

OBJETIVO 1

Alterar o nome dos seguintes predicados em todos os lugares onde é chamado no texto.

Esses predicados aparecem pela primeira vez em **4.1.2 – PREDICADOS**. Contudo, eles são chamados várias vezes ao longo do texto e você vai precisar mudar em todos os lugares onde são chamados. Por enquanto você não precisa mudar os predicados das regras **4.1.4** pq vai precisar mudar elas mais tarde.

PREDICADOS BÁSICOS	VELHOS
<i>adoptsRole</i> (AG, ROLE)	<i>hasRole</i>
<i>hasObligation</i> (ROLE, GOAL)	<i>hasObligation</i> (ROLE, GOAL) (é o mesmo, n precisa mudar)
<i>hasPermission</i> (ROLE, GOAL)	<i>hasPermission</i> (ROLE, GOAL) (é o mesmo, n precisa mudar)
<i>requiresCirc</i> (GOAL, CIRC)	<i>hasCondition</i>
<i>requiresEntity</i> (GOAL, ENT).	<i>hasEntity</i>
<i>possEntityRel</i> (REL, ENTA, ENTB)	<i>thereIsRelation</i>
<i>condition</i>	

PREDICADOS RISCOS E VIOLAÇÕES	VELHOS
<i>entityViol</i> (AG, GOAL, ENT)	<i>violationEntity</i> (AG,GOAL,REL)
<i>relationViol</i> (AG, GOAL, REL)	<i>violationRelation</i> (AG,GOAL,REL)
<i>conditionViol</i> (AG, GOAL, COND)	<i>violationCondition</i> (AG, GOAL, COND)
<i>hasRisk</i> (CIRC, RISK, CONSEQ)	É o mesmo, n precisa mudar
<i>affectsRel</i> (relA, relB)	<i>affectsOtherRelation</i> (relA, relB)
<i>negConseqFor</i> (GOAL, AG, RISK, CONSEQ)	<i>consequenceOfBadEvent</i> (GOAL, AG, RISK, CONSEQ)
<i>possOfNegConseqFor</i> (REL)	<i>possibilityHappensBadEvent</i> (REL)
<i>enabledToStart</i> (AG, GOAL)	<i>ableTryReach</i>

PARAMETER SETTING	VELHOS
<i>isPresent</i> (CIRC)	<i>isPresent</i> (CIRC) É o mesmo, n precisa mudar
<i>starts</i> (AG, GOAL)	<i>tryToReach</i>
<i>stopped</i> (AG, GOAL)	<i>stopIn</i> (Ag,Goal)
<i>reached</i> (GOAL)	
<i>happensNegConseqFor</i> (REL)	<i>happensBadEvent</i>

OBJETIVO 2

Os predicados *instanceOfCond*(CIRC), *instanceOfRel*(CIRC) não existem. Então precisamos resolver esse problema. Então na seção 4.1.2, depois do parágrafo *hasPermission*, faça o seguinte (nexte texto está em linguagem de WORD, mas vc vai ter que colocar na linguagem logico-matematica, para isso vc vai precisar entender de LATEX, mas eh fácil):

O predicado *instanceOfCond*(*circ_k*), *circ_k* \in *Condition* informa que *circ_k* é uma condição, ou seja, pertence ao conjunto *Conditions*. A existência desse predicado é necessária porque alguns raciocínios de violação necessitam verificar se o elemento em análise é uma condição. A mesma situação acontece para as relações. Assim sendo o predicado *instanceOfRel*(*rel_k*) \in *Relation* também deve fazer parte da estrutura do modelo.

OBJETIVO 3

As regras precisam ser refatoradas. Elas estão em 4.1.4 (vai tar em linguagem de WORD, vc tem que colocar na linguagem logico-matematica, é só ver no LATEX como faz e olhar para como as regras antigas estão escritas que vc vai entender).

A regra 29 continua a mesma, vc n precisa mudar ela.

$$\begin{aligned} &hasObligation(\rho_m, g_j) \rightarrow hasPermission(\rho_m, g_j), \\ &\rho_m \in Role \wedge g_j \in Goal \end{aligned} \quad (29)$$

A regra 30

$$\begin{aligned} &hasCondition(g_i, cg_n) \wedge \neg isPresent(c_k) \wedge (c_k \in cg_n) \wedge tryReach(ag_m, g_i) \rightarrow \\ &\quad violationCondition(ag_m, g_i, c_k) \\ &g_i \in Goal, cg_n \subset ConditionGoal, c_k \in Condition, ag_m \in Agent \end{aligned} \quad (30)$$

$$\begin{aligned} &started(ag, goal) \wedge requireCirc(goal, cond) \wedge instanceOf(cond) \wedge \sim isPresent(rel_i) \rightarrow \\ &\quad relationViol(ag, goal, rel) \end{aligned}$$

David, no texto que está em cima da regra 30 vc precisa colocar a seguinte frase:

Essa regra vale apenas para condições, contudo o predicado `requireCirc(goal, circ)` suporta, em seu segundo argumento, elementos que estão contidos em *Condition* como elementos que estão contidos em *Relation*. Para resolver esse problema se faz necessário considerar o predicados *instanceOfCond(circ_k)*.

A regra 31

$$\begin{aligned} &hasRelation(g_i, rg_n) \wedge \neg isPresent(r_k) \wedge (r_k \in rg_n) \wedge tryReach(ag_m, g_i) \rightarrow \\ &\quad violationRelation(ag_m, g_i, r_k) \\ &g_i \in Goal, rg_n \subset RelationGoal, r_k \in Relation, ag_m \in Agent \end{aligned} \quad (31)$$

$$\begin{aligned} &started(ag, goal) \wedge requireCirc(goal, rel) \wedge instanceOfRel(Reli) \wedge \sim isPresent(rel) \rightarrow \\ &\quad relationViol(ag, goal, rel) \end{aligned}$$

David, no texto que está em cima da regra 31 vc precisa colocar a seguinte frase:

Essa regra vale apenas para relações, contudo o predicado `requireCirc(goal, circ)` suporta, em seu segundo argumento, elementos que estão contidos em *Condition* como elementos que estão contidos em *Relation*. Para resolver esse problema se faz necessário considerar o predicados *instanceOfRelation(circ_k)*.

A regra 32

$$\begin{aligned}
& hasEntity(g_i, eg_n) \wedge \neg isPresent(e_k) \wedge (e_k \in eg_n) \wedge tryReach(ag_m, g_i) \rightarrow \\
& \quad violationEntity(ag_m, g_i, e_k) \\
& g_i \in Goal, eg_n \subset EntityGoal, e_k \in Entity, ag_m \in Agent \quad (32)
\end{aligned}$$

$$started(ag, goal) \wedge requireEntity(goal, entity) \wedge \sim isPresent(entity) \rightarrow entityViol(ag, goal, ent)$$

A regra 33

$$\begin{aligned}
& violationCondition(ag_m, g_i, c_k) \wedge hasRisk(c_k, risk_j, cs_m) \rightarrow \\
& \quad consequenceOfBadEvent(g_i, ag_m, risk_j, cs_m) \\
& ag_m \in Agent, g_i \in Goal, c_k \in Condition, risk_k \in Risk, cs_m \in Consequence \quad (33)
\end{aligned}$$

$$conditionViol(ag, goal, cond) \wedge hasRisk(cond, risk, cs) \rightarrow negConseqFor(goal, ag, risk, cs)$$

A regra 34

$$\begin{aligned}
& violationRelation(ag_m, g_i, r_k) \wedge hasRisk(r_k, risk_j, cs_m) \rightarrow \\
& \quad consequenceOfBadEvent(g_i, ag_m, risk_j, cs_m) \\
& ag_m \in Agent, g_i \in Goal, r_k \in Relation, risk_k \in Risk, cs_m \in Consequence \quad (34)
\end{aligned}$$

$$relationViolation(ag, goal, rel) \wedge hasRisk(r, risk, cs) \rightarrow negConseqFor(goal, ag, risk, cs)$$

A regra 35

$$\begin{aligned}
& violationRelation(ag_m, g_i, r_k) \wedge affectsOtherRelations(r_k, r_n) \\
& \quad \rightarrow possibilityHappensBadEvent(r_n) \\
& ag_m \in Agent, g_i \in Goal, r_k, r_n \in Relation, \quad (35)
\end{aligned}$$

$$relationViol(ag, goal, r) \wedge affectsRel(r_k, r_n) \rightarrow possOfNegConseqFor(r_n)$$

A regra 36

$$\begin{aligned}
& possibilityHappensBadEvent(r_k) \wedge happensBadEvent(r_k) \wedge hasRelation(g_i, rg_n) \wedge (r_k \in rg_n) \\
& \quad \wedge hasRisk(r_k, risk_j, cs_m) \wedge tryReach(ag_m, g_i) \\
& \quad \rightarrow consequenceOfBadEvent(g_i, ag_m, risk_j, cs_m) \\
& r_k \in Relation, g_i \in Goal, rg_n \subset RelationGoal, risk_k \in Risk, cs_m \in Consequence \quad (36)
\end{aligned}$$

$$\text{possOfNegConseqFor}(r_n) \wedge \text{started}(ag, goal) \wedge \text{requireCirc}(goal, rel) \wedge \text{isPresent}(r) \wedge \text{happensNegConeqFor}(r) \wedge \text{hasRisk}(r, risk, cs) \rightarrow \text{negConseqFor}(goal, ag, risk, cs)$$

A regra 38 (ele pula mesmo, n tem problema, eh q eu coloco um exemplo da 36)

$$\begin{aligned} & \text{violationEntity}(ag_m, g_i, e_k) \rightarrow \text{stopIn}(g_i) \\ & ag_m \in \text{Agent}, g_i \in \text{Goal}, e_k \in \text{Entity} \end{aligned} \quad (38)$$

$$\text{entityViol}(ag, goal, ent) \rightarrow \text{stopped}(ag, goal)$$

A regra 39

$$\begin{aligned} & \text{consequenceOfBadEvent}(g_k, ag_m, risk_j, cs_m) \rightarrow \text{stopIn}(g_k) \\ & g_k \in \text{Goal}, risk_j \in \text{Risk}, cs_m \in \text{Consequence} \end{aligned} \quad (39)$$

$$\text{negConseqFor}(g, ag, risk, cs) \rightarrow \text{stopped}(ag, g)$$

A regra 40

$$\begin{aligned} & \neg \text{stopIn}(g_k, agg_n) \wedge (ago_n \subset agg_n) \rightarrow \text{isReached}(g_k) \\ & g_k \in \text{Goal}, agg_n \in \text{Agg}, ago_n \in \text{Ago} \end{aligned} \quad (40)$$

$$\sim \text{stopped}(ag, goal) \rightarrow \text{reached}(goal)$$

A regra 41

$$\begin{aligned} & \text{hasRole}(ag_n, \rho_m) \wedge \text{hasPermission}(\rho_m, g_j) \wedge \text{nextGoal}(g_i, g_j) \wedge \text{isReached}(g_i) \\ & \rightarrow \text{ableTryReach}(ag_i, g_j) \\ & ag_i, ag_n \in \text{Agent}, \rho_m \in \text{Role}, g_j \in \text{Goal}, g_i \in \text{Goal} \end{aligned} \quad (41)$$

$$\text{adoptsRole}(ag, role) \wedge \text{reached}(goal) \wedge \text{nextGoal}(goal) \wedge \text{hasPermission}(role, goal) \rightarrow \text{enabledToStart}(ag, goal)$$

A regra 42

$$\begin{aligned} & \text{hasRole}(ag_n, \rho_m) \wedge \text{hasPermission}(\rho_m, g_i) \wedge \text{lastGoal}(g_i, \rho_m) \wedge \text{isReached}(g_i) \\ & \rightarrow \text{stopIn}(g_i) \\ & ag_n \in \text{Agent}, \rho_m \in \text{Role}, g_i \in \text{Goal} \end{aligned} \quad (42)$$

$$\text{start}(ag, goal) \wedge \text{adoptsRole}(ag, role) \wedge \sim \text{hasPermission}(role, goal) \rightarrow \text{permissionViol}(ag, goal)$$

OBJETIVO 4

Na seção 4.1.6 → Predicados Abertos e Fechados.

Mudar a terminologia: “Predicado aberto” por “Predicado de Controle” e “aberto” - > “controle”.

Mudar a terminologia: “Prechado Fechado” por “Predicado de Estrutura”

OBJETIVO 5

Na pasta de arquivos que eu te forneço vai ter um arquivo com o seguinte nome

“dissertacao-2019-03-06-tacla.pdf”

Lá tem uma série de recomendações do meu Professor.

Em azul ele dá o termo pelo qual ele quer q você substitua.

Em vermelho ele quer que você elimine o texto.

Em amarelo envolve mudanças mais complexas onde ele recomenda modificações aprofundadas.

Aplique as mudanças em Azul e Vermelho.

O título em Amarelo,
mude para:

Um Modelo Conceitual para Construção de Cenários de Acidentes em Atividades de Risco.