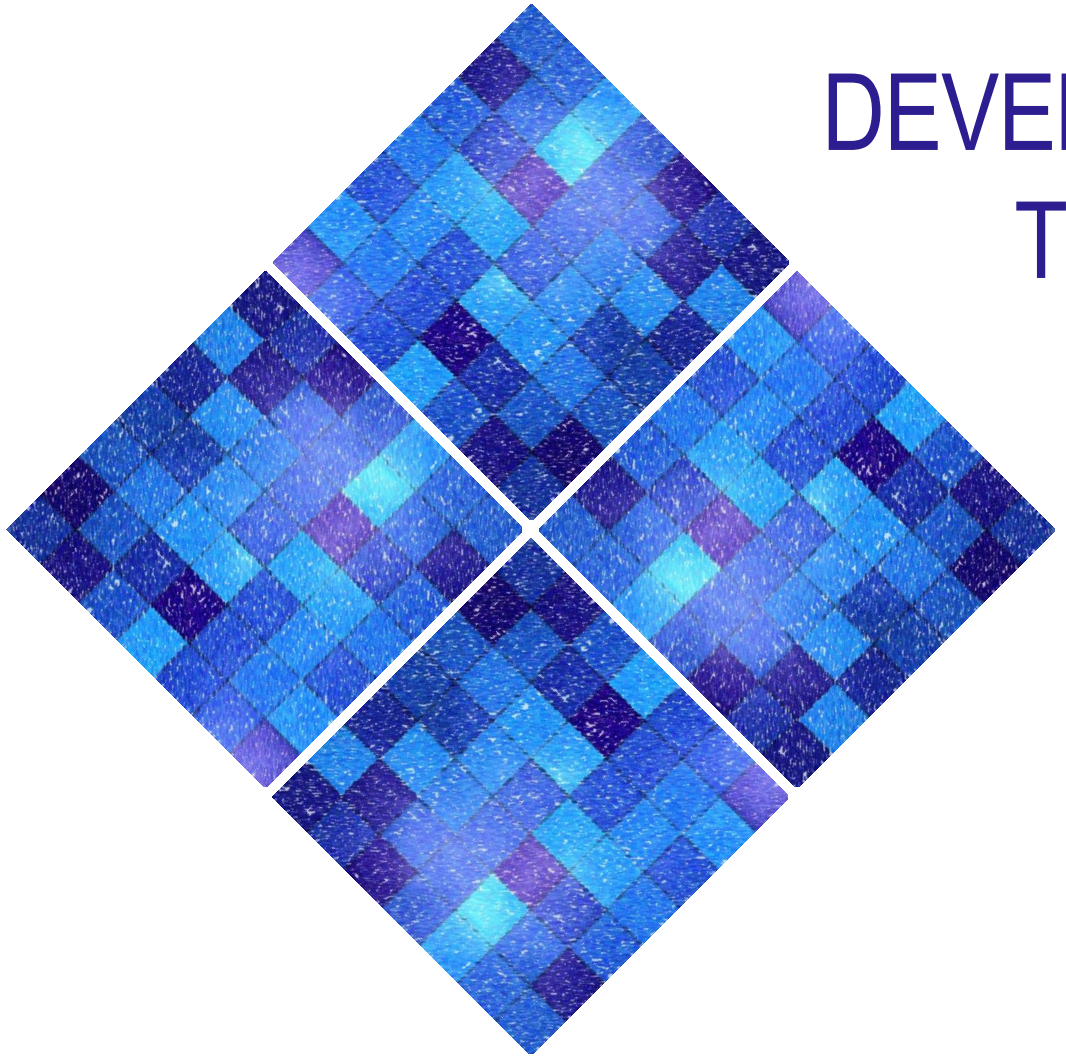


DEVELOPING SOFTWARE THAT UNDERSTANDS THE REAL WORLD

Dimitri Lyras,
Director of Lyras Shipping,
Founder of Ulysses Systems

Computers and the real world

Software today is mostly geared towards developing quick and cost efficient solutions

The result is a focus on developing software languages that emphasize passing commands to machines as expediently as possible

Less attention is given to teaching computers about the world they will be operating in and more given to building programs

Computers and the real world

Why shouldn't maximizing the efficiency of programming,
which is mostly a method for passing commands to computers,
be the only focus?

Why should we care about teaching computers about the real world
beyond traditional programming?

Command Oriented software development

Because passing commands to computers combined with describing real world situations, makes us mix two different skills and toolsets

Yet, there are plenty of ways to teach computers the situations they will be working in and to ensure they are designed to behave desirably now and in the future

However we are still focusing on technology that makes modelling the world harder and writing software easier

Searching for Stored Information


```
SELECT e.ID, e.prisonerID, e.prisonID
FROM tblPRIexperience e
INNER JOIN tblPRIprisoner PR ON PR.ID = e.prisonerID
INNER JOIN tblPRISON P ON P.ID = e.prisonID
WHERE e.PrisonID IN ('D3A71B4C-52BF-40AA-A4B6-212C60DD88F8', '26A56662-
B431-4E61-B95F-8F61BE6379B2', 'B87F5C5D-173B-45F0-B8D0-9BB44D2B775E')
AND e.PrisonerID = '66C6E52D-3975-4EDE-946E-50C6A2285BE5‘
AND e.StartDate > 1962
```

How will either party know which of millions of memory retrievals actually answers the unspoken questions:
“How did you feel? What contributed to your resilience? What gave you the strength?”

In their heads, people model all the time

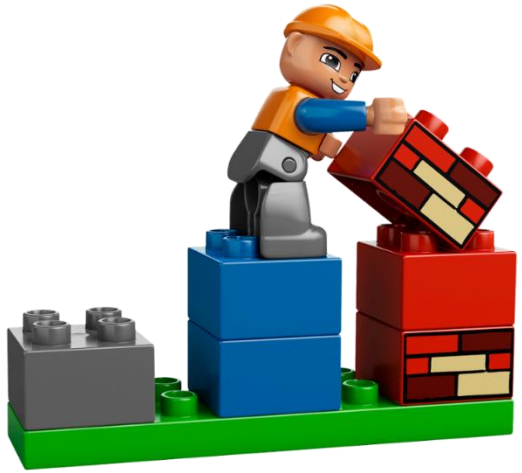
When you enter a room and recognize an individual, you know whether to greet them or not and what sort of greeting is appropriate

This is because people have a template or blueprint, known in psychology as a schema, of what to expect, and what is expected of one in different situations

Furthermore, the schema is not static; changes and additions can and are applied as we explore a situation further

You are modelling all the time

In other words, people make, maintain and expand models as they go along



Whether going to a restaurant, eating at home or at a work function, we have a picture of what those scenarios are going to be like, what we and other individuals need to do as well as what to anticipate from the situation

Comprehension of the real world

For example a person's age has multiple uses in society, therefore this data needs to inform the activities it is relevant to. How can this occur without knowing how the age factor affects or modifies these activities?

If one is to assume that simply knowing someone's age can be utilized by anther party without a model in which this propagates, wouldn't young children know what the significance of their age is? A question like "'why are regular vision tests required to renew drivers licenses after age 60?" would be answerable by 3 year old.

However, children do not inherently know how age affects the world. They need to build up a model of the world.

Understanding the world

Moreover, age is something that is required in many models; for example the right to vote, got to school, join the military, engage in recreational activities, drive without supervision, rightful inheritance and even what behavior society deems appropriate. All are complex models involving biological, psychological, and other factors affected by one's age. This is why we teach children particulars and how these factors and age are significant to many processes.

That we need to teach children such norms makes it clear that facts like age need to be built into models of the world with cause and effect. In order for a software system to infer what to say to whom and when, requires the compute to have awareness of these same associations.

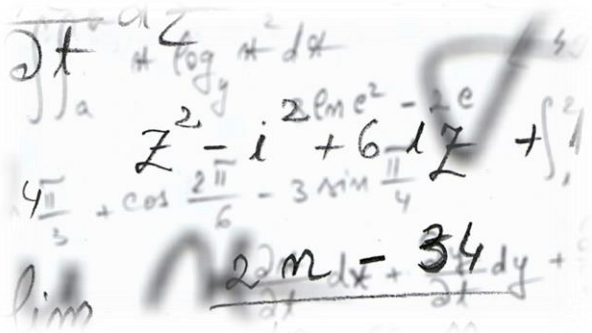
Understanding the world

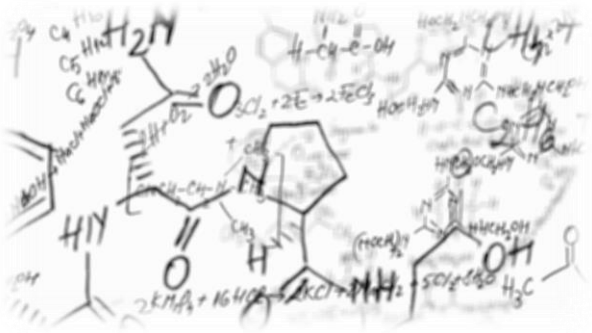
However, there is a huge gulf between a computer understanding the world it operates within and a computer understanding commands in day to day software development

Of course, we have data models and rules that try to emulate the world. However, the structure of data models such as UML, are primarily aimed at making it easy for developers to program computers and avoid duplicated data. Allot less work has gone into providing structures for describing the world, such as ‘context’

Describing the real world

Modelling the real world for computers can in certain scenarios approach the complexity found in physics discovery

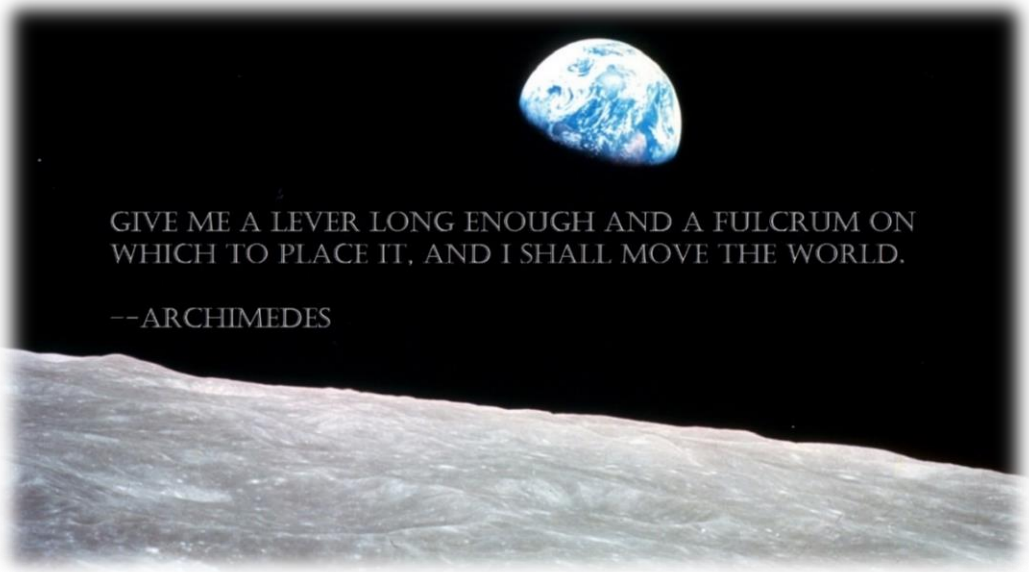




However, usually it is much simpler and more akin to mathematical models that expose relevance between things that exist, that are used, quantified and qualified in the real world

Describing the real world

Archimedes and Newton were not software developers or experts in passing commands to machines



They were experts in discovering and describing the relevance between phenomena in the real world: What are the components that make ‘that’ happen?

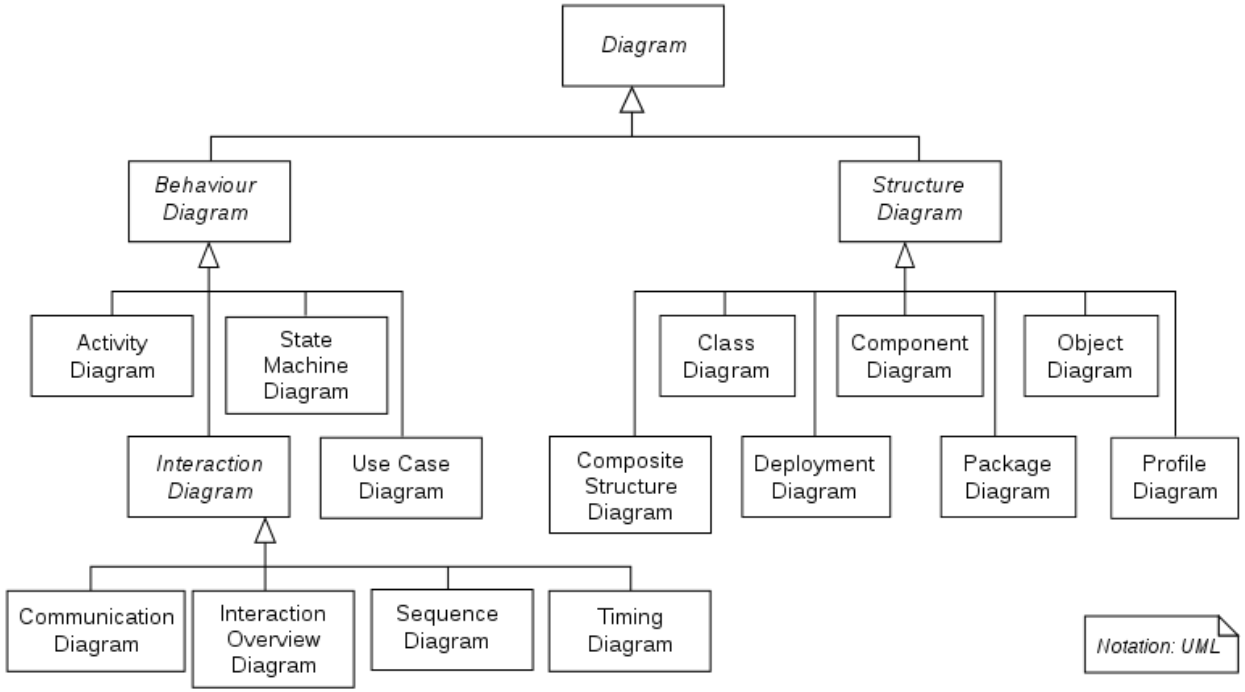
Simulating the real world

Archimedes and Newton did not rely on commands to develop their models. They described the world of new concepts and how these concepts interact with existing ones

They developed models for this in Physics and Mathematics to communicate this to domain experts. Modeling relationships in the real world is not the same as passing commands to a machine. Doing both together makes it hard for the domain expert and the software expert

Describing the real world

Do we hear of physicists using UML to describe physical phenomena?



Challenges in modelling the world

When creating a model, one needs to take multiple parameters into account

However, these parameters may not all be known when a model is begun, nor how they are related to each other and enterprise goals

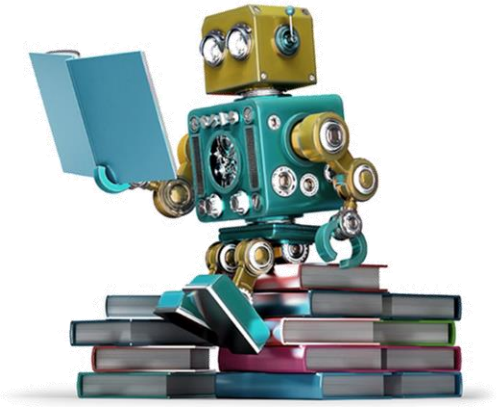
Therefore further relevance connections or attributes can be discovered in due course.

Trying to do too much at the design stage by way of attribute relationships severely restricts accurate simulation of an environment, sustainability and extensibility of systems

Machine Learning

So neither the designers nor the end users can get an intuitive grasp of what the computer has been taught via the data models and rules

Machine learning is the first attempt to address this. However by focusing on correlation it is often missing the rich contribution to understanding that cause and effect offers



When do you need modelling?

Whenever you are designing something; architects, mechanical and electrical engineers always use models

When you need to remember how you defined a set of relationships

When do you need models most?

When there are many exceptions to rules

When there are variations in processes

When there is a need to predict behavior in a variety of situations

When you need a lot of logic to be easily scrutinized by stakeholders

When complex logic needs to be maintained

When complex logic needs to be extended regularly

What type of models are there in general?

Existing modelling technologies
These models typically require conventional software to execute

Models for writing code more easily with less programming expertise (low code) as well as models to facilitate the understanding of relational databases and the querying of databases by end users

Conceptual models for managing complex real world issues such as software requirements and complex interactions such as prediction logic

What type of models are there in general?

Emerging modelling technologies:

Conceptual models that include the execution code
Conceptual models that can be converted to code automatically

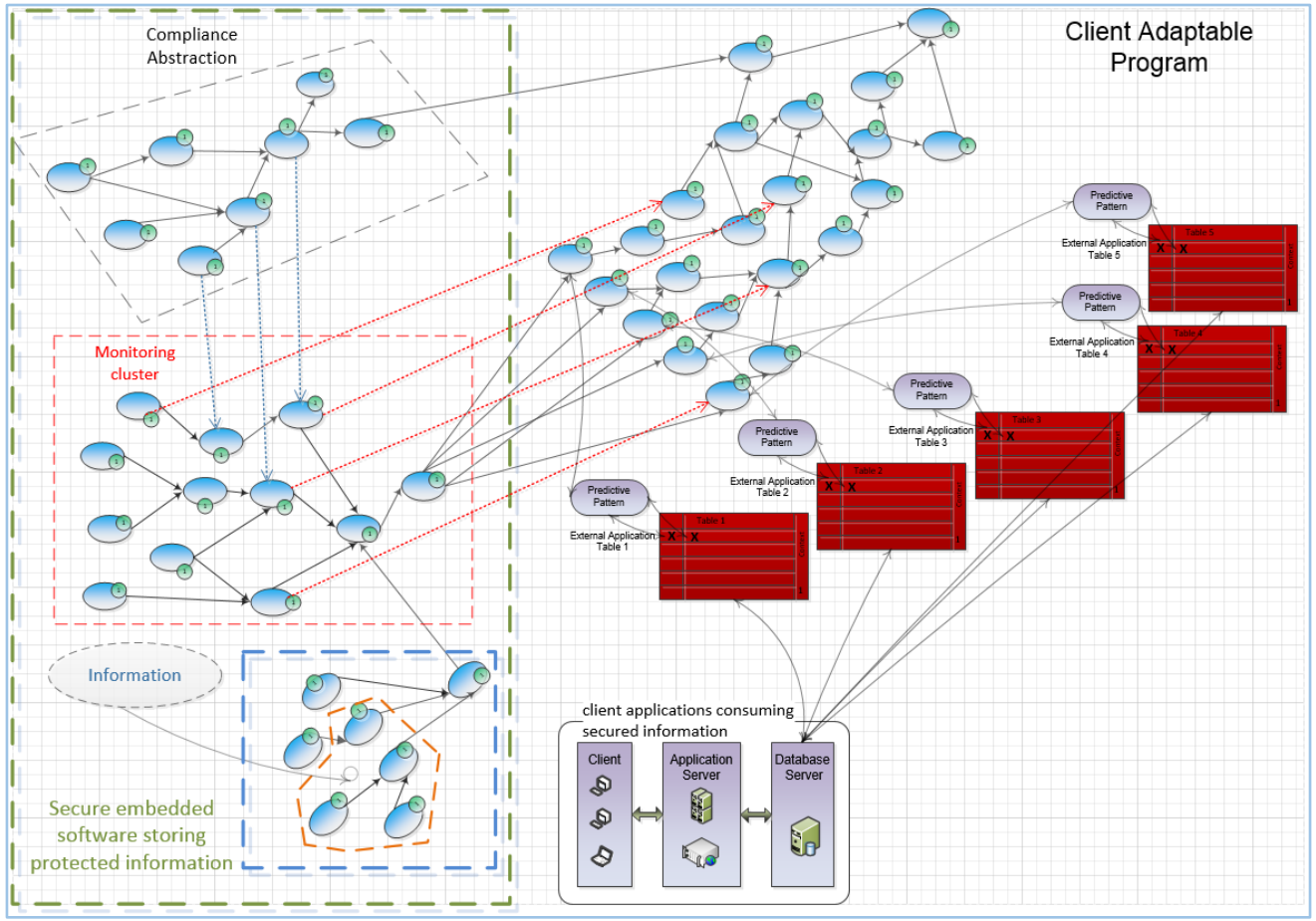
Future expected technologies:

Conceptual models that can be read by a computer as written. This goal has been pursued since the inception of computerization, however it is possible that the challenges far outweigh the value

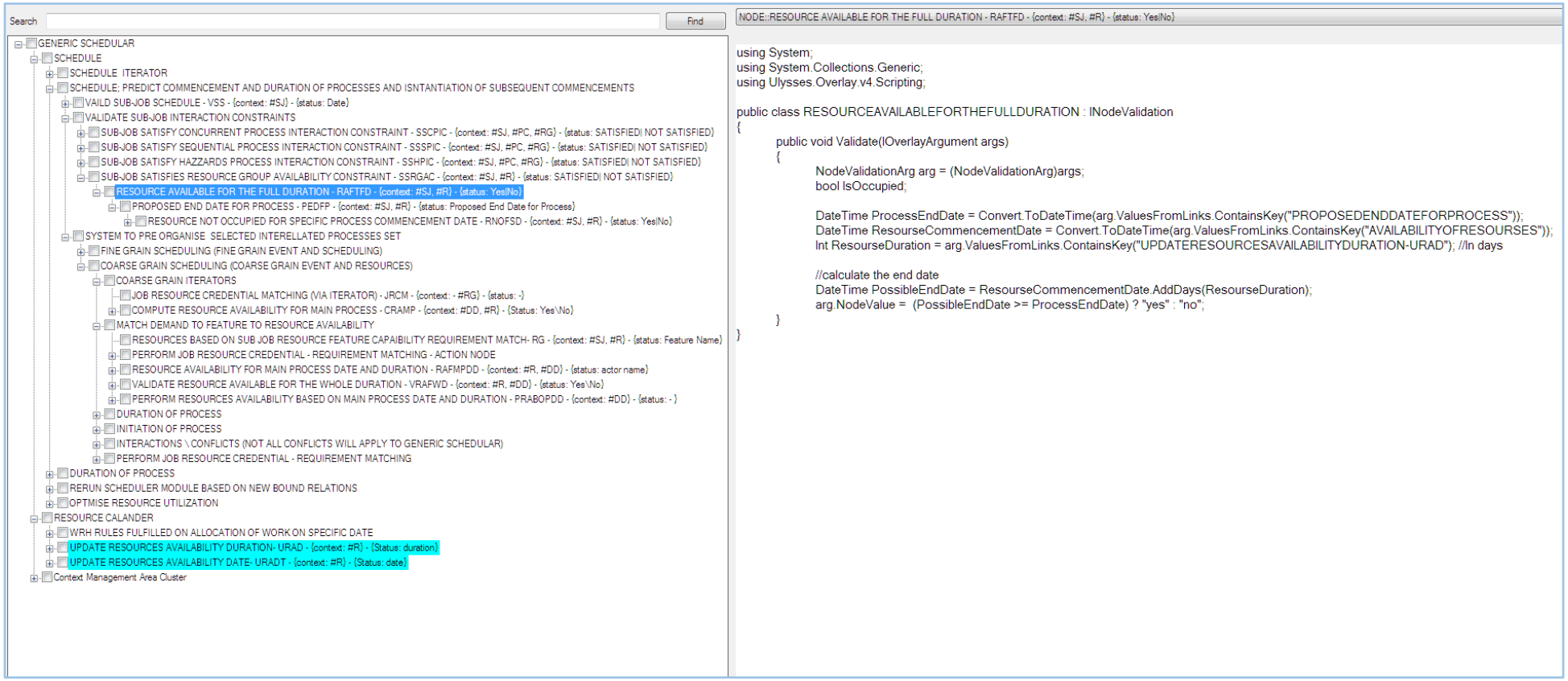
Interrogation by stakeholders: Logic needing to be scrutinized by many stakeholders

Imagine a self-driving car making decisions. How many stakeholders are affected?

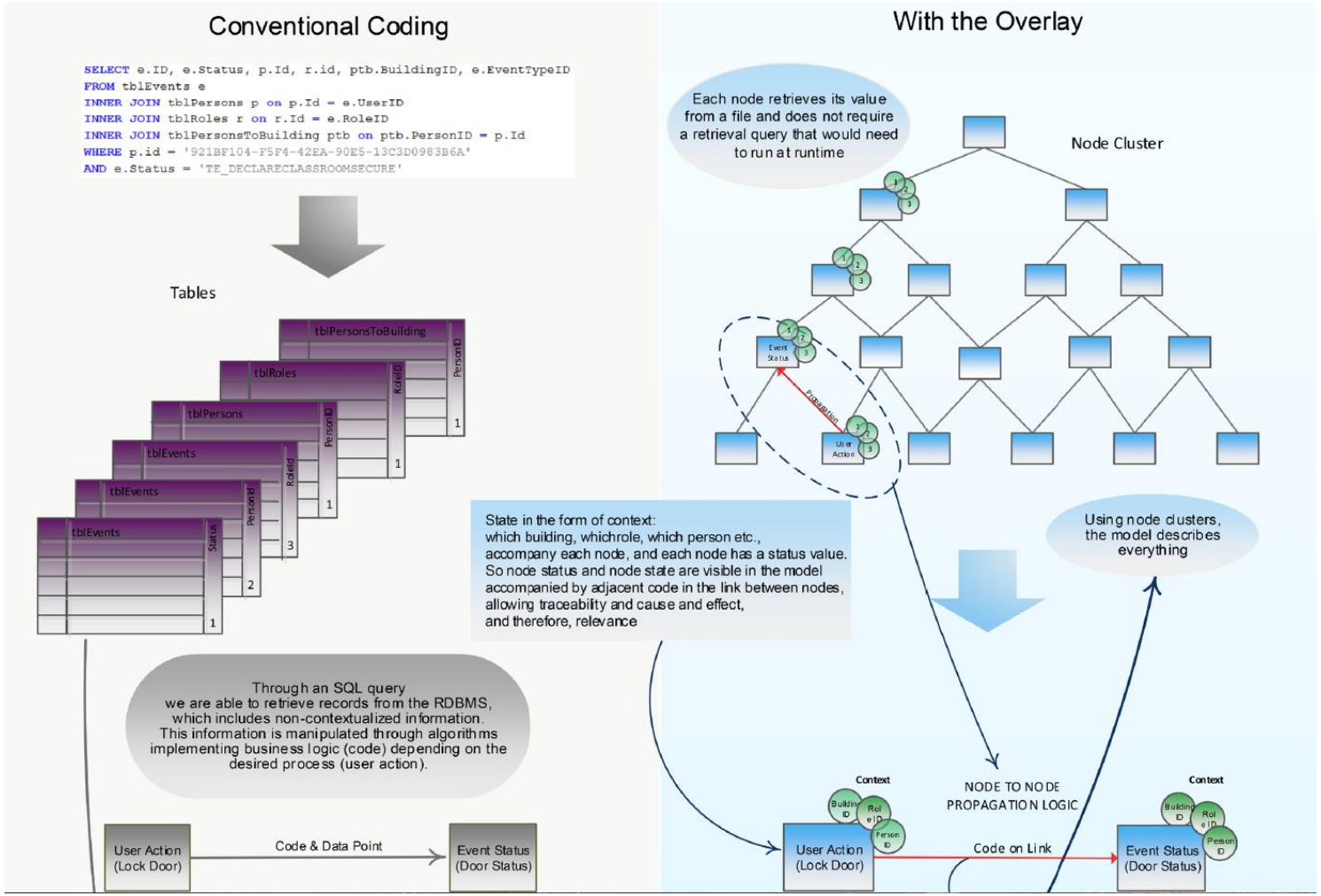
Predicting behavior: Secure data needing to be protected from unauthorized access



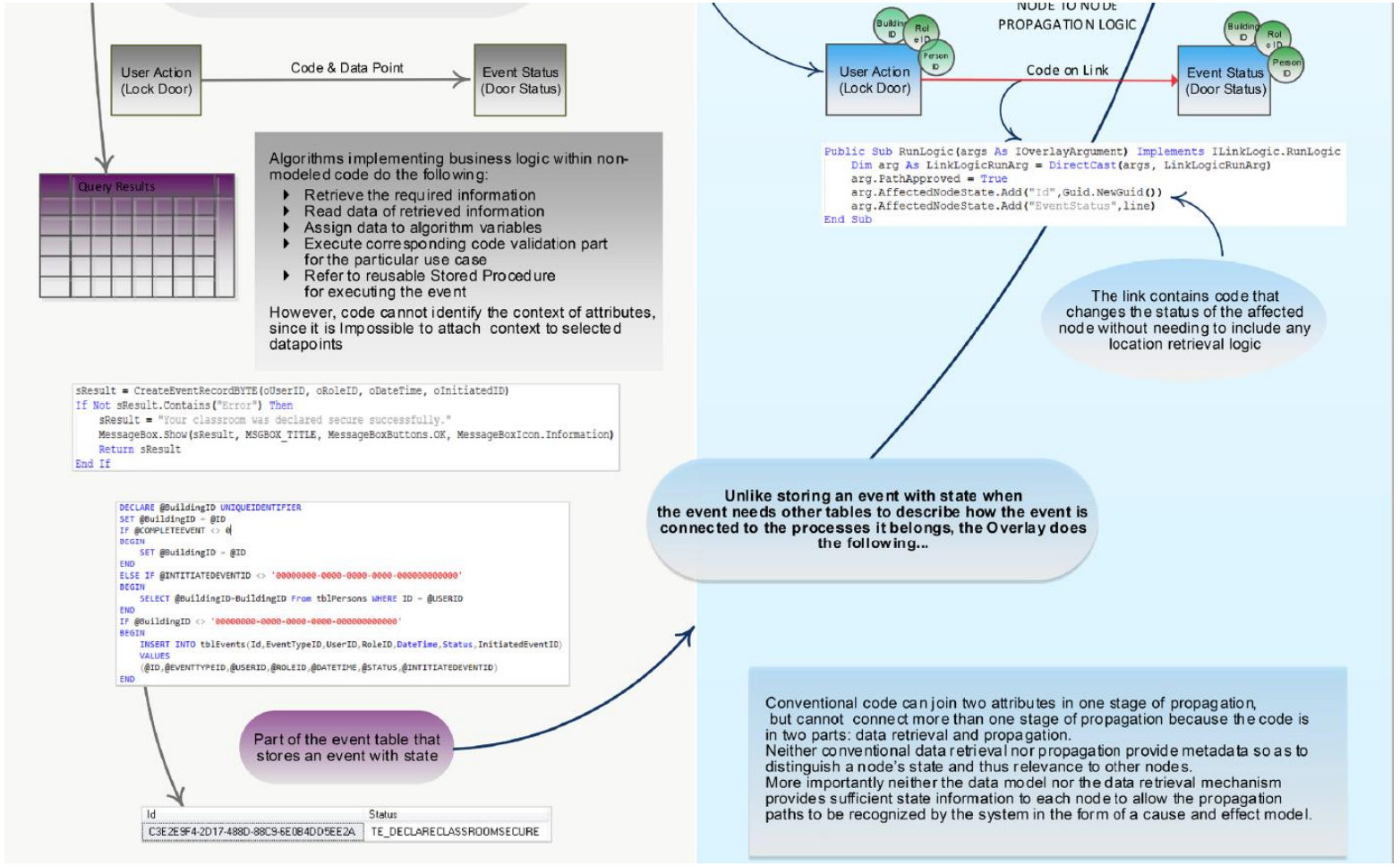
A lot of logic; Scheduling repairs in the marine business



Executing models versus non executing models; What's the difference in technical terms;



Executing models versus non executing models; What's the difference in technical terms;



TASK ASSISTANT
Software that works

Ulysses
SYSTEMS

THANK YOU

Since 1996, Ulysses Systems is a leader in software innovation for the Maritime Industry, providing management solutions to ship-owners and ship managers. Its award winning software, Task Assistant® enables both office and seagoing personnel to work intuitively, efficiently and effectively. Class certified by DNV, Bureau Veritas and ClassNK, Task Assistant® is designed to require minimal training. Managers should expect a fast return on total software lifecycle cost from reliable and mature process optimization and minimization of information gaps.

info@ulysses-systems.com

ATHENS - LONDON - MUMBAI - CHICAGO

www.ulysses-systems.com

BUREAU VERITAS
DNV
ClassNK