Automatic Requirements Analysis, Validation and Verification

Anthony J H Simons
University of Sheffield

25th June 2018

FES Context

- The Future Engineering System (FES) aims to bring model-based systems engineering (MBSE) to aerospace design projects.

- One area which currently lacks formality is the validation, verification and tracking of aerospace engineering requirements.

- Omissions or ambiguities in requirements may be the cause of errors propagated into the design lifecycle, which are not detected until much later.

Example

- Given the natural English requirement:

  “The engine shall produce a maximum thrust no less than 350kN at sea level.”

- Some entities and properties are identifiable:
  - Entity “engine”
  - Property “thrust”
  - Unit “kilonewton”

Constraints

- Some constraints are explicit: *“thrust no less than 350kN”*

  ⇒ thrust ≥ 350 kN

- Some constraints are implicit: *“at sea level”*

  ⇒ altitude(engine, sea-level) → …

- Some constraints are ambiguous or hidden: *“maximum thrust”*

  ? ⇒ thrust ≤ 350 kN

  ? ⇒ throttle(engine, maximum) → …

Natural Language

- Natural language is notoriously difficult to analyse:
  ⇒ lexical, syntactic, semantic ambiguity
  ⇒ co-reference resolution, implicit knowledge
- Three possible approaches
  - Full linguistic analysis (POS-tagging, syntax parsing, named entity resolution, dependency analysis…)
  - Restricted syntax languages (Attempto Controlled English, ASD Simplified Technical English, …)
  - Form-driven template filling (syntax-directed editors)

Restricted Syntax

- EARS (Easy Approach to Requirements Syntax) originally proposed by Rolls-Royce

- Used successfully to rewrite requirements from the Certification Specification for Engines (EASA, 2007)

- OARS (Operational Approach to Requirements Specification) developed during FES

- Used to write requirements containing operational constraints, to be used later during testing

EARS Templates

EARS (Mavin, et al., 2009; Mavin & Wilkinson, 2010) recommends 5 template styles:

Requirement ::= Ubiquitous | Event-Driven | Fault-Handling | State-Driven | Optional-Feature

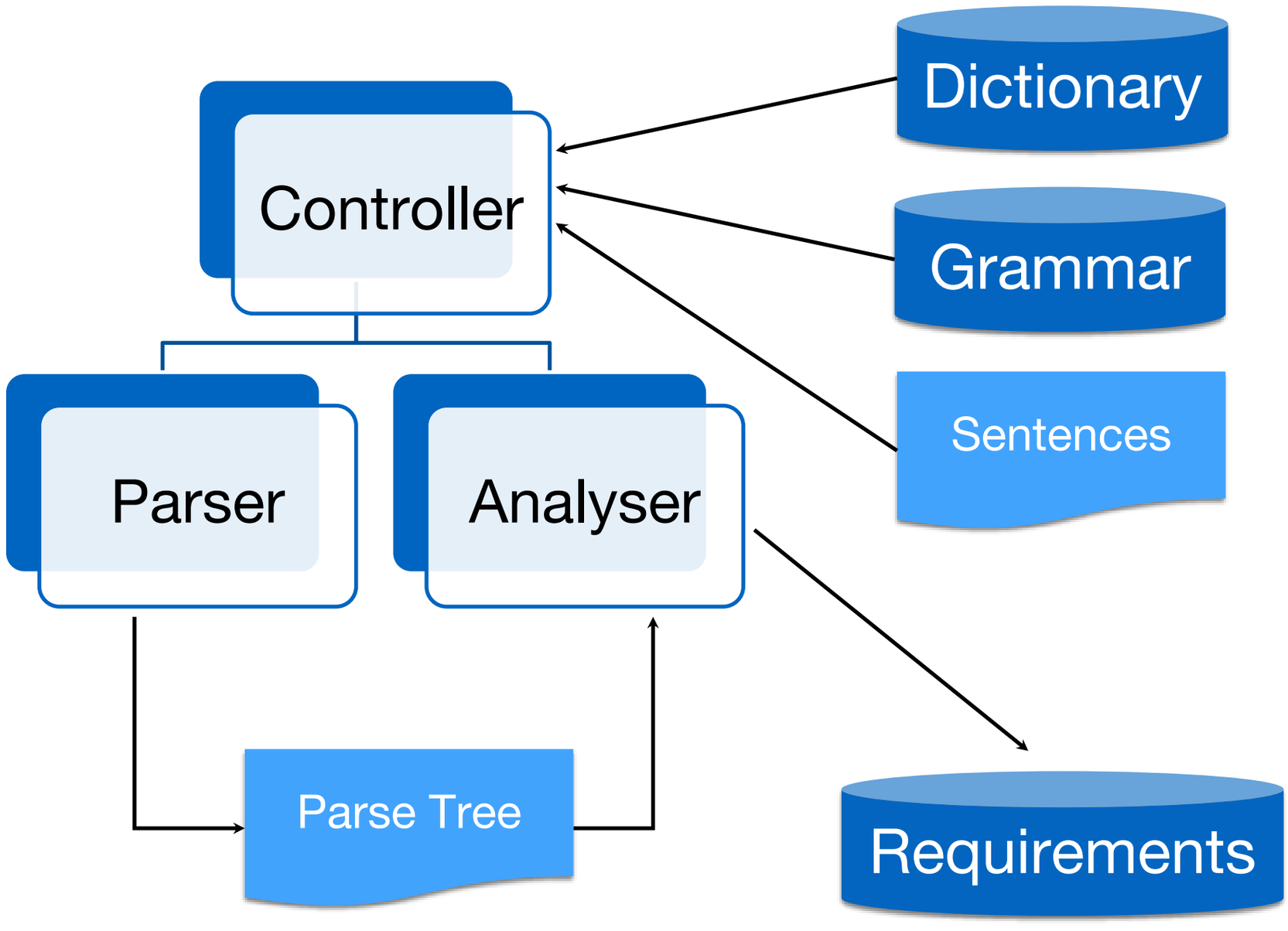
Ubiquitous ::= “THE” System “SHALL” Response
Event-Driven ::= “WHEN” Trigger-Condition, Requirement
Fault-Handling ::= “IF” Fault-Condition “THEN” Requirement
State-Driven ::= “WHILE” State-Condition, Requirement
Optional-Feature ::= “WHERE” Feature-Present, Requirement

EARS Applied

Original CS-E 50 text	Simplified EARS format	Requirement type
“It must be substantiated by tests, analysis or a combination thereof that the Engine Control System performs the intended functions in a manner which does not create unacceptable thrust or power oscillations.”	“THE Engine Control System SHALL not cause unacceptable thrust or power oscillations.”	Ubiquitous
“It must be demonstrated that, when a Fault or Failure results in a change from one Control Mode to another, or from one channel to another, or from the Primary System to the Back-up System, the change occurs so that the Engine does not exceed any of its operating limitations.”	“WHEN the Engine Control System changes Operational Mode, THE Engine Control System SHALL maintain the engine within approved operational limits.”	Event-Driven
“Single Failures leading to loss, interruption or corruption of Aircraft-Supplied Data must not result in a Hazardous Engine Effect for any Engine.”	“IF a single Failure leads to deficient Aircraft-Supplied Data, THEN the Engine Control System SHALL not cause a Hazardous Engine Effect.”	Fault-Handling

(Mavin, et al., 2009)

OARS Parser/Analyser



OARS Details

- OARS sentences have a single main clause (with adjuncts), expressing an *operational constraint*

- Parser: is an active chart parser (Earley, 1970) that finds all ranked parses
  - Uses a bespoke grammar of OARS sentences
  - Uses a bespoke dictionary of aerospace and physics terms

- Analyser: is a rule-based model transformation based on ReMoDeL (Simons, 2010)
  - Uses a bespoke set of parse tree to logical predicate transformation rules

OARS Sentences

Assumption: ubiquitous, with operational constraint

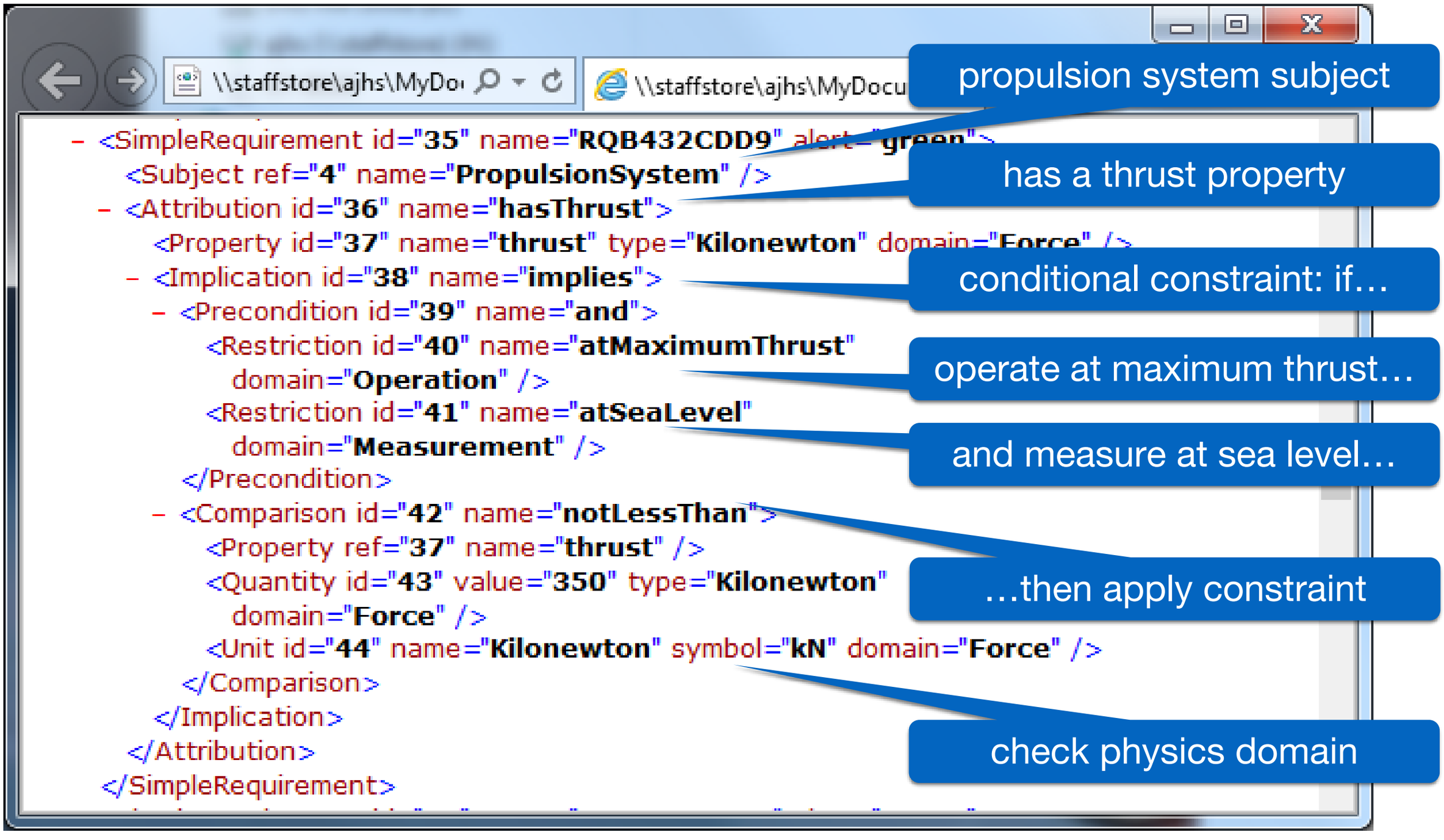
- The propulsion system life shall have a minimum time between overhaul of 15000 hours under nominal operating conditions.
- The total propulsion system weight shall be no greater than 6000kg.
- The propulsion system unit cost of production shall be less than $10,000,000.
- The propulsion system shall have a maximum thrust no less than 350kN at sea level.
- The propulsion system shall have an overall efficiency of no less than 90%.
- The thrust specific fuel consumption of the propulsion system shall be greater than 15lb/(h.lbf)
- The propulsion system noise shall be less than 90dB at a distance of 1000ft from a ground observer at maximum thrust.
- The propulsion system NOx emissions shall be less than 30g/kN.

EARS Analysis

Original sentences	Revised EARS format
“The propulsion system life shall have a minimum time between overhaul of 15000 hours under nominal operating conditions.”	“WHILE operating under nominal conditions, THE propulsion system life SHALL have a minimum time between overhaul of 15000 hours.”
“The propulsion system shall have a maximum thrust no less than 350kN at sea level.”	“WHILE operating at sea-level, WHILE maximum thrust is being applied, THE propulsion system SHALL have a thrust no less than 350kN.”
“The propulsion system noise shall be less than 90dB at a distance of 1000ft from a ground observer at maximum thrust.”	“WHILE operating at ground-level, WHILE measured at a distance of 1000ft, WHILE maximum thrust is being applied, THE propulsion system noise SHALL be less than 90dB.”

- OARS sentences violate EARS ubiquitous template
- Actually, contain concealed state-driven triggers
- OARS Parser/Analyser is robust enough to handle this

OARS Requirement



Logical Meaning

- Given the natural English requirement:

  “The propulsion system shall produce a maximum thrust no less than 350kN at sea level.”

- OARS Parser/Analyser understands the following:

```
Requirement RQB432CDD9 [green alert]:
    THE subject: PropulsionSystem
    SHALL HAVE a property: thrust, of the type: Kilonewton
    SUCH THAT
        IF Operation is atMaximumThrust
        AND Measurement is atSeaLevel
    THEN the thrust SHALL BE notLessThan 350 Kilonewton
```

More Sentences

- The aircraft will be a small personal or short haul aircraft.
- The average flight time will not ex
- The maximum altitude will not ex
- The aircraft shall sustain a climb-rate of [x] f/s for a minimum duration of [y] minutes.
- The time between landing and next take-off should be no more than 2 hours.
- The aircraft shall be all-electric with a storage energy source.

no operational constraint

missing subject

missing subject

non-numeric values

new interval syntax

relational requirement

multiple requirements

Pathological!

- The aircraft requires 50kN of thrust at take-off and 25kN of thrust at cruise.
- Take-off thrust SHALL be maintainable for a maximum period of 2 minutes.
- The total mass of the propulsion system must be no more than 1500Kg.
- The lift of the system must exceed 5000 flight hours.
- The system must not exceed 15m3.
- The total cost of the system will not exceed £10,000,000 through-life.

multiple requirements

missing subject

passive voice

clashing type domains

missing property

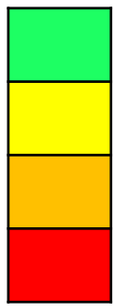
generic subject term

Challenges

- Weasel-words and, or, with hide multiple requirements
  ⇒ need to deal with conjunction, disjunction, intervals
  ⇒ need to decompose non-atomic requirements
- Modal constraints for X minutes, at Y distance conceal nested recursive requirements
  ⇒ need to deal with time, space preconditions
  ⇒ need nested predicate structure, modal logic
- New styles of requirement without explicit operational constraints shall be an X, shall use a Y
  ⇒ need to deal with complements, relationships
  ⇒ need different kinds of predicate (not just attribution)

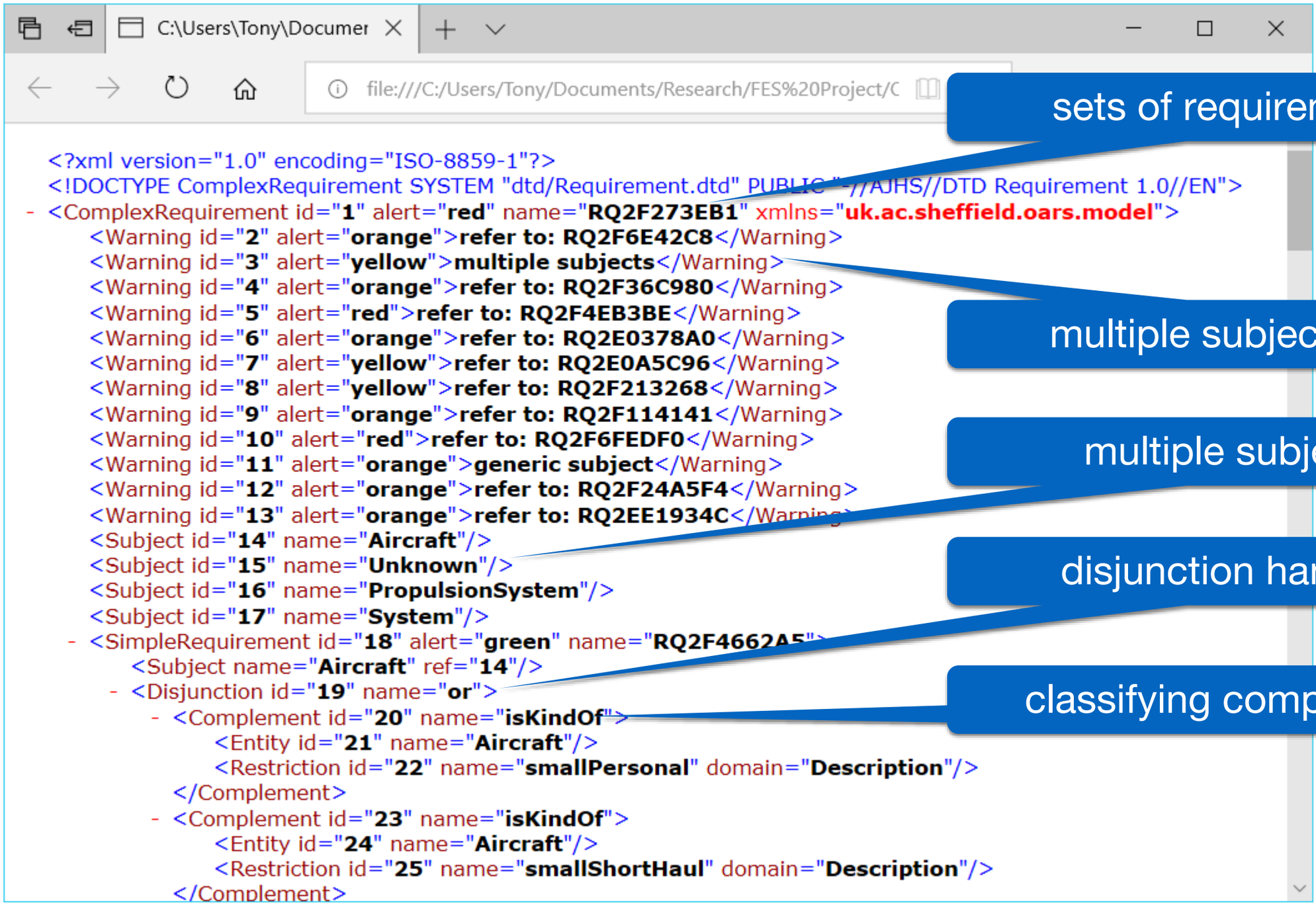
Improvements

- Grammar *misrules* to recognise badly-formed sentences; parses ranked by quality
- Resolution of *and*/*or* coordination structures
- Predicates: attribution, relationship, complement; temporal/spatial quantifiers
- Warning alerts using traffic-light scheme:

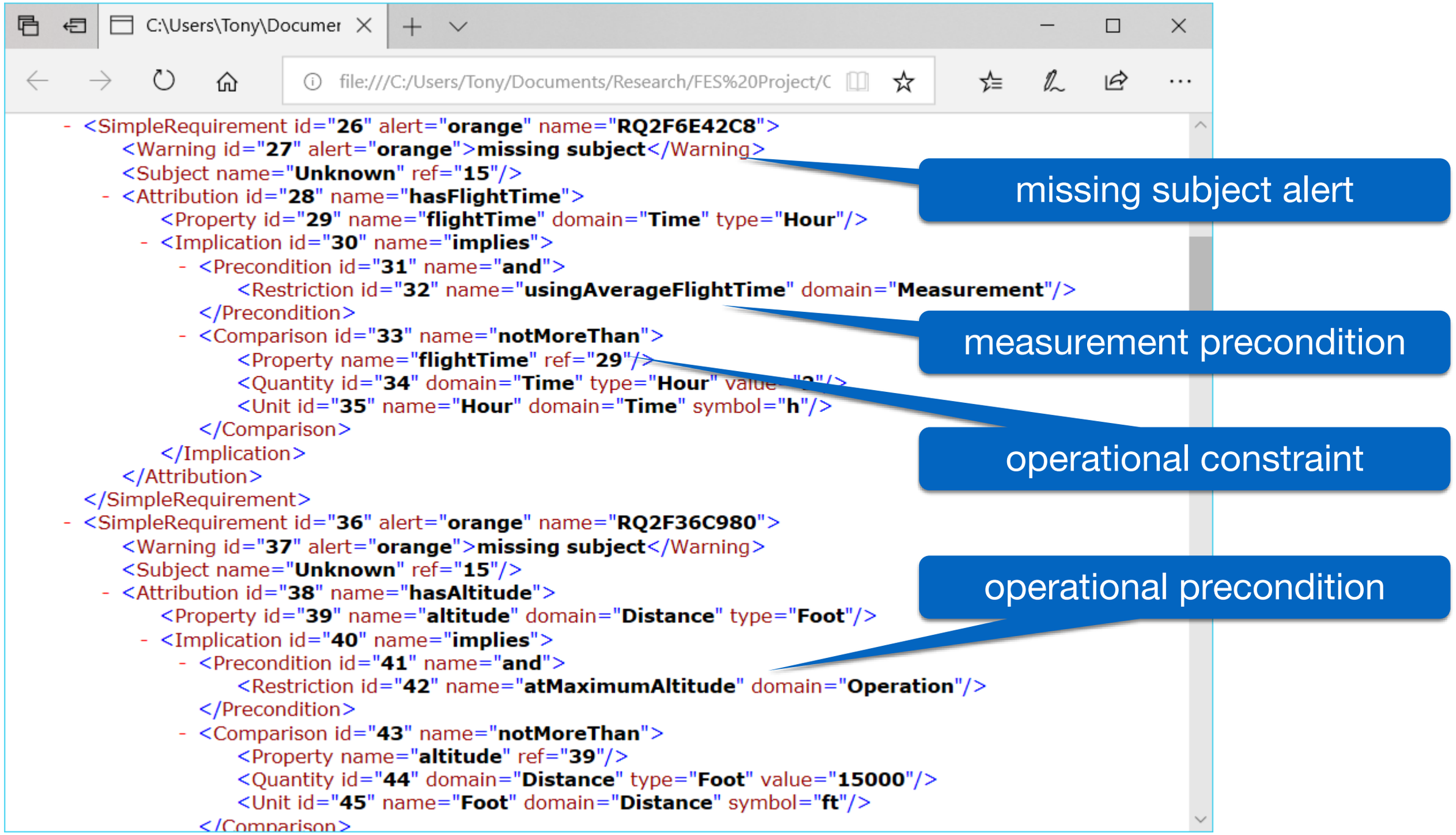


green ⇒ requirement is healthy
yellow ⇒ breaks conventions, but is fixable
orange ⇒ missing info, guess at risk of error
red ⇒ broken beyond ability to repair

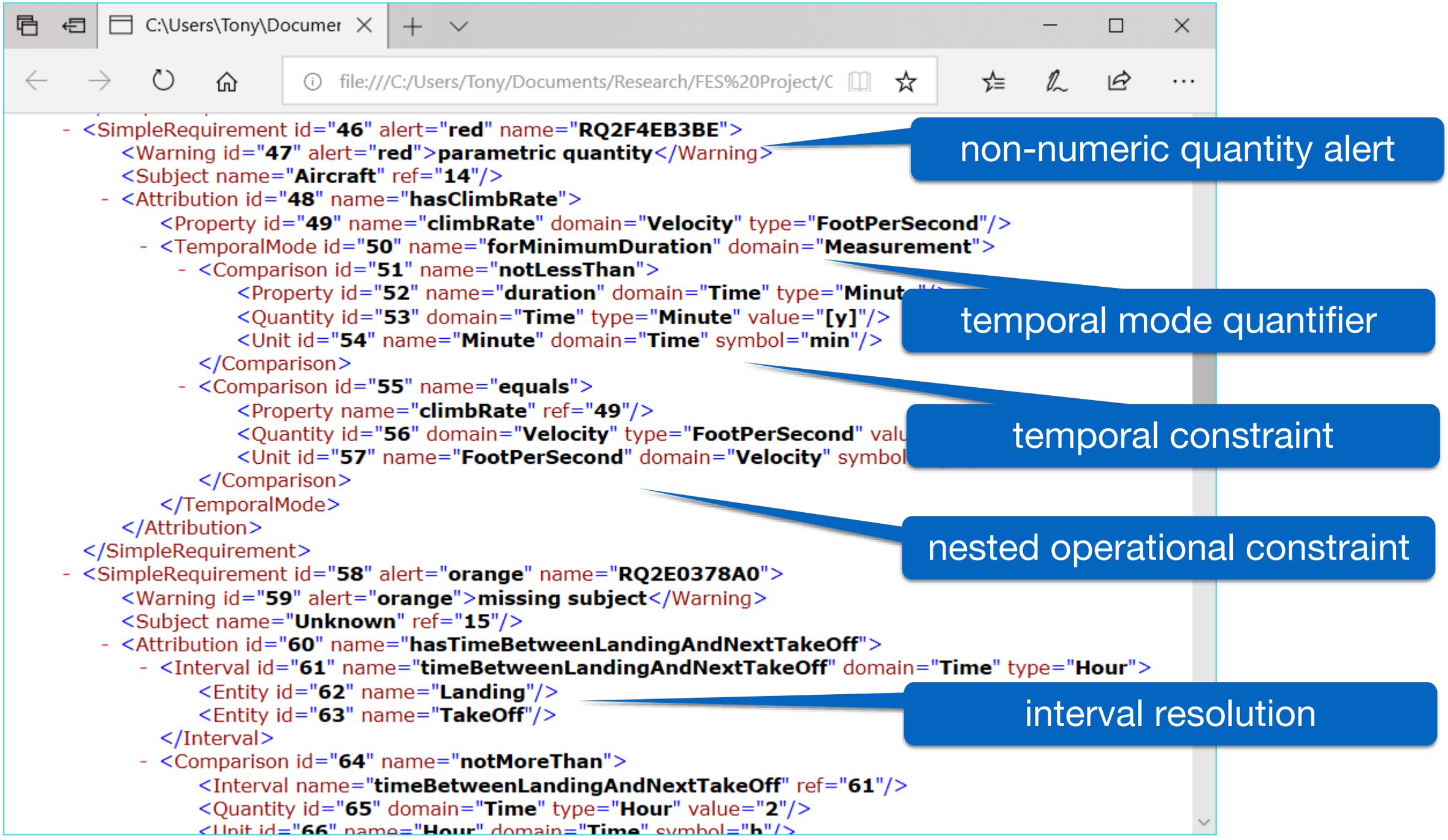
Outputs - 1


C:\Users\Tony\Documer
file:///C:/Users/Tony/Documents/Research/FES%20Project/C
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE ComplexRequirement SYSTEM "dtd/Requirement.dtd" PUBLIC "-//AJHS//DTD Requirement 1.0//EN">
- <ComplexRequirement id="1" alert="red" name="RQ2F273EB1" xmlns="uk.ac.sheffield.oars.model">
<Warning id="2" alert="orange">refer to: RQ2F6E42C8</Warning>
<Warning id="3" alert="yellow">multiple subjects</Warning>
<Warning id="4" alert="orange">refer to: RQ2F36C980</Warning>
<Warning id="5" alert="red">refer to: RQ2F4EB3BE</Warning>
<Warning id="6" alert="orange">refer to: RQ2E0378A0</Warning>
<Warning id="7" alert="yellow">refer to: RQ2E0A5C96</Warning>
<Warning id="8" alert="yellow">refer to: RQ2F213268</Warning>
<Warning id="9" alert="orange">refer to: RQ2F114141</Warning>
<Warning id="10" alert="red">refer to: RQ2F6FEDF0</Warning>
<Warning id="11" alert="orange">generic subject</Warning>
<Warning id="12" alert="orange">refer to: RQ2F24A5F4</Warning>
<Warning id="13" alert="orange">refer to: RQ2EE1934C</Warning>
<Subject id="14" name="Aircraft"/>
<Subject id="15" name="Unknown"/>
<Subject id="16" name="PropulsionSystem"/>
<Subject id="17" name="System"/>
- <SimpleRequirement id="18" alert="green" name="RQ2F4662A5">
<Subject name="Aircraft" ref="14"/>
- <Disjunction id="19" name="or">
- <Complement id="20" name="isKindOf">
<Entity id="21" name="Aircraft"/>
<Restriction id="22" name="smallPersonal" domain="Description"/>
</Complement>
- <Complement id="23" name="isKindOf">
<Entity id="24" name="Aircraft"/>
<Restriction id="25" name="smallShortHaul" domain="Description"/>
</Complement>
sets of requirements
multiple subject alert
multiple subjects
disjunction handled
classifying complement

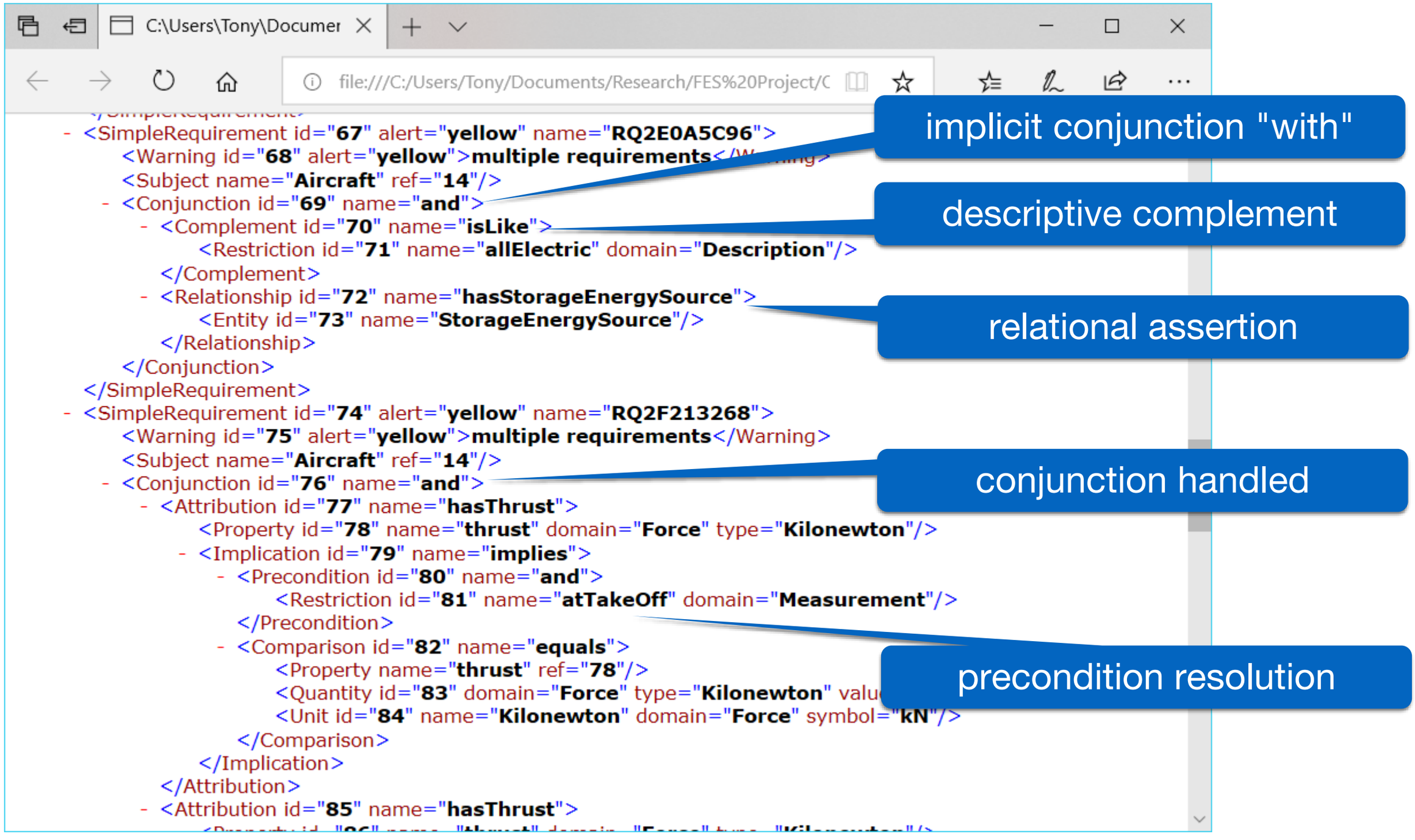
Outputs - 2


C:\Users\Tony\Documer
file:///C:/Users/Tony/Documents/Research/FES%20Project/C
- <SimpleRequirement id="26" alert="orange" name="RQ2F6E42C8">
<Warning id="27" alert="orange">missing subject</Warning>
<Subject name="Unknown" ref="15"/>
- <Attribution id="28" name="hasFlightTime">
<Property id="29" name="flightTime" domain="Time" type="Hour"/>
- <Implication id="30" name="implies">
- <Precondition id="31" name="and">
<Restriction id="32" name="usingAverageFlightTime" domain="Measurement"/>
</Precondition>
- <Comparison id="33" name="notMoreThan">
<Property name="flightTime" ref="29"/>
<Quantity id="34" domain="Time" type="Hour"
<Unit id="35" name="Hour" domain="Time" symbol="h"/>
</Comparison>
</Implication>
</Attribution>
</SimpleRequirement>
- <SimpleRequirement id="36" alert="orange" name="RQ2F36C980">
<Warning id="37" alert="orange">missing subject</Warning>
<Subject name="Unknown" ref="15"/>
- <Attribution id="38" name="hasAltitude">
<Property id="39" name="altitude" domain="Distance" type="Foot"/>
- <Implication id="40" name="implies">
- <Precondition id="41" name="and">
<Restriction id="42" name="atMaximumAltitude" domain="Operation"/>
</Precondition>
- <Comparison id="43" name="notMoreThan">
<Property name="altitude" ref="39"/>
<Quantity id="44" domain="Distance" type="Foot" value="15000"/>
<Unit id="45" name="Foot" domain="Distance" symbol="ft"/>
</Comparison>
missing subject alert
measurement precondition
operational constraint
operational precondition

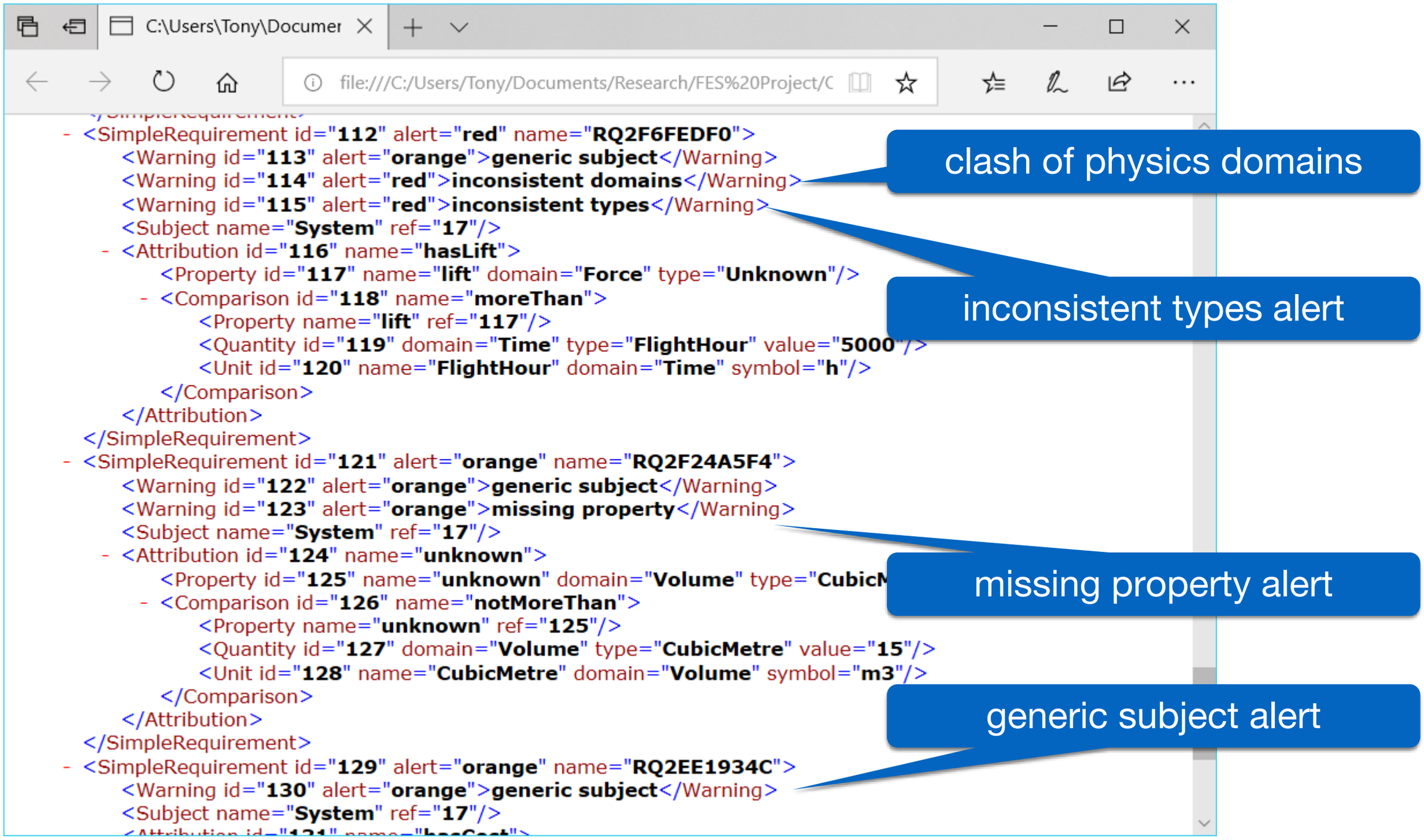
Outputs - 3


C:\Users\Tony\Documer
file:///C:/Users/Tony/Documents/Research/FES%20Project/C
- <SimpleRequirement id="46" alert="red" name="RQ2F4EB3BE">
<Warning id="47" alert="red">parametric quantity</Warning>
<Subject name="Aircraft" ref="14"/>
- <Attribution id="48" name="hasClimbRate">
<Property id="49" name="climbRate" domain="Velocity" type="FootPerSecond"/>
- <TemporalMode id="50" name="forMinimumDuration" domain="Measurement">
- <Comparison id="51" name="notLessThan">
<Property id="52" name="duration" domain="Time" type="Minut
<Quantity id="53" domain="Time" type="Minute" value="[y]"/>
<Unit id="54" name="Minute" domain="Time" symbol="min"/>
</Comparison>
- <Comparison id="55" name="equals">
<Property name="climbRate" ref="49"/>
<Quantity id="56" domain="Velocity" type="FootPerSecond" valu
<Unit id="57" name="FootPerSecond" domain="Velocity" symbol
</Comparison>
</TemporalMode>
</Attribution>
</SimpleRequirement>
- <SimpleRequirement id="58" alert="orange" name="RQ2E0378A0">
<Warning id="59" alert="orange">missing subject</Warning>
<Subject name="Unknown" ref="15"/>
- <Attribution id="60" name="hasTimeBetweenLandingAndNextTakeOff">
- <Interval id="61" name="timeBetweenLandingAndNextTakeOff" domain="Time" type="Hour">
<Entity id="62" name="Landing"/>
<Entity id="63" name="TakeOff"/>
</Interval>
- <Comparison id="64" name="notMoreThan">
<Interval name="timeBetweenLandingAndNextTakeOff" ref="61"/>
<Quantity id="65" domain="Time" type="Hour" value="2"/>
<Unit id="66" name="Hour" domain="Time" symbol="h"/>
non-numeric quantity alert
temporal mode quantifier
temporal constraint
nested operational constraint
interval resolution

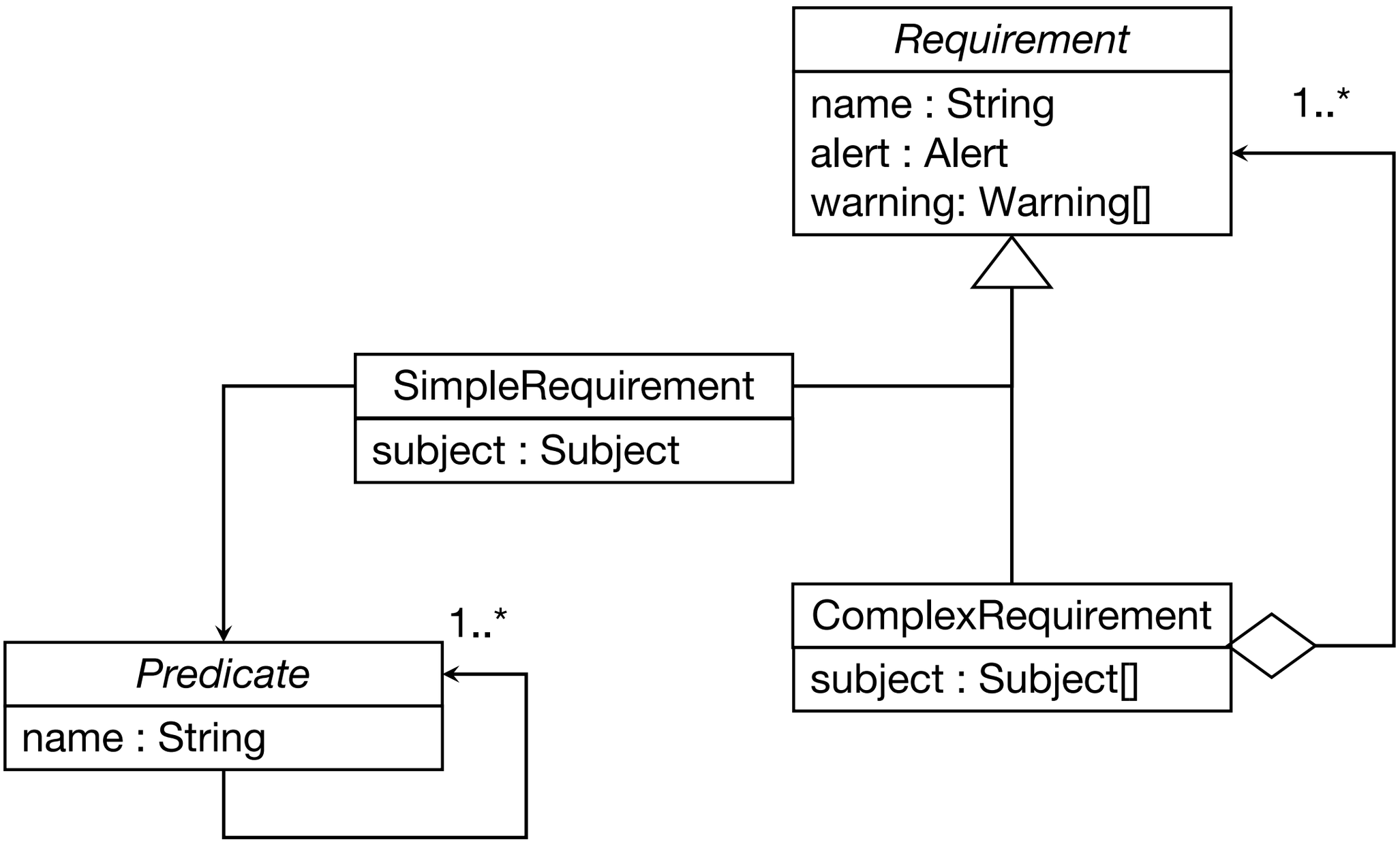
Outputs - 4


C:\Users\Tony\Documer
file:///C:/Users/Tony/Documents/Research/FES%20Project/C
- <SimpleRequirement id="67" alert="yellow" name="RQ2E0A5C96">
<Warning id="68" alert="yellow">multiple requirements</W
<Subject name="Aircraft" ref="14"/>
- <Conjunction id="69" name="and">
- <Complement id="70" name="isLike">
<Restriction id="71" name="allElectric" domain="Description"/>
</Complement>
- <Relationship id="72" name="hasStorageEnergySource">
<Entity id="73" name="StorageEnergySource"/>
</Relationship>
</Conjunction>
</SimpleRequirement>
- <SimpleRequirement id="74" alert="yellow" name="RQ2F213268">
<Warning id="75" alert="yellow">multiple requirements</Warning>
<Subject name="Aircraft" ref="14"/>
- <Conjunction id="76" name="and">
- <Attribution id="77" name="hasThrust">
<Property id="78" name="thrust" domain="Force" type="Kilonewton"/>
- <Implication id="79" name="implies">
- <Precondition id="80" name="and">
<Restriction id="81" name="atTakeOff" domain="Measurement"/>
</Precondition>
- <Comparison id="82" name="equals">
<Property name="thrust" ref="78"/>
<Quantity id="83" domain="Force" type="Kilonewton" valu
<Unit id="84" name="Kilonewton" domain="Force" symbol="kN"/>
</Comparison>
</Implication>
</Attribution>
- <Attribution id="85" name="hasThrust">
implicit conjunction "with"
descriptive complement
relational assertion
conjunction handled
precondition resolution

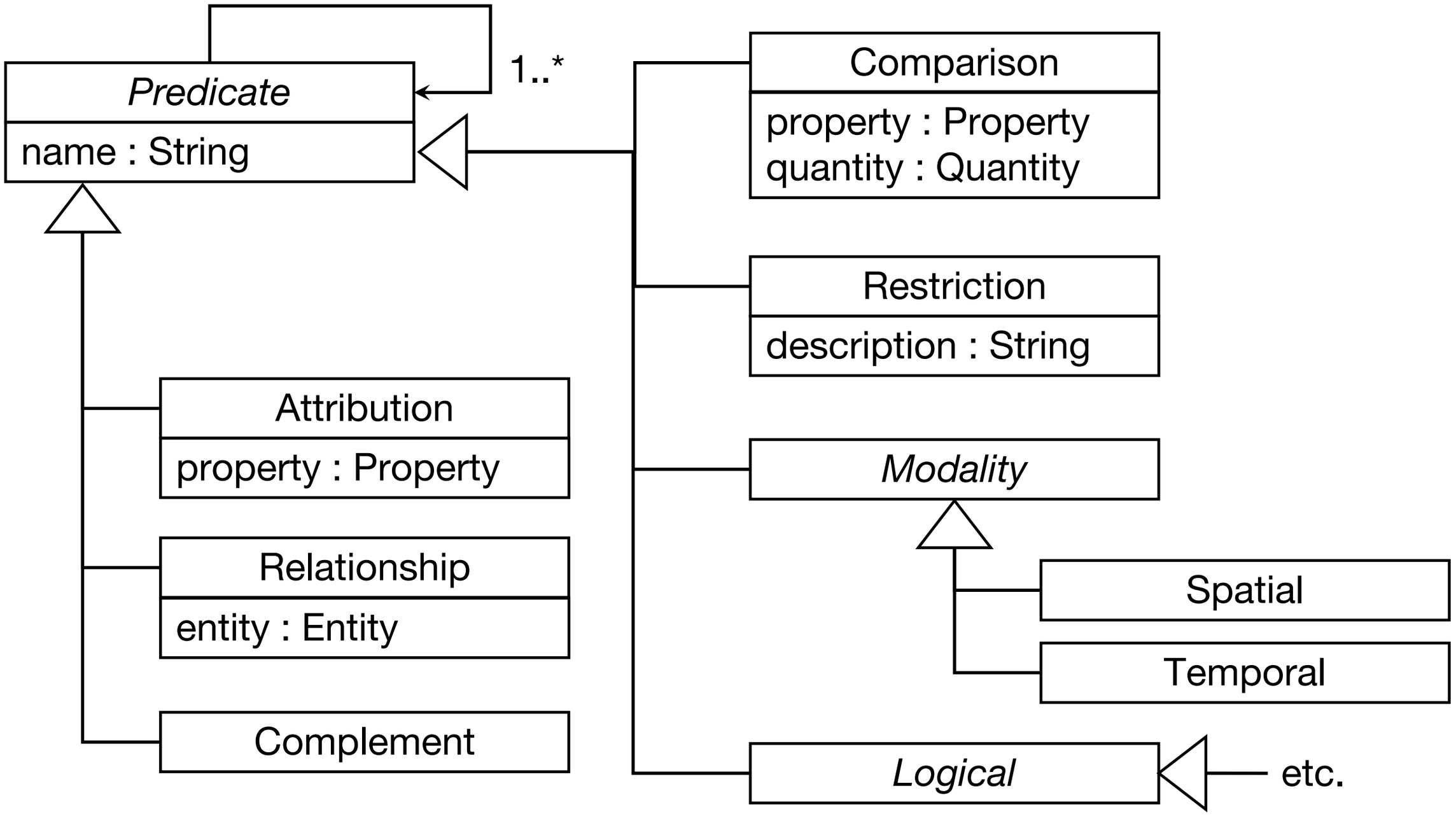
Outputs - 5


C:\Users\Tony\Documer
file:///C:/Users/Tony/Documents/Research/FES%20Project/C
- <SimpleRequirement id="112" alert="red" name="RQ2F6FEDF0">
<Warning id="113" alert="orange">generic subject</Warning>
<Warning id="114" alert="red">inconsistent domains</Warning>
<Warning id="115" alert="red">inconsistent types</Warning>
<Subject name="System" ref="17"/>
- <Attribution id="116" name="hasLift">
<Property id="117" name="lift" domain="Force" type="Unknown"/>
- <Comparison id="118" name="moreThan">
<Property name="lift" ref="117"/>
<Quantity id="119" domain="Time" type="FlightHour" value="5000"/>
<Unit id="120" name="FlightHour" domain="Time" symbol="h"/>
</Comparison>
</Attribution>
</SimpleRequirement>
- <SimpleRequirement id="121" alert="orange" name="RQ2F24A5F4">
<Warning id="122" alert="orange">generic subject</Warning>
<Warning id="123" alert="orange">missing property</Warning>
<Subject name="System" ref="17"/>
- <Attribution id="124" name="unknown">
<Property id="125" name="unknown" domain="Volume" type="CubicM
- <Comparison id="126" name="notMoreThan">
<Property name="unknown" ref="125"/>
<Quantity id="127" domain="Volume" type="CubicMetre" value="15"/>
<Unit id="128" name="CubicMetre" domain="Volume" symbol="m3"/>
</Comparison>
</Attribution>
</SimpleRequirement>
- <SimpleRequirement id="129" alert="orange" name="RQ2EE1934C">
<Warning id="130" alert="orange">generic subject</Warning>
<Subject name="System" ref="17"/>
clash of physics domains
inconsistent types alert
missing property alert
generic subject alert

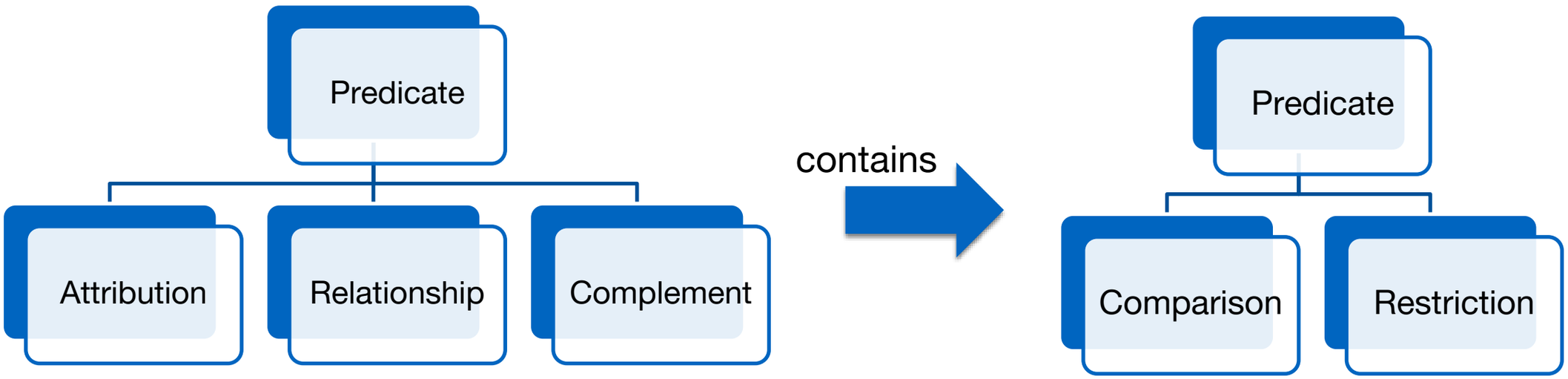
Requirement Model

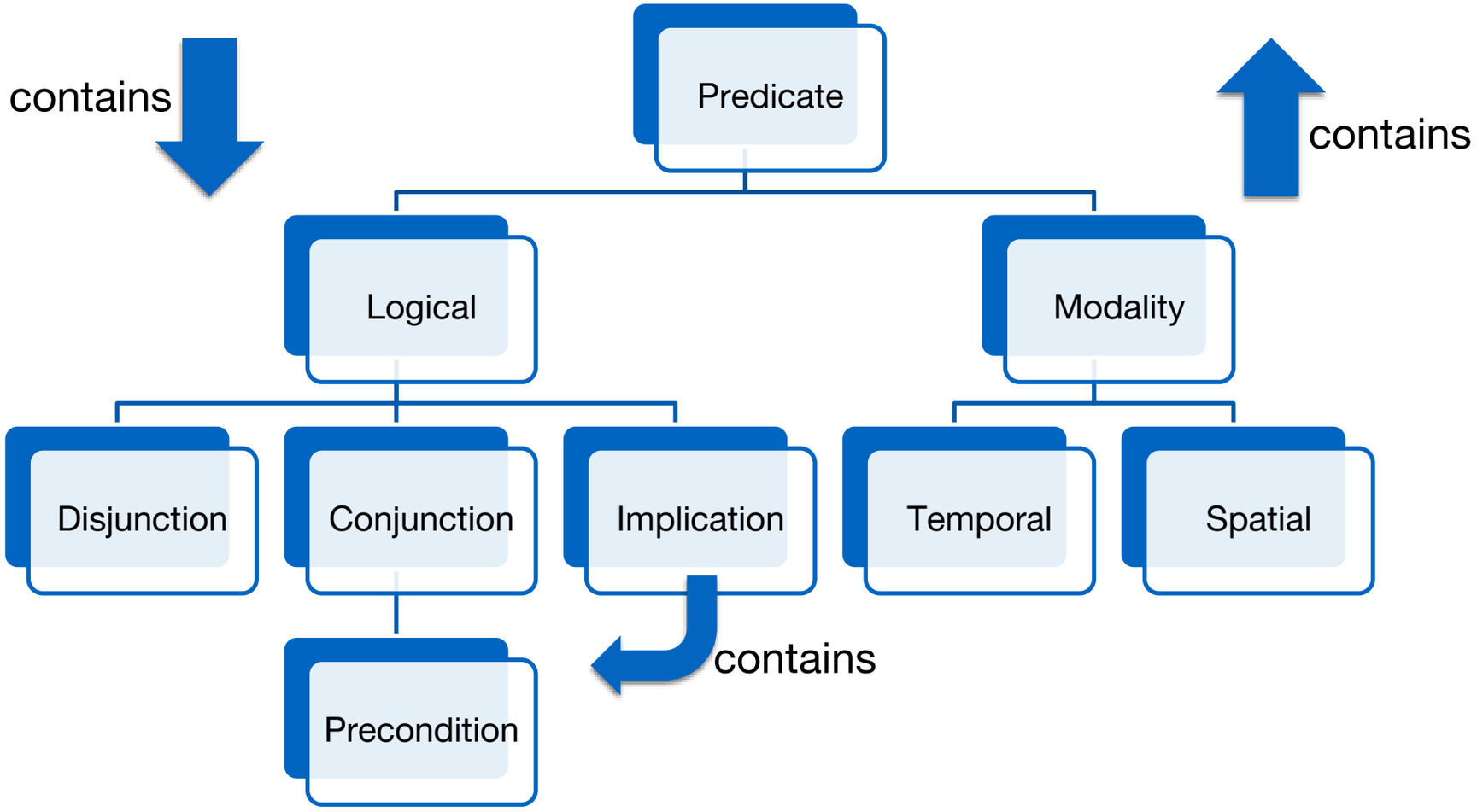


Predicate Model



Combinations





Feedback

- Possible to reflect back to the engineer
  - ⇒ regenerate English from the requirement
  - ⇒ reflects the captured logical meaning
  - ⇒ is this what the engineer really intended?

- Possible to repair certain kinds of fault
  - ⇒ split up non-atomic requirements
  - ⇒ group sets of requirements by subject
  - ⇒ fill in missing subject, property (?)

Validation

Requirement RQ4E5B74F8 [yellow alert]:
Warnings: {multiple requirements}
THE subject: Aircraft
SHALL HAVE a property: thrust, of the type: Kilonewton
SUCH THAT
IF Measurement is atTakeOff
THEN the thrust SHALL BE exactly equal to 50
Kilonewton
AND
SHALL HAVE a property: thrust, of the type: Kilonewton
SUCH THAT
IF Measurement is atCruise
THEN the thrust SHALL BE exactly equal to 25
Kilonewton

did the engineer mean?

did the engineer mean?

Repair

Requirement RQ52286B26 [orange alert]:
Warnings: {generic subject, missing property}
THE subject: System
SHALL HAVE a property: unknown, of the type:
CubicMetre
SUCH THAT the unknown
SHALL BE notMoreThan 15 CubicMetre

m^3 from volume domain

Requirement RQ52286B26 [orange alert]:
Warnings: {generic subject}
THE subject: System
SHALL HAVE a property: volume, of the type: CubicMetre
SUCH THAT the volume
SHALL BE notMoreThan 15 CubicMetre

suggest name *volume*

Test Scripts

Test Schedule	
Test Artefact	Propulsion System
Property Under Test	thrust
Measurement Unit	Kilonewton
Test Procedure	· Set up test artefact and measuring equipment at *sea-level* · Operate test artefact at setting: *maximum thrust* · Measure the thrust obtained under these conditions
Test Objective	thrust >= 350kN

schedule for engineer

Test Schedule	
Test Artefact	Propulsion System
Property Under Test	noise
Measurement Unit	Decibels
Test Procedure	· Set up test artefact and measuring equipment at *ground-level* · Set up measuring equipment at 1000ft from the test artefact · Operate test artefact at setting: *maximum thrust* · Measure the noise obtained under these conditions
Test Objective	noise < 90dB

test physical artefact

Model Tests

```
import org.junit.*;
import static org.junit.Assert.*;

public class PropulsionTestSuite {

  private PropulsionSystem subject;

  public PropulsionTestSuite(PropulsionSystem subject) {
    this.subject = subject;
  }

  @Before public void resetSubject() {
    subject.reset();
  }

  @Test public void testThrustLevel() {
    subject.setMeasurementState(Altitude.SEA_LEVEL);
    subject.setOperatingState(Throttle.MAXIMUM_THRUST);
    int thrust = subject.getThrust(Force.KILONEWTON);
    assertTrue("thrust not less than 350kN", thrust >= 350);
  }

  @Test public void testNoiseLevel() {
    subject.setMeasurementState(Altitude.GROUND_LEVEL);
    subject.setMeasurementDistance(Distance.FEET, 1000);
    subject.setOperatingState(Throttle.MAXIMUM_THRUST);
    int noise = subject.getNoise(SoundPressure.DECIBEL);
    assertTrue("noise less than 90dB", noise < 90);
  }
}
```

generated test code

test model artefact

Partner Integration