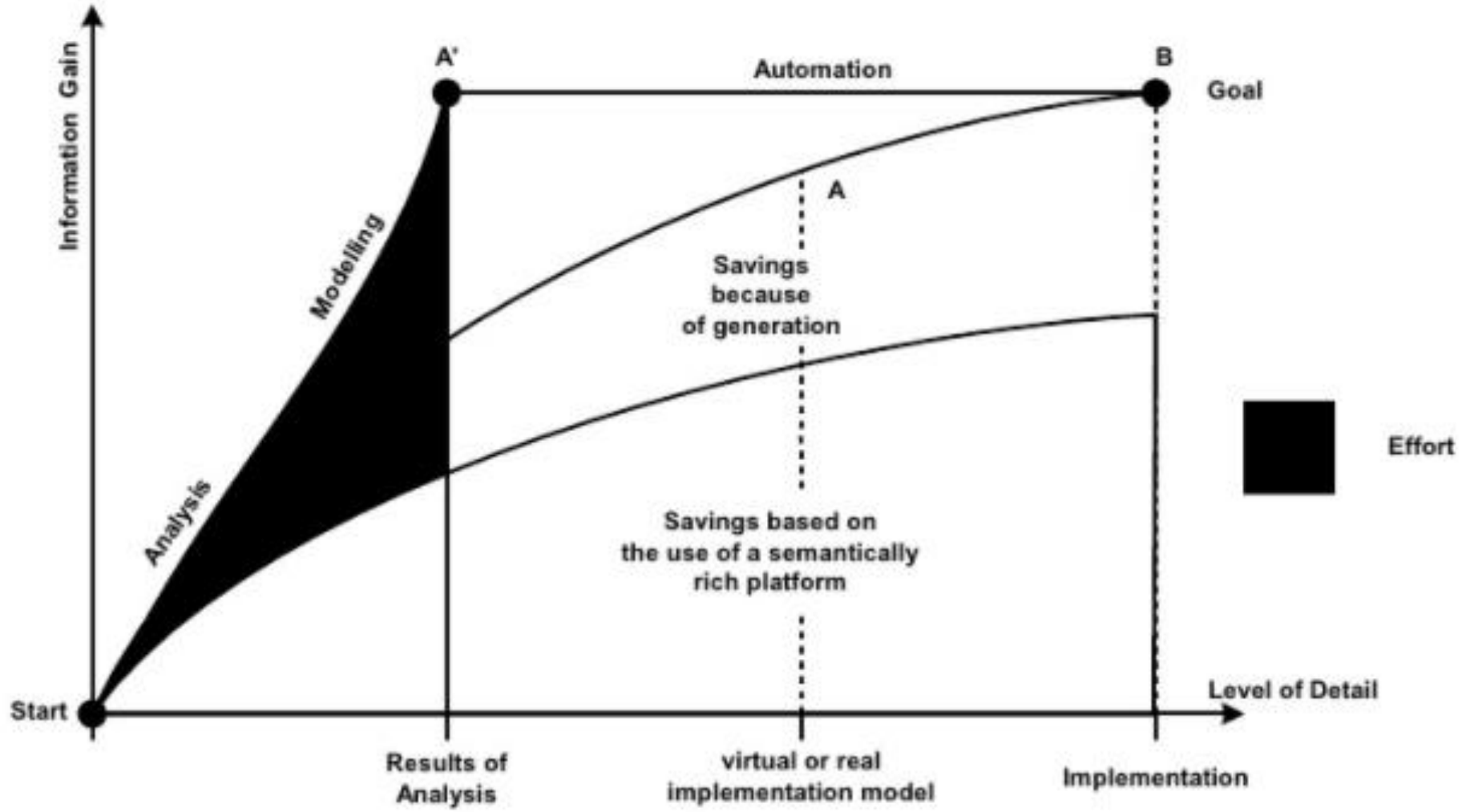
MDSD (compared to "normal" Software Development)



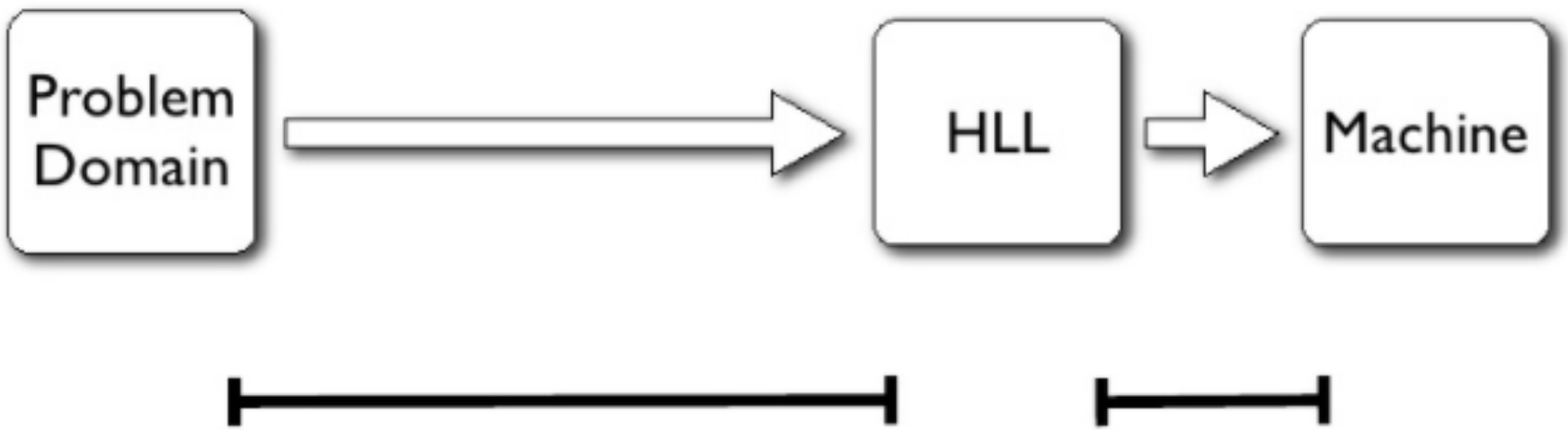
MDSD Effort (stage 1)



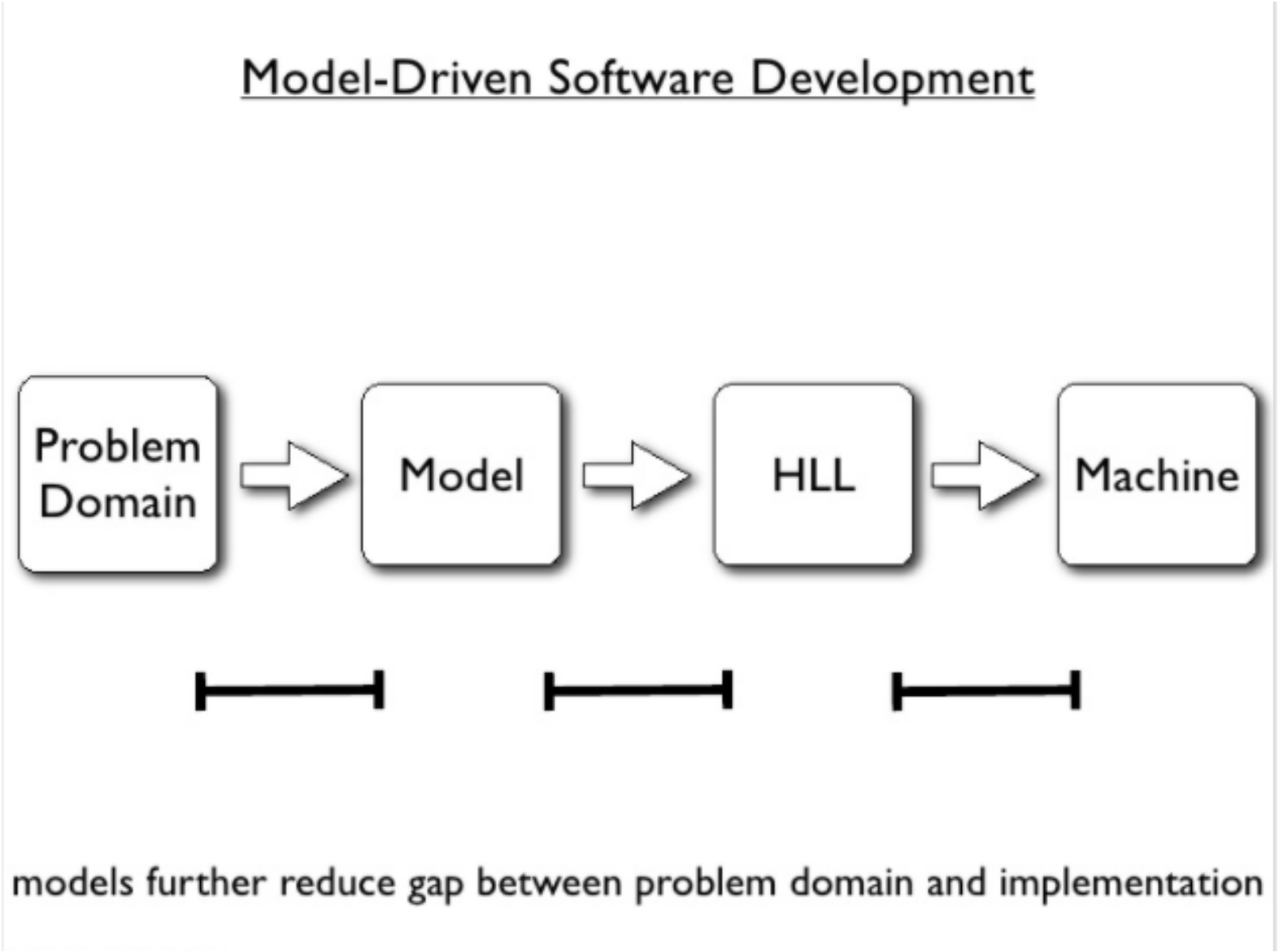
MDSD Effort (stage 2)



High-level languages reduce problem/solution gap



Next level of abstraction



Model-Driven Architecture (MDA)

Vision from OMG

- Models at different level of abstraction
  - ★ Platform Independent Model (PIM)
  - ★ Platform Specific Model (PSM)
- Model transformation
  - ★ e.g. PIM to PSM to implementation
  - ★ transformations not necessarily automatic
- UML as standard modeling language
  - ★ models are 'visual' or 'graphical'

Types of Models

- Platform-Independent Model (PIM)
  - Describes the business concerns of an application in a technology-agnostic way
  - Bridges between problem and solution spaces
  - Holds true regardless of what technology is used to implement the system

- Platform-Specific Model (PSM)
  - Describes how the PIM will be realized using a specific set of implementation technologies (e.g. Java, .NET, Ruby, Objective-C, etc.)
  - Describes the implementation mechanisms of a system
  - PIM can be implemented using any number of PSMs