

# Inspiration: Regeleditor 2017



Wissenserwerbskomponente

Datei Bearbeiten Ansicht

Vorhandene Regeln: HMI\_Connection\_9\_a

Anderungen speichern Anderungen verwerfen

Name der Regel: HMI\_Connection\_9\_a Regel aktiviert: ☒ Ersteller: Sieber

Kommentar: Verbundene Anschlüsse besitzen die gleichen xy-Koordinaten (Hier: X-Koordinate) [VDI/VDE/NAMUR - Richtlinie 2658 - Blatt 2, Kap. 2.3.8.2 Nr. 9]

Prämisse:

WENN

?IH1 has\_CAEXObjectName "HMI" ☒

Argument1 Rolle Argument 2

1 ↓ Entfernen

UND WENN

?IE1 has\_CAEXInstanceHierarchy ?IH1 ☒

Argument1 Rolle Argument 2

↑ ↓ Entfernen

UND WENN

?IE1 has\_CAEXInternalLink ?IL1 ☒

Argument1 Rolle Argument 2

↑ ↓ Entfernen

UND WENN

?IEA is\_CAEXInternalLink\_PartnerA ?IL1 ☒

Argument1 Rolle Argument 2

1 ↓ Entfernen

UND WENN

?IEB is\_CAEXInternalLink\_PartnerB ?IL1 ☒

Argument1 Rolle Argument 2

Konklusion:

DANN

Failure\_CAEXInternalLink bei ?IE1 fehlerhafter Zustand ☒

Beschreibung:

Diesem InternalElement wurde ein fehlerhafter InternalLink untergeordnet. Die dadurch verbundenen InternalElements stimmen in der X-Koordinate nicht überein [VDI/VDE/NAMUR-Richtlinie 2658 - Blatt 2, Kap. 2.3.8.2 Nr. 9]

Entfernen

© Professur für Automatisierungstechnik  
Institut für Automatisierungstechnik  
Fakultät für Maschinenbau  
Helmut-Schmidt-Universität  
Universität der Bundeswehr Hamburg

- Ziel: „Intuitive“ Erstellung von Regeln
- Feinziele – Allgemein:
  - Aufteilung in Prämisse und Konklusion
  - Berücksichtigung von sämtlichen Klassen, Properties und Individuals der geladenen Ontologie
  - Berücksichtigung von SWRL-Builtins
  - Erstellung von Grundsatz- und Ausnahmeregeln
  - Regeln können erstellt, verändert, argumentiert oder gelöscht werden

- Feinziele Prämisse:
  - Modularer Aufbau der Prämisse durch beliebig viele UND-Verknüpfungen
  - Argumente werden anhand von vorhandenen Klassen oder Individuen bestimmt
  - Vorhandene Properties können ausgewählt werden
  - Variablen können erstellt werden

- Feinziele Konklusion:
  - Modularer Aufbau der Prämisse durch beliebig viele UND-Verknüpfungen (Ich benötige hier strenggenommen nur ein „Zuspitzen“ auf einen Fehlerzustand)
  - Argumente werden anhand von vorhandenen Klassen oder Individuen bestimmt
  - Vorhandene Properties (Fehlerzustände) können ausgewählt werden
  - Variablen der Prämisse können gewählt werden

- Zu vorhandenen Grundsatzregeln soll automatisiert eine Ausnahme erstellt werden können.
- Dies erstellt einerseits eine Ausnahmeregel, andererseits wird die Grundsatzregel verändert
- Beispiel Use-Case:

Grundsatz:

`UAV(?myUAV)^isabove(?mySuchzone)->minaltitude(?myUAV, '20.0'^^xsd:float)`

Ausnahme:

`UAV(UAV_1)^isabove(?mySuchzone)->minaltitude(?myUAV, '10.0'^^xsd:float)`

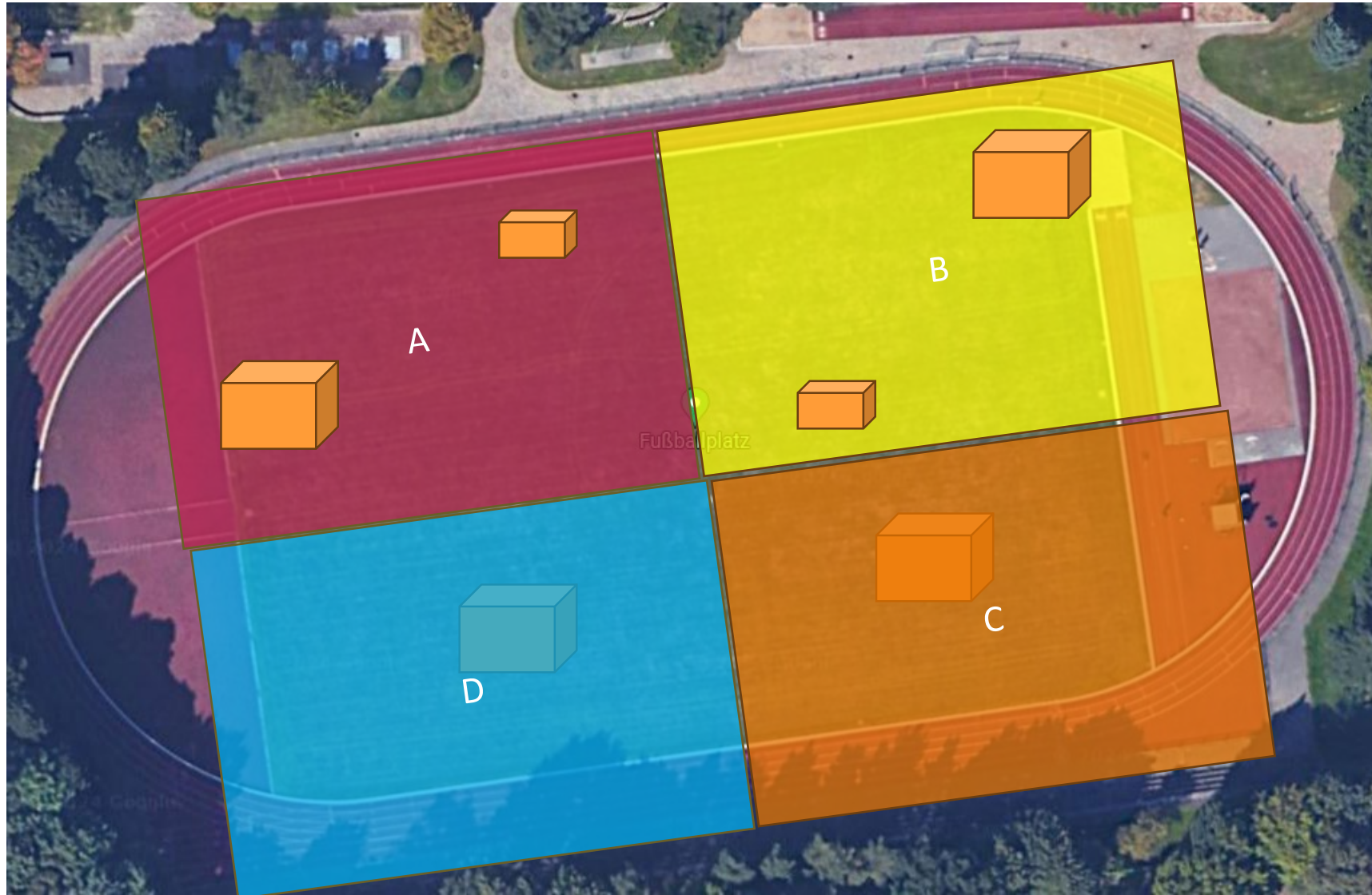
Neuer Grundsatz:

`UAV(?myUAV)^swrlb:notEqual(?myUAV, UAV_1)^isabove(?mySuchzone)->minaltitude(?myUAV, '20.0'^^xsd:float)`

- Implementierungsvorschlag: Zu jeder Regel kann für ein beliebiges Argument in der Prämisse ein Ausnahmeindividual gewählt werden. Der rote Teil der veränderten Grundsatzregel erstellt sich darauf automatisch. Weitere Ausnahmen sind dann nicht zulässig

- Ziel des Use-Case:
  - Aufzeigen der Funktionalität des Konzepts zum Teilsystemgrenzen-übergreifenden Wissensmanagement
  - Wissen der Teilsystemen (4x UAV, 2x UGV) wird durch Kommunikation untereinander / mit einem Leitstand aktuell, verfügbar und korrekt (=gemanaged) gehalten
- Feinziele:
  - Faktenwissen wird zwischen den UAV und UGV automatisch geteilt, sprich kommuniziert (vgl atp-Paper) und aktualisiert
  - Klassenwissen wird zwischen UAV und UGV automatisch geteilt
  - Regelwissen wird zwischen Leitstand und den UAVs geteilt

# Aufbau Use-Case: Search and XXX



- In einem Suchraum (Sportplatz) befinden sich vier Suchzonen (A-D)
- In diesen Suchzonen sollen Pakete (Hilfsgüter, Müll, ...) gesucht (durch UAV) und geborgen (durch UGV) werden
- Gefundene Pakete werden durch Faktenwissen beschrieben und kommuniziert
- Aufgrund veralteter Wissensbasis kennen UGV nur die Klasse „Pakete“, UAV kennen zudem die Subklassen „großes Paket“ und „kleines Paket“. Dieses Klassenwissen wird kommuniziert, um ein Verständnis bei den UGV zu ermöglichen. UGV sind jedoch nicht in der Lage große von kleinen Paketen zu unterscheiden.
- Aufgrund von Wind (?) im Suchraum wird die Suchhöhe von UAV 1 von 10m auf 20m erhöht (-> Ausnahmeregel). Folge ist, dass nun nur noch große Pakete gefunden werden.  
Ausblick: UAV 2 muss in den Suchzonen von UAV 1 unterstützen