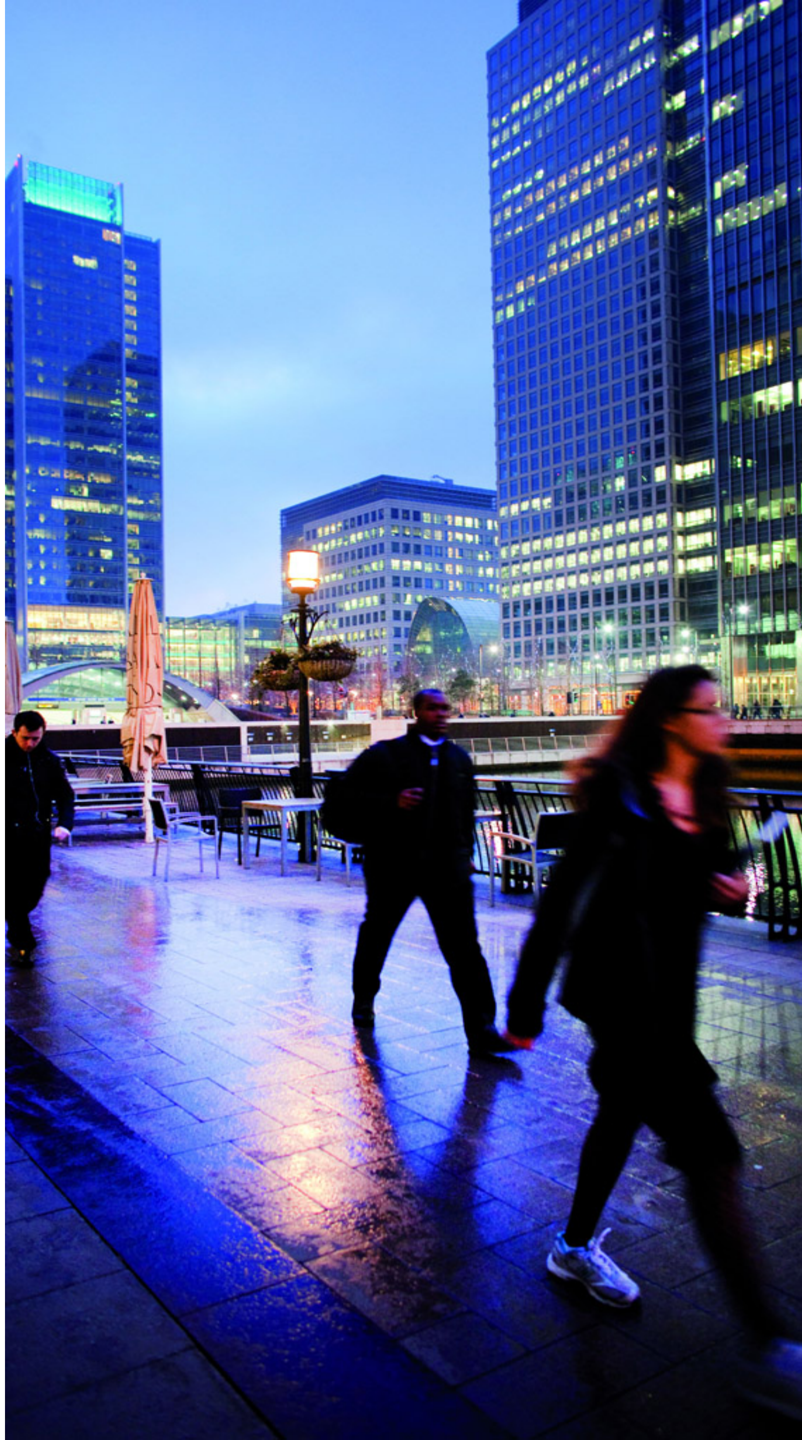
Can AI Help to Accurately Formalise Ambiguous Natural Language Requirements?

Thuy NGUYEN
Sophie EBERSOLD - *Toulouse 2, CNRS-IRIT*
Jean-Michel BRUEL - *Toulouse 2, CNRS-IRIT*
Imen SAYAR - *INRIA*

GDR GPL, RADIA & IE
November 21st, 2024
Paris, France



Summary

- **Automatic formalisation of natural language requirements: the *k3* case study**
- **Ambiguities in *k3* requirements**
- **BASAALT and FORM-L in a nutshell**
- **Examples of *k3* requirements disambiguation and formalisation**

Automatic Formalisation of Natural Language Requirements

- ***https://github.com/kevinlano/RequirementsFormalisation* is a public case studies repository**
  - □ Each case specifies natural language (NL) requirements for a particular system
  - □ The challenge is to formalise them automatically

- **A special issue of *Software and Systems Modeling* (SoSym) is dedicated to case *k3***
  - □ 35 requirements in English for the information system of a nursing department in a teaching institution
    - *https://github.com/kevinlano/RequirementsFormalisation/blob/main/casestudies/k3data.txt*
    - *https://github.com/kevinlano/RequirementsFormalisation/blob/main/casestudies/k3ucs.txt*

- **The BASAALT method and the FORM-L language (both developed at EDF) were applied manually**
  - □ *Identifying and Fixing Ambiguities in, and Semantically Accurate Formalisation of, Behavioural Requirements*
    - *https://link.springer.com/article/10.1007/s10270-023-01142-0*

- **... to show that due to ambiguities, semantically accurate formalisation cannot be fully automated**
  - □ Current automatisation results are not very accurate: many elements present in the original NL text are not represented in the formalised text

- **But can AI help?**

k3data

d1 A class shall be either a non-clinical class or a clinical class.

d2 A non-clinical class shall specify the course name, lecture room requirements and instructor needs.

d3 A clinical class shall specify the course name, lecture room requirements, clinical site needs, lecture instructor needs and clinical lab instructor needs.

d4 A Program of Study shall consist of a program name and listing of required classes (both clinical and non-clinical) that must be completed.

d5 The report of needed classes shall include (but not be limited to) classes to be offered, number of sections needed, number of labs needed, and room types needed.

d6 Classes for a given cohort shall not conflict with regards to the time and day that they are offered.

d7 A clinical lab section shall include the clinical site name, the class instructor, day and time of the lab.

d8 Clinical site information shall include (but not be limited to) the name of the site, contact person and contact information.

d9 The system shall contain contact information (e-mail and phone number) for all people relevant to the system including (but not limited to) staff members, students, lecture instructors, clinical lab instructors and clinical site administrators.

k3ucs - 1/4

r1 **Program Administrators and Nursing Staff Members shall be able to add clinical classes or sections to a sequence of classes.**

r2 **Program Administrators and Nursing Staff Members shall be able to add a new cohort to the system identified by start month and year.**

r3 **Program Administrators and Nursing Staff Members shall be able to add new classes for the nursing department into the system.**

r4 **Program Administrators/Nursing Staff Members shall be able to create a new Program of Study.**

r5 **Program Administrators and Nursing Staff Members shall have the ability to specify which classes are required for a Program of Study.**

r6 **The system shall be able to display a report of needed classes for a given quarter for all cohorts of all programs for Program Administrators/Nursing Staff Members planning purposes.**

r7 **Program Administrators and Nursing Staff Members shall be able to add a new clinical site into the system.**

r8 **Program Administrators and Nursing Staff Members shall be able to add a new clinical lab section for an existing clinical class into the System.**

k3ucs - 2/4

r9 Program Administrators and Nursing Staff Members shall be able to add a student who has registered for a clinical class to a clinical lab section for that class.

r10 The system shall allow a Program Administrator or Nursing staff member to remove a student from a clinical lab section.

r11 The system shall allow a Program Administrator/Nursing Staff Member to move a student from one clinical lab section to another clinical lab section corresponding to the same clinical class.

r12 Program Administrators/Nursing Staff Members shall be able to cancel a clinical lab section only if there are no students registered for that clinical lab section.

r13 A Program Administrator/Nursing Staff Member shall be able to add a new nursing student to the system.

r14 A staff member shall have the ability to manage the progress of a part time student which includes modifying their sequence of classes and timeline of classes.

r15 The system shall be able to display a printable summary for individual cohorts, which will include the students enlisted, the Program of study, sequence of classes, cohort progress through the program, and timeline of completion.

k3ucs - 3/4

r16 The system shall be able to display a printable summary for individual clinical sites, which will include information on labs being held at that site for the given quarter (Clinical site, general information on labs held at that location, dates and times of all labs held there).

r17 The system shall be able to display a printable summary for individual clinical labs which will include (but not be limited to) class name, class number, lab section, listing of students enrolled, instructor, day(s), time and location.

r18 The system shall be able to display a printable summary for individual nursing students, which will include (but not be limited to) student name, student ID, admission date, classes, credits, GPA and the cohort that the student is enrolled in.

r19 The system shall be able to display a printable summary of the system log.

r20 The system will notify affected parties when changes occur affecting cohorts, including but not limited to changes to the sequence for a cohort's program of study and changes to a given week's schedule (lab cancelled this week due to instructor illness).

r21 The system will notify affected parties when changes occur affecting specific students, including but not limited to changing a student's status from full time to part time and moving a student to a different clinical section.

k3ucs - 4/4

r22 **Program Administrators/Nursing Staff Members shall have the ability to modify information relating to cohorts, including cohort identifier, program of study, preferred sequence of classes and quarters that a cohort will be taking specific classes.**

r23 **Program Administrators/Nursing Staff Members shall have the ability to modify information relating to a Program of Study within the Nursing Department, including the Program of study name, and required classes for that Program of Study.**

r24 **Program Administrators/Nursing Staff Members shall have the ability to modify information relating to a Clinical Site, including the clinical site name, site contact person, contact information, and address.**

r25 **Program Administrators/Nursing Staff Members shall have the ability to modify information relating to a Clinical Lab Section, including the clinical site name, the department/section for the lab, the class instructor, day and time of lab, and contact information.**

r26 **Program Administrators/Nursing Staff Members shall have the ability to modify information relating to a Nursing Student, including student ID, student name, phone number, e-mail status (full time or part time), and program of study.**

Summary

- **Automatic formalisation of natural language requirements: the *k3* case study**
- **Ambiguities in *k3* requirements**
- **BASAALT and FORM-L in a nutshell**
- **Examples of *k3* requirements disambiguation and formalisation**

Ambiguities and Implicit Information

- **For communication concision and “efficiency”, in NL, authors do not express what is “obvious”**
  - □ Audiences rely on context and common sense to determine what authors want to convey
- **NL processing (NLP) / AI might help identify ambiguities**
- **... but fixing them often requires a deep, application-specific understanding of**
  - □ The system of interest
  - □ Its environment and what is assumed about it
  - □ What is achievable and what is not in a given project context
- **Even humans endowed with common sense cannot correctly resolve certain ambiguities if they lack the full application context and are not sufficiently informed of system owners intentions**
  - □ Fot the *k3* study, we had to make many arbitrary choices
- **In addition**
  - □ Elements not mentioned in NL are often needed for well-formed formal statements
  - □ Some NL elements are given for information but are not part of the requirements

Five Categories of Ambiguity in *k3*: 1 - System Ambiguity

- **When it is unclear**
  - What the system of interest (to which the requirements apply) is
  - What belongs to it and what belongs to its environment
  - What is assumed regarding that environment
- **In *k3*, there is no indication of what the system of interest is**
  - Is it purely technical, users being part of its environment?
  - Or is it a socio-technical system that includes its users?

Five Categories of Ambiguity in *k3*: 1 - System Ambiguity

- **When it is unclear**
  - What the system of interest (to which the requirements apply) is
  - What belongs to it and what belongs to its environment
  - What is assumed regarding that environment
- **In *k3*, there is no indication of what the system of interest is**
  - Is it purely technical, users being part of its environment?
  - Is it an information system?
  - Or is it a socio-technical system that includes its users?
  - Or is it a cyber-physical system?

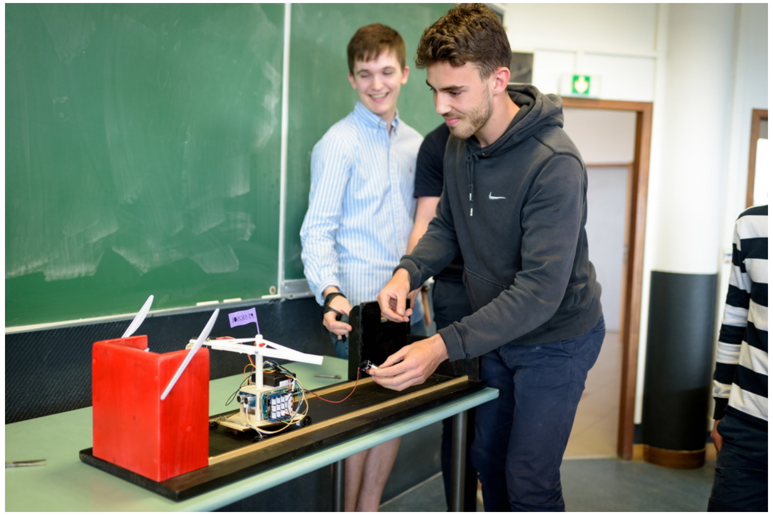
Five Categories of Ambiguity in *k3*: 1 - System Ambiguity

- **When it is unclear**
  - □ What the system of interest (to which the requirements apply) is
  - □ What belongs to it and what belongs to its environment
  - □ What is assumed regarding that environment
- **In *k3*, there is no indication of what the system of interest is**
  - □ Is it purely technical, users being part of its environment?
  - □ Is it an information system?
  - □ Is it a system for the exclusive use of the Nursing department?
    - Then, *students* are necessarily *nursing students*
  - □ Or is it a socio-technical system that includes its users?
  - □ Or is it a cyber-physical system?

Five Categories of Ambiguity in *k3*: 1 - System Ambiguity

- **When it is unclear**
  - What the system of interest (to which the requirements apply) is
  - What belongs to it and what belongs to its environment
  - What is assumed regarding that environment

- **In *k3*, there is no indication of what the system of interest is**
  - Is it purely technical, users being part of its environment?
  - Is it an information system?
  - Is it a system for the exclusive use of the Nursing department?
    - Then, *students* are necessarily *nursing students*
  - Or is it a socio-technical system that includes its users?
  - Or is it a cyber-physical system?
  - Or is it for the complete teaching institution?
    - Then, *students* also includes *non-nursing students*



Five Categories of Ambiguity in *k3*: 2a - Lexical Ambiguity

- **When the exact meaning of terms denoting notions important to the application is unclear**
  - □ In *k3*, that concerns terms like *class*, *cohort*, *section*, *student*, and some others

- **In an educational context, the Merriam-Webster dictionary defines *class* as**
  - □ A set of students studying a subject together
    - *A student of the class is absent*
  - □ A period where such a body meets
    - *No class today*
  - □ A course of instruction
    - *The algebra class*
  - □ A set of students or alumni who graduated the same year
    - *The class of 1999*

Five Categories of Ambiguity in *k3*: 2a - Lexical Ambiguity

- **When the exact meaning of terms denoting notions important to the application is unclear**
  - □ In *k3*, that concerns terms like *class*, *cohort*, *section*, *student*, and some others

- **In an educational context, the Merriam-Webster dictionary defines *class* as**
  - □ A set of students studying a subject together
    - *A student of the class is absent*
  - □ A period where such a body meets
    - *No class today*
  - □ A course of instruction
    - *The algebra class*
  - □ A set of students or alumni who graduated the same year
    - *The class of 1999*

r5 **Program Administrators and Nursing Staff Members shall have the ability to specify which classes are required for a Program of Study.**

Five Categories of Ambiguity in *k3*: 2a - Lexical Ambiguity

- **When the exact meaning of terms denoting notions important to the application is unclear**
  - In *k3*, that concerns terms like *class*, *cohort*, *section*, *student*, and some others

- **In an educational context, the Merriam-Webster dictionary defines *class* as**
  - ✗ A set of students studying a subject together
    - *A student of the class is absent*
  - ? A period where such a body meets
    - *No class today*
  - ✓ A course of instruction
    - *The algebra class*
  - ✗ A set of students or alumni who graduated the same year
    - *The class of 1999*

r5 **Program Administrators and Nursing Staff Members shall have the ability to specify which classes are required for a Program of Study.**

Five Categories of Ambiguity in *k3*: 2a - Lexical Ambiguity

- **When the exact meaning of terms denoting notions important to the application is unclear**
  - □ In *k3*, that concerns terms like *class*, *cohort*, *section*, *student*, and some others

- **In an educational context, the Merriam-Webster dictionary defines *class* as**
  - ✗ A set of students studying a subject together
    - *A student of the class is absent*
  - ? A period where such a body meets
    - *No class today*
  - ✓ A course of instruction
    - *The algebra class*
  - ✗ A set of students or alumni who graduated the same year
    - *The class of 1999*

r5 **Program Administrators and Nursing Staff Members shall have the ability to specify which classes are required for a Program of Study.**

- **The answer might not be the same throughout the set of requirements**

Five Categories of Ambiguity in *k3*: 2b - Lexical Ambiguity

- **The same dictionary defines *cohort* as**
  - □ Companion, colleague
  - □ A set of individuals having a factor in common in a demographic study
  - □ One of 10 divisions of a Roman legion
  - □ A group of warriors or soldiers
- **None of these definitions evidently apply in an educational context**
- **A bespoke interpretation had to be figured out, based on a close inspection of all the specified requirements**

Five Categories of Ambiguity in *k3*: 2b - Lexical Ambiguity

- **The same dictionary defines *cohort* as**
  - Companion, colleague
  - A set of individuals having a factor in common in a demographic study
  - One of 10 divisions of a Roman legion
  - A group of warriors or soldiers
- **None of these definitions evidently apply in an educational context**
- **A bespoke interpretation had to be figured out, based on a close inspection of all the specified requirements**
  - The set of students following the same program of study and the same classes at the same time
  - Class: period where such a body meets?

Five Categories of Ambiguity in *k3*: 2c - Lexical Ambiguity

- **May also affect nominal groups (nouns with adjectives) and verbal groups (verbs with adverbs)**
- **For example, *k3* mentions *section*, *clinical section*, *lab section* and *clinical lab section***

Five Categories of Ambiguity in *k3*: 2c - Lexical Ambiguity

- **May also affect nominal groups (nouns with adjectives) and verbal groups (verbs with adverbs)**
- **For example, *k3* mentions *section*, *clinical section*, *lab section* and *clinical lab section***
- **Many possible interpretations**
  - The four nominal groups represent the same notion
    - *Section*, *clinical section* and *lab section* being shortcuts for *clinical lab section*

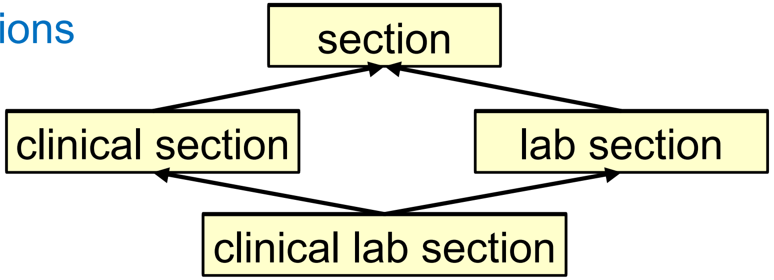
((clinical) (lab)) section

Five Categories of Ambiguity in *k3*: 2c - Lexical Ambiguity

- **May also affect nominal groups (nouns with adjectives) and verbal groups (verbs with adverbs)**
- **For example, *k3* mentions *section*, *clinical section*, *lab section* and *clinical lab section***
- **Many possible interpretations**
  - □ The four nominal groups represent the same notion
    - *Section*, *clinical section* and *lab section* being shortcuts for *clinical lab section*

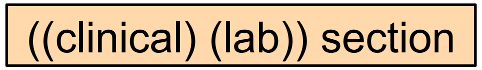
((clinical) (lab)) section

  - □ The four nominal groups represent four different but interrelated notions



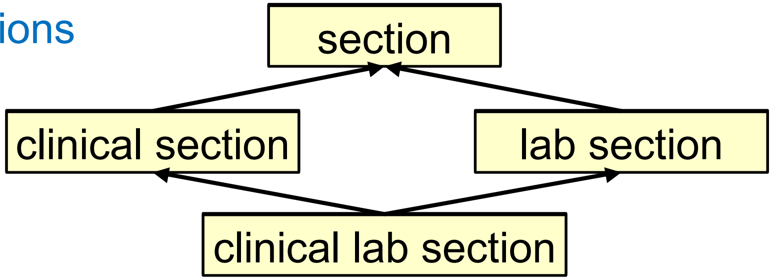
Five Categories of Ambiguity in *k3*: 2c - Lexical Ambiguity

- **May also affect nominal groups (nouns with adjectives) and verbal groups (verbs with adverbs)**
- **For example, *k3* mentions *section*, *clinical section*, *lab section* and *clinical lab section***
- **Many possible interpretations**
  - The four nominal groups represent the same notion
    - *Section*, *clinical section* and *lab section* being shortcuts for *clinical lab section*



((clinical) (lab)) section

  - The four nominal groups represent four different but interrelated notions



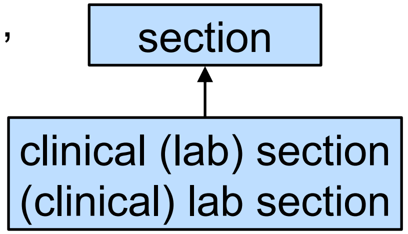
section

clinical section

lab section

clinical lab section

  - *Clinical section*, *lab section* and *clinical lab section* represent the same notion, *section* represent a different but interrelated notion



section

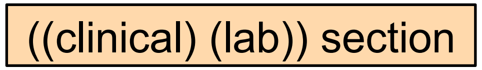
clinical (lab) section
(clinical) lab section

Five Categories of Ambiguity in *k3*: 2c - Lexical Ambiguity

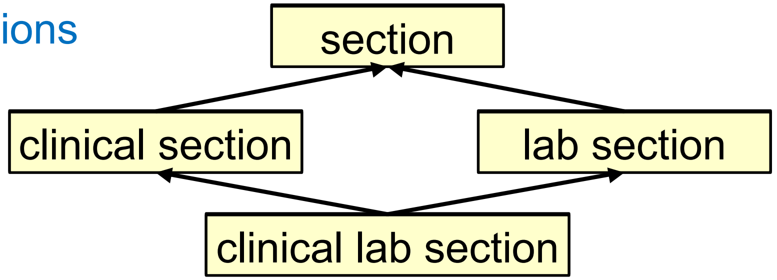
- **May also affect nominal groups (nouns with adjectives) and verbal groups (verbs with adverbs)**

- **For example, *k3* mentions *section*, *clinical section*, *lab section* and *clinical lab section***

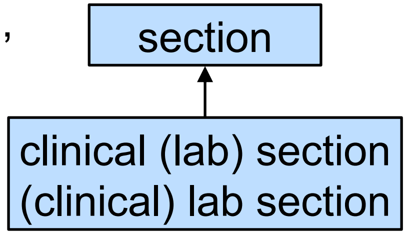
- **Many possible interpretations**
  - The four nominal groups represent the same notion
    - *Section*, *clinical section* and *lab section* being shortcuts for *clinical lab section*



  - The four nominal groups represent four different but interrelated notions



  - *Clinical section*, *lab section* and *clinical lab section* represent the same notion, *section* represent a different but interrelated notion



  - Etc.

Five Categories of Ambiguity in *k3*: 2d - Lexical Ambiguity

- **Also, how a well-defined term relates to the real, physical world can be unclear**
  - Again, the answer may not be the same throughout the set of requirements

- **In *k3*, term *student* sometimes refers to actual human beings**

d9 **The system shall contain contact information (e-mail and phone number) for all people relevant to the system including (but not limited to) staff members, students, lecture instructors, clinical lab instructors and clinical site administrators.**

Five Categories of Ambiguity in *k3*: 2d - Lexical Ambiguity

- **Also, how a well-defined term relates to the real, physical world can be unclear**
  - And the answer may not be the same throughout the set of requirements

- **In *k3*, term *student* sometimes refers to actual human beings**

d9 **The system shall contain contact information (e-mail and phone number) for all people relevant to the system including (but not limited to) staff members, students, lecture instructors, clinical lab instructors and clinical site administrators.**

- **... but at other times it most probably refers to information about them**

r10 **The system shall allow a Program Administrator or Nursing staff member to remove a student from a clinical lab section.**

Five Categories of Ambiguity in *k3*: 2d - Lexical Ambiguity

- **Also, how a well-defined term relates to the real, physical world can be unclear**
  - □ And the answer may not be the same throughout the set of requirements

- **In *k3*, term student sometimes refers to actual human beings**

d9 **The system shall contain contact information (e-mail and phone number) for all people relevant to the system including (but not limited to) staff members, students, lecture instructors, clinical lab instructors and clinical site administrators.**

- **... but at other times it most probably refers to information about them**

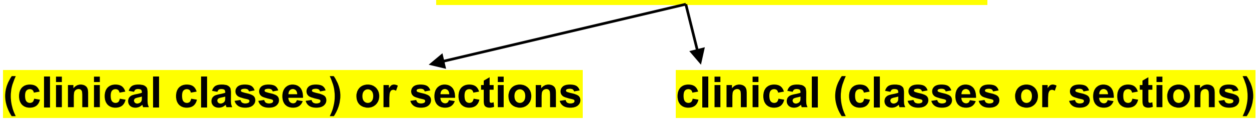
r10 **The system shall allow a Program Administrator or Nursing staff member to remove a student from a clinical lab section.**

A possible (but unlikely) interpretation of ***r10***

Five Categories of Ambiguity in *k3*: 3 - Syntactic Ambiguity

- **When an NL sentence can be grammatically parsed in different ways**

r1 **Program Administrators and Nursing Staff Members shall be able to add clinical classes or sections to a sequence of classes.**



(clinical classes) or sections

clinical (classes or sections)

- **Even after close examination of all the 35 *k3* requirements, both are equally plausible**

- **But each leads to a different data model**

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
    Class PA "Program Administrators" ...;
```

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
    Class PA "Program Administrators" ...;
```

  - □ Each individual Program Administrator

```
    Requirement r1 is
      for all x of PA
      assert x can...;
```

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
    Class PA "Program Administrators" ...;
```

  - □ Each individual Program Administrator

universal quantifier

```
    Requirement r1 is
      for all x of PA
      assert x can...;
```

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
    Class PA "Program Administrators" ...;
```

  - □ Each individual Program Administrator

```
    Requirement r1 is
      for all x of PA
      assert x can...;
```

universal quantifier

set of all PA instances

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

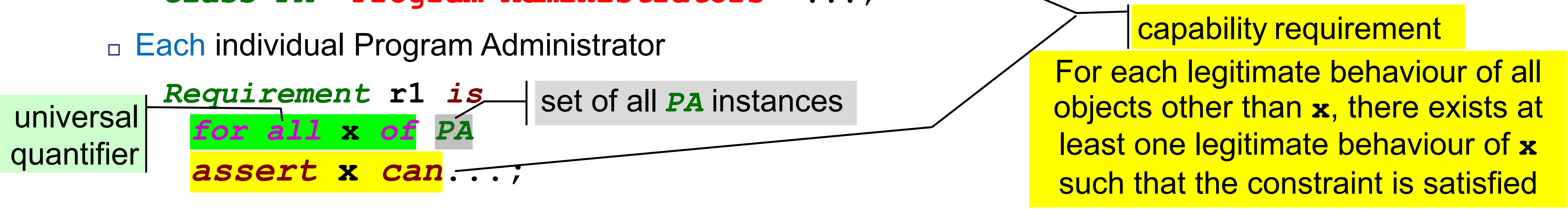
- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
    Class PA "Program Administrators" ...;
```

  - □ Each individual Program Administrator



```
    Requirement r1 is
      for all x of PA
      assert x can...;
```

universal quantifier

set of all *PA* instances

capability requirement

For each legitimate behaviour of all objects other than **x**, there exists at least one legitimate behaviour of **x** such that the constraint is satisfied

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
Class PA "Program Administrators" ...;
```

- □ Each individual Program Administrator

```
Requirement r1 is
  for all x of PA
  assert x can...;
```

universal quantifier

set of all *PA* instances

capability requirement

For each legitimate behaviour of all objects other than **x**, there exists at least one legitimate behaviour of **x** such that the constraint is satisfied

- □ Some individual Program Administrators

```
Requirement r1 is
  for some x of PA
  assert x can ...;
```

existential quantifier

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
Class PA "Program Administrators" ...;
```

capability requirement

For each legitimate behaviour of all objects other than **x**, there exists at least one legitimate behaviour of **x** such that the constraint is satisfied

- □ Each individual Program Administrator

universal quantifier

```
Requirement r1 is
  for all x of PA
  assert x can...;
```

set of all *PA* instances

- □ Some individual Program Administrators

existential quantifier

```
Requirement r1 is
  for some x of PA
  assert x can ...;
```

- □ All Program Administrators together, as a whole (like in *Every four years, citizens shall elect the head of state*)

```
Requirement r1 is
  assert PA can ...;
```

Five Categories of Ambiguity in *k3*: 4 - Semantic Ambiguity

- **When the exact meaning of commonly used words, which generally do not need to be explicitly defined, is unclear within the framework of a given sentence**

r1 **Program Administrators ... shall be able to add clinical classes or sections to a sequence of classes.**

- **The plural form could be interpreted in different manners even when there is a clear definition of *Program Administrator***

```
Class PA "Program Administrators" ...;
```

capability requirement

For each legitimate behaviour of all objects other than **x**, there exists at least one legitimate behaviour of **x** such that the constraint is satisfied

  - □ Each individual Program Administrator

universal quantifier

```
Requirement r1 is
  for all x of PA
  assert x can...;
```

set of all *PA* instances

  - □ Some individual Program Administrators

existential quantifier

```
Requirement r1 is
  for some x of PA
  assert x can  ...;
```

  - □ All Program Administrators together, as a whole (like in *Every four years, citizens shall elect the head of state*)

```
Requirement r1 is
  assert PA can  ...;
```

- **The same can be said of the singular form: *a Program Administrator* could mean *a single one of* or *each one of***

Five Categories of Ambiguity in *k3*: 5a - Silence

- **When one or more pieces of information essential to precise, complete, unambiguous requirements are missing**
  - □ *When event E occurs, the system shall perform action A*

No timing specified, although no system can react instantaneously

- **Some level of silence might be acceptable in NL requirements, for which one expects and can tolerate a certain level of informality**
- **That is absolutely not the case in formal requirements, for which one expects utmost precision**
- **In *k3*, no time limits are given in any of the requirements specifying actions or outcomes required of the system**
  - □ During validation tests, system developers can always pretend that one just has to wait a little more to observe the required actions or outcomes

Five Categories of Ambiguity in *k3*: 5b - Noise

- **When parts of the text do not represent real requirements**

r20 **The system will notify affected parties when changes occur affecting cohorts, including but not limited to changes to the sequence for a cohort's program of study and changes to a given week's schedule (lab cancelled this week due to instructor illness).**

- **Common sense can usually be applied**

- **... but can an AI do it?**

Summary

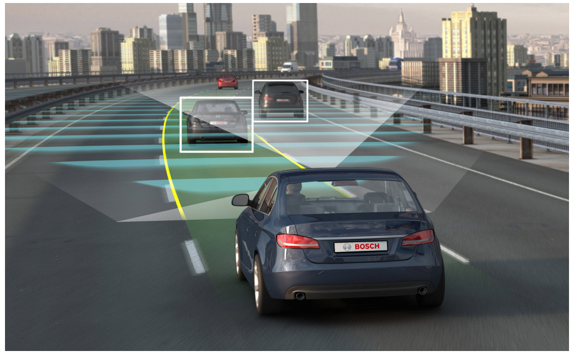
- **Automatic formalisation of natural language requirements: the *k3* case study**

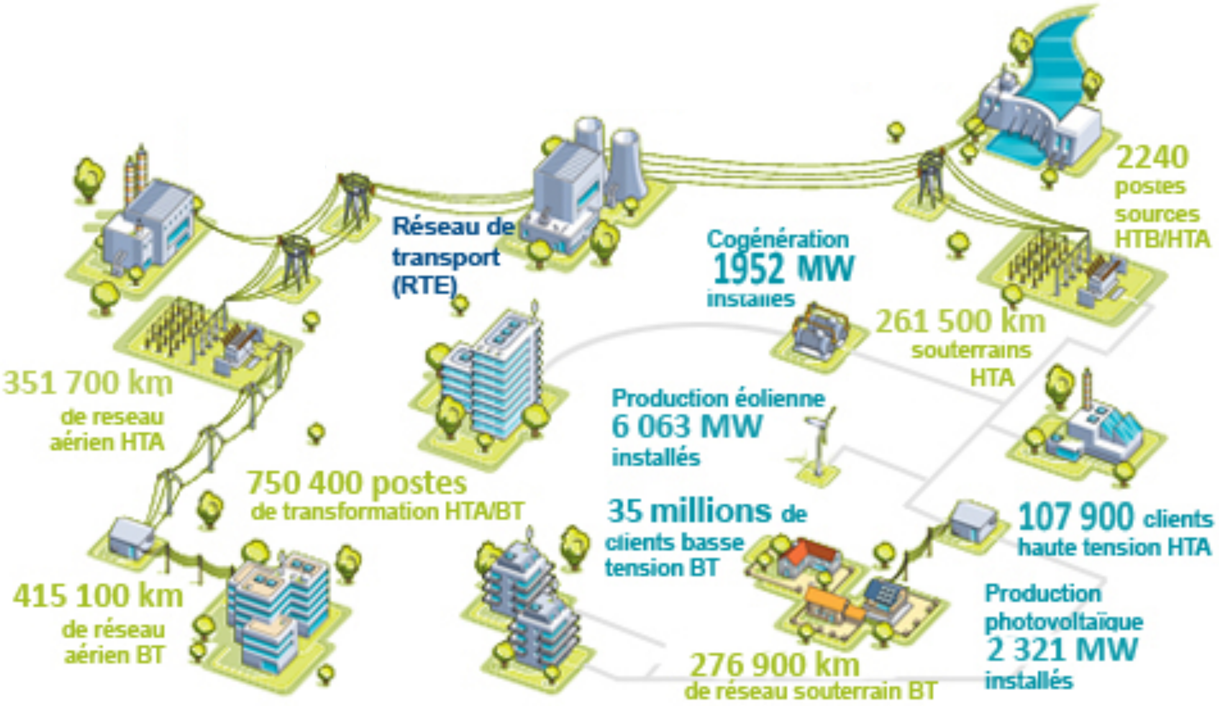
- **Ambiguities in *k3* requirements**

- **BASAALT and FORM-L in a nutshell**

- **Examples of *k3* requirements disambiguation and formalisation**

Industrial Systems




Réseau de transport (RTE)
Cogénération 1952 MW installés
2240 postes sources HTB/HTA
261 500 km souterrains HTA
351 700 km de reseau aérien HTA
Production éolienne 6 063 MW installés
750 400 postes de transformation HTA/BT
35 millions de clients basse tension BT
107 900 clients haute tension HTA
415 100 km de réseau aérien BT
Production photovoltaïque 2 321 MW installés
276 900 km de réseau souterrain BT

Primarily large and complex physical systems,
with significant computing, networking, human and organisational aspects

Their failures can cause unacceptable human, societal, environmental and economic harm

BASAALT and FORM-L

- **At EDF, we listed criteria for selecting a method and a modelling language for the engineering of industrial systems**
  - With extensive tool-support, in particular for rigorous simulation-based verification of requirements and solutions
    - Like for software, errors are inevitable, need to be corrected as early as possible, and are unlikely to be detected only by manual approaches
  - For that, formality and semantic accuracy are essential

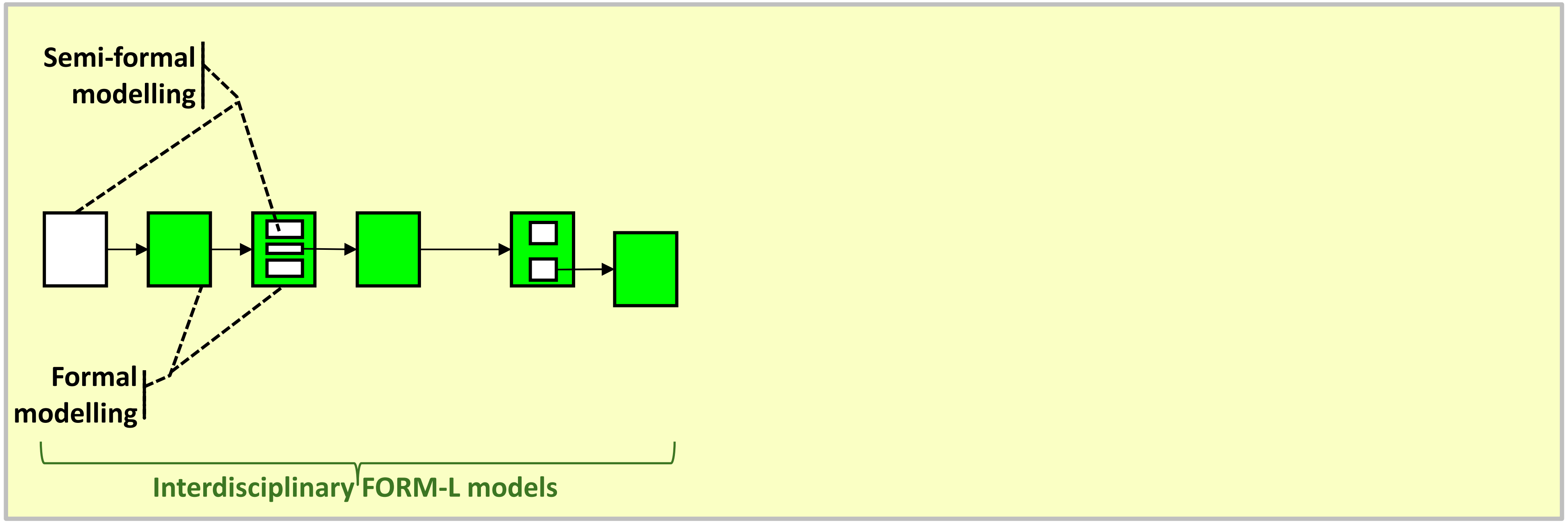
- **Unfortunately, surveys found no good candidates**

- **That work matured into the BASAALT method and the FORM-L language**
  - BASAALT: Behaviour Analysis & Simulation All Along systems Life Time
    - Behaviour: dynamic aspects and phenomena such as functions, physical states, processes, performance, costs, workloads, risks, ...
  - FORM-L: FORmal Modelling Language, a simulable language supporting BASAALT
    - Formal statements accurately reflecting all necessary behavioural elements
  - For more information, see *https://github.com/CoCoVaD/basaalt*

BASAALT in a Nutshell

Modelling & simulation framework
(behavioural assumptions, requirements and solutions)

WHAT
WHEN
WHICH
HOW
HOW WELL



Progressive refinement, possibly in agile approaches

Prospective studies
Conceptual design
Basic design

Construction, Retrofits, Upgrades

Requirements
Architecture
Detailed design

Commissioning
Validation
Integration
Unit testing

Operation
Diagnostics
Prognostics
Optimisation
Outage Planning

Data assimilation
Data reconciliation
Faster than RT simulation
Reverse time simulation
Operation optimisation
Simulation-based training

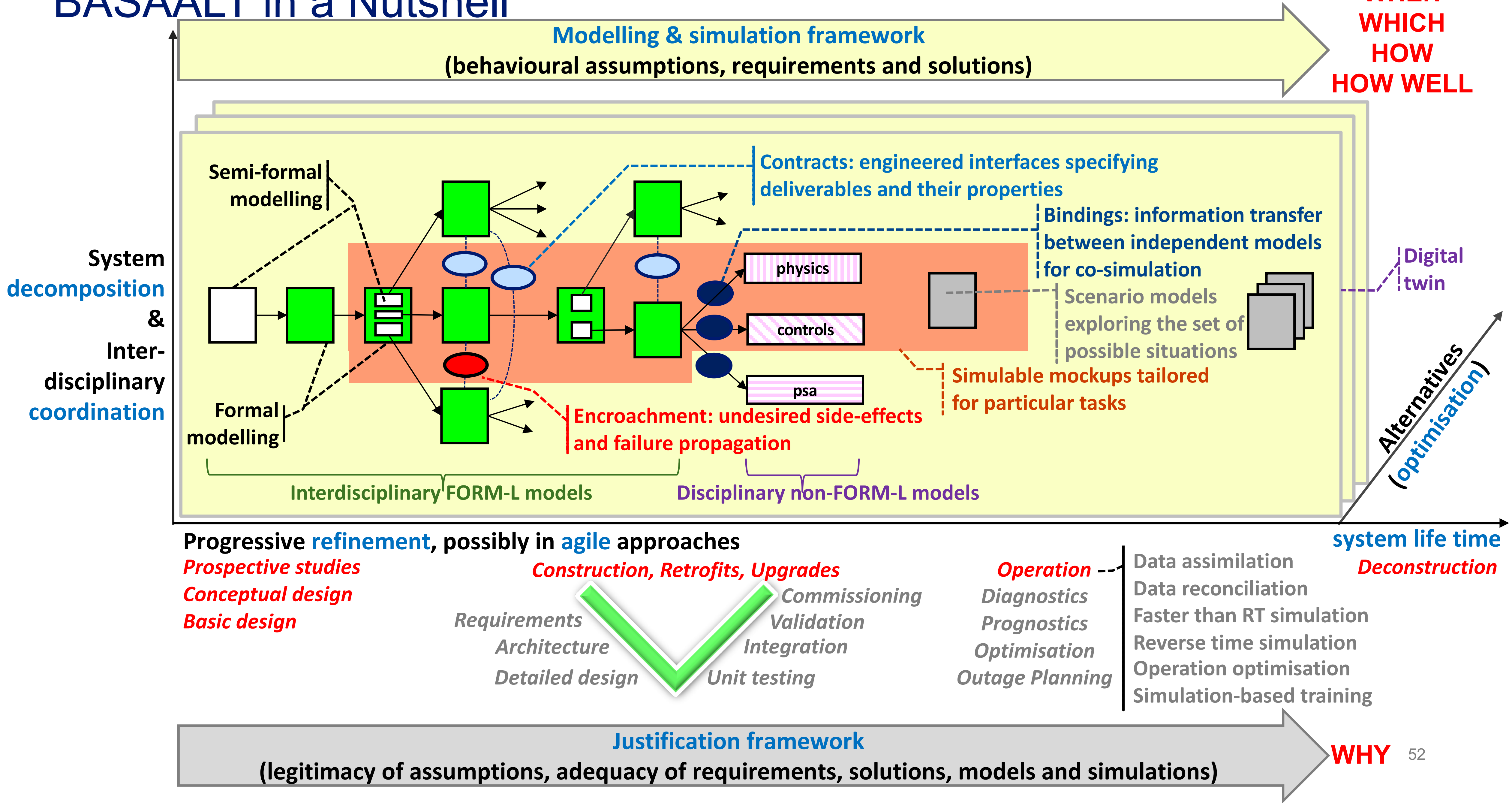
system life time

Deconstruction

Justification framework
(legitimacy of assumptions, adequacy of requirements, solutions, models and simulations)

WHY

BASAALT in a Nutshell



WHAT
WHEN
WHICH
HOW
HOW WELL

Modelling & simulation framework
(behavioural assumptions, requirements and solutions)

System decomposition & Inter-disciplinary coordination

Semi-formal modelling

Formal modelling

Contracts: engineered interfaces specifying deliverables and their properties

Bindings: information transfer between independent models for co-simulation

Scenario models exploring the set of possible situations

Digital twin

Simulable mockups tailored for particular tasks

Encroachment: undesired side-effects and failure propagation

Interdisciplinary FORM-L models

Disciplinary non-FORM-L models

Alternatives (optimisation)

Progressive refinement, possibly in agile approaches

system life time

Prospective studies
Conceptual design
Basic design

Construction, Retrofits, Upgrades

Requirements
Architecture
Detailed design

Commissioning
Validation
Integration
Unit testing

Operation
Diagnostics
Prognostics
Optimisation
Outage Planning

Data assimilation
Data reconciliation
Faster than RT simulation
Reverse time simulation
Operation optimisation
Simulation-based training

Deconstruction

Justification framework
(legitimacy of assumptions, adequacy of requirements, solutions, models and simulations)

WHY

Requirements Engineering (RE) with BASAALT: a Multi-Step Process Focused on Semantics

- **Experience shows that defects in requirements are a significant cause of systems failing to meet expectations**

- **RE always starts with imperfect requirements**
  - □ Inadequacy in certain situations
  - □ Ambiguity and incompleteness
  - □ Over-specification of solutions rather than goals
  - □ Intangibility, with abstract concepts that can be neither observed nor measured
  - □ Over-ambition, apathy and infeasibility

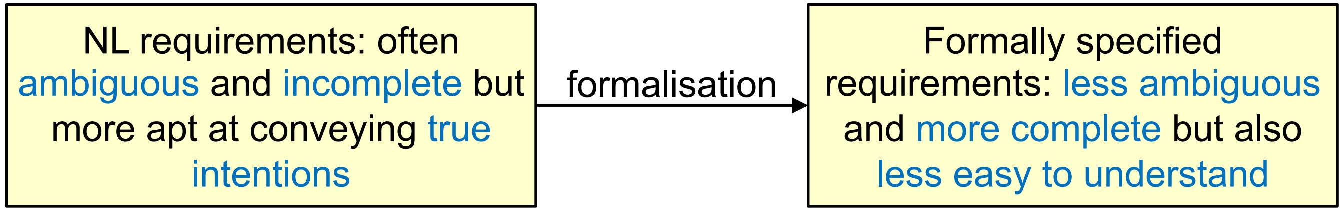
- **It is not only inevitable: sometimes, it is desirable**

Requirements Engineering (RE) with BASAALT: a Multi-Step Process Focused on Semantics

- **Experience shows that defects in requirements are a significant cause of systems failing to meet expectations**

- **RE always starts with imperfect requirements**
  - □ Inadequacy in certain situations
  - □ Ambiguity and incompleteness
  - □ Over-specification of solutions rather than goals
  - □ Intangibility, with abstract concepts that can be neither observed nor measured
  - □ Over-ambition, apathy and infeasibility

- **It is not only inevitable: sometimes, it is desirable**



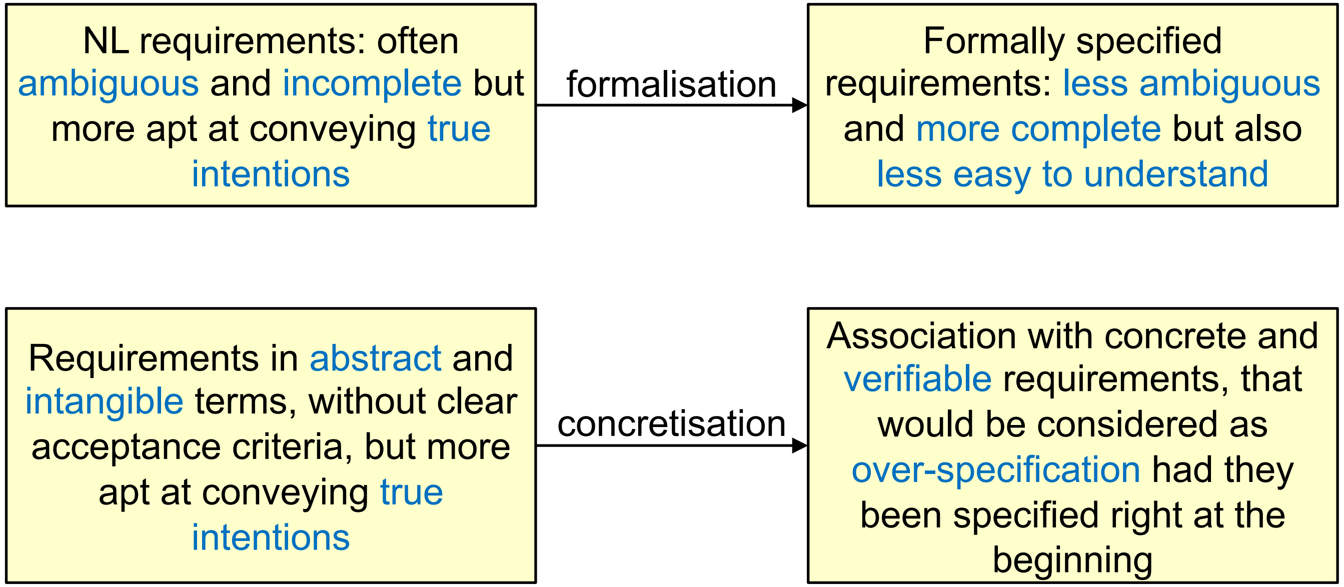
NL requirements: often ambiguous and incomplete but more apt at conveying true intentions

formalisation

Formally specified requirements: less ambiguous and more complete but also less easy to understand

Requirements Engineering (RE) with BASAALT: a Multi-Step Process Focused on Semantics

- **Experience shows that defects in requirements are a significant cause of systems failing to meet expectations**
- **RE always starts with imperfect requirements**
  - □ Inadequacy in certain situations
  - □ Ambiguity and incompleteness
  - □ Over-specification of solutions rather than goals
  - □ Intangibility, with abstract concepts that can be neither observed nor measured
  - □ Over-ambition, apathy and infeasibility
- **It is not only inevitable: sometimes, it is desirable**



NL requirements: often ambiguous and incomplete but more apt at conveying true intentions

formalisation

Formally specified requirements: less ambiguous and more complete but also less easy to understand

Requirements in abstract and intangible terms, without clear acceptance criteria, but more apt at conveying true intentions

concretisation

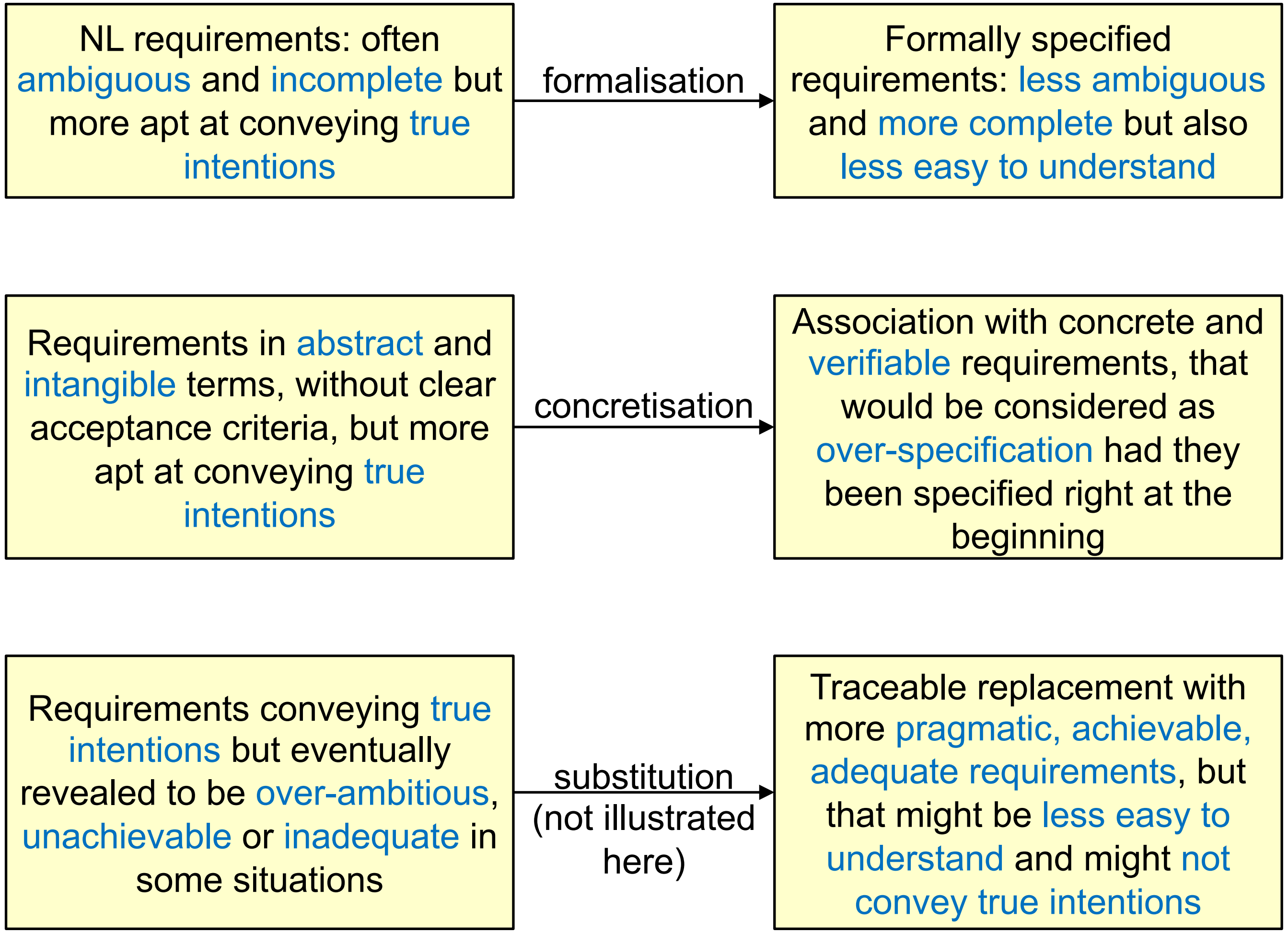
Association with concrete and verifiable requirements, that would be considered as over-specification had they been specified right at the beginning

Requirements Engineering (RE) with BASAALT: a Multi-Step Process Focused on Semantics

- **Experience shows that defects in requirements are a significant cause of systems failing to meet expectations**

- **RE always starts with imperfect requirements**
  - Inadequacy in certain situations
  - Ambiguity and incompleteness
  - Over-specification of solutions rather than goals
  - Intangibility, with abstract concepts that can be neither observed nor measured
  - Over-ambition, apathy and infeasibility

- **It is not only inevitable: sometimes, it is desirable**



NL requirements: often ambiguous and incomplete but more apt at conveying true intentions

formalisation

Formally specified requirements: less ambiguous and more complete but also less easy to understand

Requirements in abstract and intangible terms, without clear acceptance criteria, but more apt at conveying true intentions

concretisation

Association with concrete and verifiable requirements, that would be considered as over-specification had they been specified right at the beginning

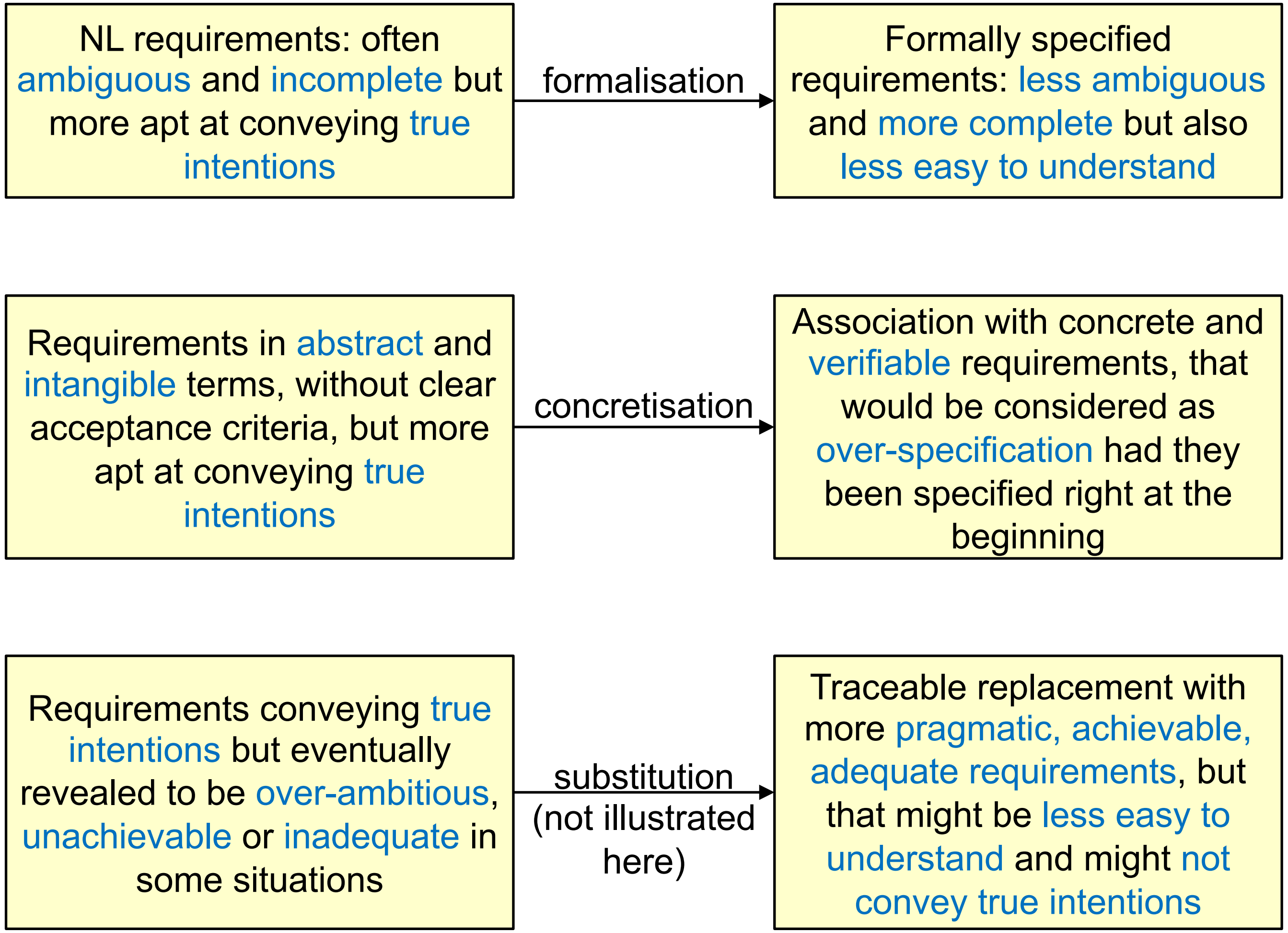
Requirements conveying true intentions but eventually revealed to be over-ambitious, unachievable or inadequate in some situations

substitution (not illustrated here)

Traceable replacement with more pragmatic, achievable, adequate requirements, but that might be less easy to understand and might not convey true intentions

Requirements Engineering (RE) with BASAALT: a Multi-Step Process Focused on Semantics

- **Experience shows that defects in requirements are a significant cause of systems failing to meet expectations**
- **RE always starts with imperfect requirements**
  - Inadequacy in certain situations
  - Ambiguity and incompleteness
  - Over-specification of solutions rather than goals
  - Intangibility, with abstract concepts that can be neither observed nor measured
  - Over-ambition, apathy and infeasibility
- **It is not only inevitable: sometimes, it is desirable**
- **One BASAALT objective is to gradually correct these defects, but also to keep track of the process, when that provides useful insights**



NL requirements: often ambiguous and incomplete but more apt at conveying true intentions

formalisation

Formally specified requirements: less ambiguous and more complete but also less easy to understand

Requirements in abstract and intangible terms, without clear acceptance criteria, but more apt at conveying true intentions

concretisation

Association with concrete and verifiable requirements, that would be considered as over-specification had they been specified right at the beginning

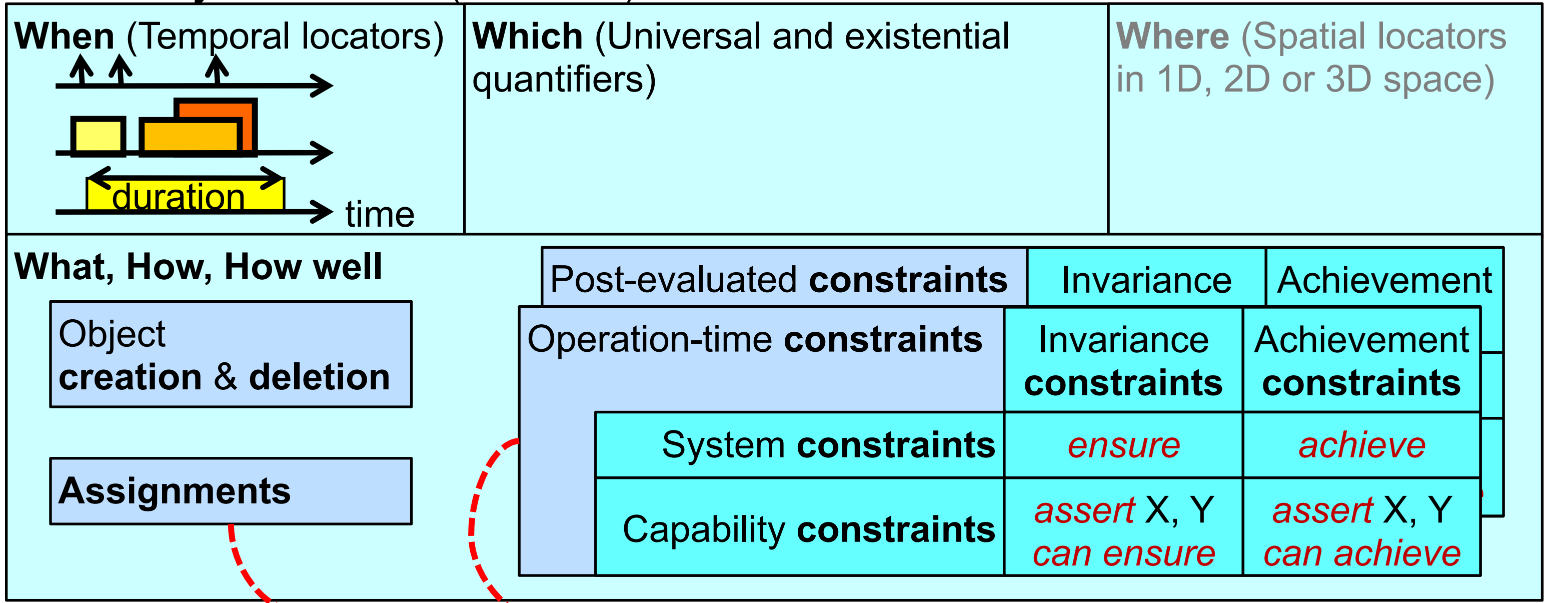
Requirements conveying true intentions but eventually revealed to be over-ambitious, unachievable or inadequate in some situations

substitution (not illustrated here)

Traceable replacement with more pragmatic, achievable, adequate requirements, but that might be less easy to understand and might not convey true intentions

FORM-L in a Nutshell

Elementary Instructions (behaviour)



When (Temporal locators)

Which (Universal and existential quantifiers)

Where (Spatial locators in 1D, 2D or 3D space)

What, How, How well

Object **creation & deletion**

Assignments

Post-evaluated **constraints**	Invariance	Achievement
Operation-time **constraints**	Invariance **constraints**	Achievement **constraints**
System **constraints**	*ensure*	*achieve*
Capability **constraints**	*assert* X, Y *can ensure*	*assert* X, Y *can achieve*

Objects (and their built-in ***attributes***)

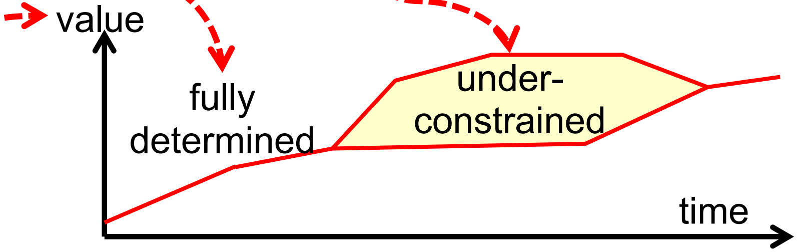
Non-valued Objects

Variables (Booleans, statecharts, Integers, Reals, quantities, Strings)

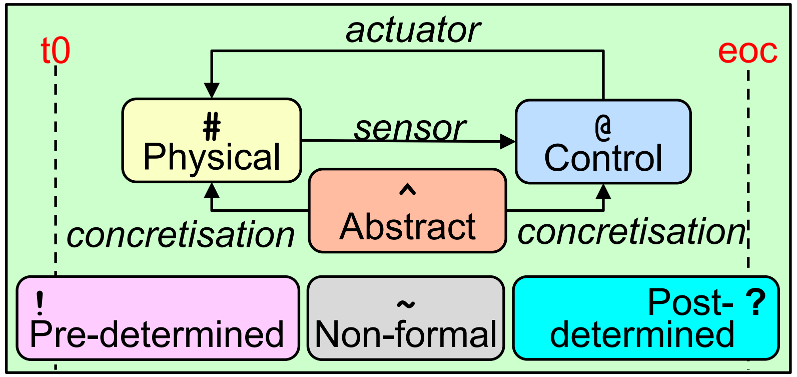
Events

Sets of objects or values

Properties, assumptions, objectives, requirements, guarantees, guards



Object determiners



FORM-L in a Nutshell

Time Domains (in Newtonian time)

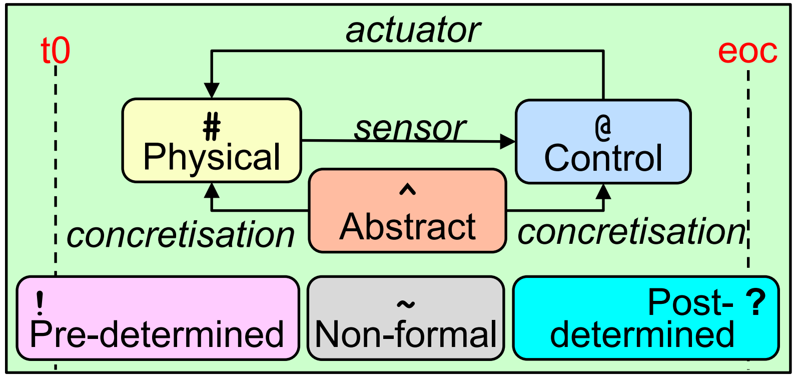
One single **Continuous Time Domain** for physical processes & human actions

Multiple **Discrete Time Domains** for Globally Asynchronous but Locally Synchronous (GALS) digital systems

Objects (and their built-in ***attributes***)

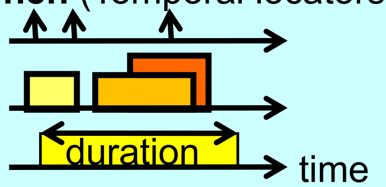
Non-valued Objects
Variables (Booleans, statecharts, Integers, Reals, quantities, Strings)
Events
Sets of objects or values
Properties, assumptions, objectives, requirements, guarantees, guards

Object determiners



Elementary Instructions (behaviour)

When (Temporal locators)



Which (Universal and existential quantifiers)

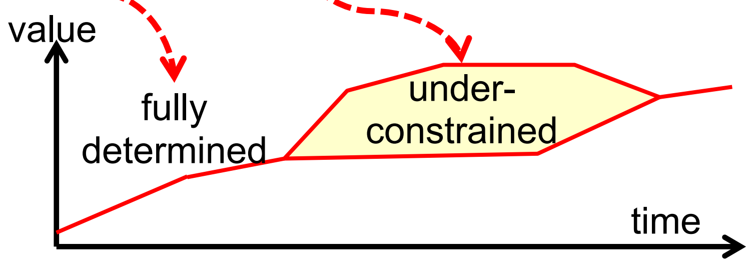
Where (Spatial locators in 1D, 2D or 3D space)

What, How, How well

Object **creation & deletion**

Assignments

Post-evaluated **constraints**	Invariance	Achievement
Operation-time **constraints**	Invariance constraints	Achievement constraints
System **constraints**	*ensure*	*achieve*
Capability **constraints**	*assert* X, Y *can ensure*	*assert* X, Y *can achieve*



Composite Instructions

Time exclusion	**Sequence**
Set exclusion	**Concurrence**
Selection	**Iteration**
(Deterministic or probabilistic)	

Interfaces

Contracts (Engineered, mutually agreed interfaces between concerned parties)

Party

- Deliverables (variables, events, sets, non-valued objects)
- Properties of deliverables

Encroachments (Undesired side effects, due e.g. to failure propagation, proximity or physical connections)

Bindings (enabling co-simulation of FORM-L models developed independently or with non FORM-L models)

Summary

- **Automatic formalisation of natural language requirements: the *k3* case study**
- **Ambiguities in *k3* requirements**
- **BASAALT and FORM-L in a nutshell**
- **Examples of *k3* requirements disambiguation and formalisation**

Two Main Formalisation Steps

- **Fixing ambiguities in, and formalisation of, the data model**
  - Identification of the necessary classes and objects
  - System / Environment dichotomy
  - Real life objects vs. Information objects
  - Inheritance relationships
  - Containment relationships
  - Undefined aspects may be ***`deferred`*** to future refinements
- **Fixing ambiguities in, and formalisation of, the requirements**
  - Fixing ambiguities in NL
  - Then formalisation in FORM-L
  - Again, undefined aspects may be ***`deferred`*** to future refinements

FORM-L - Fixing Ambiguities in Data Model

system of interest

actual entity of the physical world

```
main Object #system "A University-wide information management system" begin
  Department nursingDepartment "One of the Departments";
  ...
end system;
```

FORM-L - Fixing Ambiguities in Data Model

system of interest

actual entity of the physical world

```
main Object #system "A University-wide information management system" begin
  Department nursingDepartment "One of the Departments";
  ...
end system;
```

part of system environment

actual entity of the physical world

```
external abstract Class #User "of the system";
Class PA "Program Administrator" extends User;
Class NSM "Nursing Staff Member" extends User;
```

also actual entities
also part of system environment

FORM-L - Fixing Ambiguities in Data Model

system of interest

actual entity of the physical world

```
main Object #system "A University-wide information management system" begin
  Department nursingDepartment "One of the Departments";
  ...
end system;
```

part of system environment

actual entity of the physical world

```
external abstract Class #User "of the system";
Class PA "Program Administrator" extends User;
Class NSM "Nursing Staff Member" extends User;
```

also actual entities
also part of system environment

by default, part of the system of interest

piece of information that might not be accurate wrt the physical world

```
abstract Class @Person "Information about a person relevant to the system";
Class Student extends Person;
Class NursingStudent extends Student;
```

also pieces of information
also part of the system of interest

FORM-L - Fixing Requirements Ambiguities in NL

will not be formalised

```
Requirement ~r15
  "The system shall be able to display a printable summary for individual cohorts, which
   will include the students enlisted, the Program of study, sequence of classes, cohort
   progress through the program, and timeline of completion"
: concretisation is {rr15};
```

built-in attribute

by default, is or will be formalised

```
Requirement rr15
  "The system shall be able to display a printable summary for any cohort in the system"
  "That summary shall include all the students enlisted in the cohort"
  "That summary shall include the program of study of the cohort"
  "That summary shall include the complete sequence of classes of the cohort"
  "That summary shall include the progress of the cohort through its program of study"
  "That summary shall include the timeline of completion of the cohort"
...
```

FORM-L - Fixing Requirements Ambiguities in Formal Language

```
begin

external Event @eRequestForCohortSummary (Cohort cohort) "issued by a user" is deferred;

EReport @eReport "Issuance of a report" is deferred;
end rr15;
```

issued by system environment

a built-in class

an extension of class *Event*

by default, issued by the system of interest

FORM-L - Fixing Requirements Ambiguities in Formal Language

```
begin

external Event @eRequestForCohortSummary (Cohort cohort) "issued by a user" is deferred;

rr15 is  after eRequestForCohortSummary
         within !shortDelay
         achieve eReport;

EReport @eReport "Issuance of a report" is deferred;
end rr15;
```

issued by system environment

a built-in class

time limit

will be fully determined later in design

an extension of class *Event*

by default, issued by the system of interest

FORM-L - Fixing Requirements Ambiguities in Formal Language

```
begin
Information @infoOnStudents(Cohort cohort) is deferred;
Information @infoOnProgram (Cohort cohort) is deferred;
Information @infoOnClasses (Cohort cohort) is deferred;
Information @infoOnProgress(Cohort cohort) is deferred;
Information @infoOnTimeline(Cohort cohort) is deferred;

external Event @eRequestForCohortSummary (Cohort cohort) "issued by a user" is deferred;

rr15 is  after eRequestForCohortSummary
         within !shortDelay
         achieve eReport such that { infoOnStudents(bop.cohort),
                                     infoOnProgram (bop.cohort),
                                     infoOnClasses (bop.cohort),
                                     infoOnProgress(bop.cohort),
                                     infoOnTimeline(bop.cohort)
                                   } in contents;

EReport @eReport "Issuance of a report" is deferred; // its contents is a set of Information
end rr15;
```

pieces of information that shall be part of the report

issued by system environment

a built-in class

time limit

will be fully determined later in design

an extension of class *Event*

by default, issued by the system of interest

FORM-L - Modelling Open-Ended Requirements

```
Requirement ~r20
  "The system will notify affected parties when changes occur affecting cohorts, including
   but not limited to changes to the sequence for a cohort's program of study and changes
   to a given week's schedule (lab cancelled this week due to instructor illness)";
```

FORM-L - Modelling Open-Ended Requirements

```
Requirement ~r20
  "The system will notify affected parties when changes occur affecting cohorts, including
   but not limited to changes to the sequence for a cohort's program of study and changes
   to a given week's schedule (lab cancelled this week due to instructor illness)";
```

```
Requirement rr20a
  "The system shall notify affected parties when changes to the sequence for a cohort's
   program of study occur"
  is ...;

Requirement rr20b
  "The system shall notify affected parties when changes to a given week's schedule occur"
  is ...;
```

FORM-L - Modelling Open-Ended Requirements

```
Requirement ~r20
  "The system will notify affected parties when changes occur affecting cohorts, including
   but not limited to changes to the sequence for a cohort's program of study and changes
   to a given week's schedule (lab cancelled this week due to instructor illness)"
begin
  concretisation is deferred;
  Guard !g20 is ensure {rr20a, rr20b} in concretisation;
end r20;
```

will be fully determined later in design

```
Requirement rr20a
  "The system shall notify affected parties when changes to the sequence for a cohort's
   program of study occur"
  is ...;

Requirement rr20b
  "The system shall notify affected parties when changes to a given week's schedule occur"
  is ...;
```

Conclusion

- NL processing / AI could help in the identification of ambiguities in requirements and proposing consistent disambiguation solutions
- Fixing them completely automatically requires understanding and consensus-making abilities probably not in the reach of current automatic tools and technologies
- The paper makes proposals for tools to assist the disambiguation process, at least for the English language

Thank you for your attention

Any questions?