

UNIVERSIDADE FEDERAL DO RIO GRANDE - FURG
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

Dissertação de Mestrado

**Proposta de uma Ontologia para Tratamento de
Divergência de Dados na Aplicação de Fusão de Sensores
em Plantas Industriais**

Alvaro Luis Pianalto de Freitas

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande - FURG, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação

Orientador: Prof. Dr. Eder Mateus Nunes Gonçalves

Rio Grande, 2017

Ficha catalográfica

F866p Freitas, Alvaro Luis Pianalto de.
Proposta de uma ontologia para tratamento de divergência de dados na aplicação de fusão de sensores em plantas industriais / Alvaro Luis Pianalto de Freitas. – 2017.
103 p.

Dissertação (mestrado) – Universidade Federal do Rio Grande – FURG, Programa de Pós-graduação em Computação, Rio Grande/RS, 2017.
Orientador: Dr. Éder Mateus Nunes Gonçalves.

1. Ontologia 2. Automação 3. Fusão de sensores 4. Indústria
I. Gonçalves, Éder Mateus Nunes II. Título.

CDU 681.5



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO RIO GRANDE
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

DISSERTAÇÃO DE MESTRADO

**Proposta de uma Ontologia para Tratamento de Divergência de Dados na
Aplicação de Fusão de Sensores em Plantas Industriais**

Alvaro Luis Pianalto de Freitas

Banca examinadora:

Prof. Dr. Renato Ventura Bayan Henriques

Prof.ª. Dr.ª. Danúbia Bueno Espíndola

Prof. Dr. Éder Mateus Nunes Gonçalves
Orientador

Dedico esse trabalho aos entes queridos que me acompanharam espiritualmente durante esta jornada, meu pai Luis Fernando, minha irmã Fernanda que enquanto estiveram presentes sempre torceram para que eu chegasse até aqui, e a minha avó Maria Rosa que ajudou a me criar e de quem tenho muita saudade. Além deles, fazem parte dessa conquista aquelas pessoas que estiveram fisicamente a meu lado incentivando e apoiando, minha esposa Luciane, minhas filhas Luísa e Manuela, e também a minha mãe Olga e minha irmã Cláudia. Sem vocês eu não conseguiria.

AGRADECIMENTOS

Agradeço ao meu orientador, professor Eder Mateus Nunes Gonçalves, pela sua colaboração com os ensinamentos e direcionamentos para o desenvolvimento deste trabalho como também aos meus colegas da turma PPGCOMP4 e do grupo de pesquisas 3DCS pelo apoio e subsídios às minhas pesquisas.

*Conhecimento é o
melhor investimento
que podemos fazer.*
— EINSTEIN

RESUMO

FREITAS, Alvaro Luis Pianalto de. **Proposta de uma Ontologia para Tratamento de Divergência de Dados na Aplicação de Fusão de Sensores em Plantas Industriais.** 2017. 104 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

Os atuais perfis de consumo impõem às indústrias de manufatura um grande desafio, que é manterem-se competitivas dentro de uma concorrência acirrada pela preferência de clientes ávidos por novos produtos. Entre as soluções necessárias para esta realidade está o aumento do nível de automação nas linhas de produção, de modo a torna-las flexíveis, ágeis e rápidas na sua configuração/reconfiguração, seguras e também confiáveis na execução e controle dos processos produtivos. Outro fator a ser considerado é o aumento do número de componentes das plantas industriais, assim como a quantidade de informações por eles geradas, sendo que grande parte dessas informações são originadas por sensores, hoje difundidos em larga escala, sendo comumente empregados em conjunto para medir um mesmo fenômeno de um processo produtivo. Neste tipo de aplicação exige-se a fusão dos dados medidos a fim de fornecer uma única informação, precisa e confiável ao sistema de controle da planta. Esse panorama traz um crescimento do nível de complexidade da planta em si, como também dos sistemas de software responsáveis pelo seu controle. Sendo assim, neste trabalho é proposta uma ontologia com a finalidade de representar o conhecimento sobre os componentes de plantas industriais que fazem parte da estrutura física onde a fusão de sensores é empregada. Além disso, outra finalidade é a identificação das divergências de dados entre sensores quando utilizados de forma combinada no controle de processos industriais. Este conhecimento, que é agregado às capacidades dedutivas de um motor de inferência semântica, pode ser usado no apoio aos sistemas supervisão e controle de automação empregados nas indústrias. Para isso, juntamente com a ontologia foram desenvolvidas regras semânticas que asseguraram o funcionamento lógico das restrições no domínio estudado. Como resultado, esta ontologia é capaz de informar automaticamente ao usuário a existência de fusão de sensores, bem como identificar as possíveis divergências de valores nas medições dos sensores envolvidos neste tipo de aplicação.

Palavras-chave: Ontologia, Automação, Fusão de sensores.

ABSTRACT

FREITAS, Alvaro Luis Pianalto de. **Proposal of an Ontology for the Treatment of Data Divergence in the Application of Sensor Fusion in Industrial Plants.** 2017. 104 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

The current consumption profiles impose manufacturing industries a great challenge, which is to remain competitive within a fierce competition for the preference of customers eager for new products. Among the solutions necessary for this reality is the increase of the level of automation in the production lines, in order to make them flexible, agile and fast in their configuration / reconfiguration, safe and also reliable in the execution and control of the productive processes. Another factor to be considered is the increase in the number of components of industrial plants, as well as the amount of information generated by them, and a great part of this information is originated by sensors, which are now widely used and are commonly used to measure the same phenomenon of a productive process. In this type of application it is necessary to merge the measured data in order to provide a single, accurate and reliable information to the plant control system. This scenario brings a growth of the level of complexity of the plant itself, as well as of the software systems responsible for its control. Thus, in this work an ontology is proposed with the purpose of representing the knowledge about the components of industrial plants that are part of the physical structure where the fusion of sensors is used. In addition, another purpose is to identify the divergences of data between sensors when used in combination in the control of industrial processes. This knowledge, which aggregates the deductive capabilities of a semantic inference engine, can be used to support automation supervision and control systems employed in industries. For this, together with the ontology were developed semantic rules that ensured the logical operation of the constraints in the studied domain. As a result, this ontology is able to automatically inform the user the existence of sensor fusion, as well as to identify the possible divergences of values in the sensor measurements involved in this type of application.

Keywords: Ontology, Automation, Sensor Fusion.

LISTA DE FIGURAS

Figura 1	Pirâmide de automação - Disponível em: http://http://2.bp.blogspot.com	20
Figura 2	Diagrama funcional da fusão e integração multissensor na operação de um sistema. Fonte: [58]	25
Figura 3	Relação dos níveis de fusão com os níveis de abstração - Adaptado de [28]	26
Figura 4	Taxonomia apresentada por [47]	27
Figura 5	Proposta de integração da Indústria 4.0. Fonte: [19]	32
Figura 6	Tecnologias da Indústria 4.0. Fonte: [91]	33
Figura 7	Esquema de integração com CPS. Fonte: [70]	35
Figura 8	Dados AutomationML. Fonte: [10]	38
Figura 9	Visão atual da Web. Disponível em: https://www.w3.org/Talks/2002/10/16-rdf/currentweb.png	40
Figura 10	Visão com Web Semântica. Disponível em: https://www.w3.org/2002/Talks/www2002-w3ct-swintro-em/semanticweb.png	40
Figura 11	Visão geral da arquitetura da ontologia ADACOR. Fonte: [52]	46
Figura 12	Visão geral das principais classes e propriedades de objeto da MASON. Fonte: [53]	47
Figura 13	Hierarquia dos equipamentos da planta (a linha laranja mostra a recursividade da propriedade representando uma estrutura de agregação). Fonte: [64]	48
Figura 14	Hierarquia das propriedades dos elementos físicos. Fonte: [64]	48
Figura 15	Hierarquia das classes dos elementos de controle. Fonte: [64]	49
Figura 16	Arquitetura da Formal Pedigree Ontology for Level-One Sensor Fusion. Fonte: [63]	50
Figura 17	Estrutura da ontologia SSN. Fonte: [17]	51
Figura 18	Declaração RDF simples. Fonte: Elaborado pelo autor	58
Figura 19	Declaração RDF simples. Fonte: Elaborado pelo autor	59
Figura 20	Regra SWRL	65
Figura 21	Abrangência da ontologia. Fonte: Elaborado pelo autor	69
Figura 22	Estrutura de Recurso. Fonte: [18]	70
Figura 23	Módulo Element. Fonte: Elaborado pelo autor	72
Figura 24	Propriedades de objetos do módulo Element, as do lado direito são inversas das do lado esquerdo. Fonte: Elaborado pelo autor	72
Figura 25	Módulo SensorFusion. Fonte: Elaborado pelo autor	74

Figura 26	Propriedades dos parâmetros. Fonte: Elaborado pelo autor	75
Figura 27	Propriedades de dados dos tipos de parâmetros. Fonte: Elaborado pelo autor	75
Figura 28	Módulo Divergence. Fonte: Elaborado pelo autor	76
Figura 29	Visão geral da integração dos componentes da proposta. Fonte: Elaborado pelo autor	76
Figura 30	Fluxograma da simulação. Fonte: Elaborado pelo autor	78
Figura 31	Modelagem AML da planta SMAR. Fonte: Elaborado pelo autor . . .	78
Figura 32	Resultado da Regra 1: o lado superior esquerdo mostra os recursos com fusão, do mesmo lado embaixo, mostra os sensores de nível em fusão e no lado direito a consulta <i>DL Query</i> com todos os elementos participantes de fusão de sensores. Fonte: Elaborado pelo autor . . .	80
Figura 33	Resultado da Regra 2: o lado esquerdo mostra os recursos e, realçado em vermelho, as linhas de produção com fusão. No lado direito é mostrada a consulta <i>DL Query</i> com todos os elementos participantes de fusão de sensores, agora com as linhas de produção. Fonte: Elaborado pelo autor	82
Figura 34	Resultado da Regra 3: o lado esquerdo mostra os recursos, as linhas de produção e, realçado em vermelho, as áreas com fusão. No lado direito é mostrada a consulta <i>DL Query</i> com todos os elementos participantes de fusão de sensores, incluindo as áreas. Fonte: Elaborado pelo autor	83
Figura 35	Resultado da Regra 4: o lado esquerdo mostra os recursos, as linhas de produção, as áreas e, realçado em vermelho, os sites com fusão de sensores. No lado direito é mostrada a consulta <i>DL Query</i> com todos os elementos participantes de fusão de sensores, incluindo os sites. Fonte: Elaborado pelo autor	85
Figura 36	Resultado da Regra 5: a consulta <i>DL Query</i> mostra o sensor cujo valor da medição está acima do limite máximo permitido para o recurso onde ocorre a fusão de sensores. Fonte: Elaborado pelo autor	87
Figura 37	Resultado da Regra 6: a consulta <i>DL Query</i> mostra o sensor cujo valor da medição está abaixo do limite mínimo permitido para o recurso onde ocorre a fusão de sensores. Fonte: Elaborado pelo autor	88
Figura 38	Resultado da Consulta SPARQL 1. Fonte: Elaborado pelo autor . . .	89
Figura 39	Resultado da Consulta SPARQL 2. Fonte: Elaborado pelo autor . . .	90
Figura 40	Resultado da Consulta SPARQL 3. Fonte: Elaborado pelo autor . . .	91

LISTA DE TABELAS

Tabela 1	Quadro comparativo entre metodologias avaliadas.	57
Tabela 2	Propriedades de objeto - Módulo Element	73

LISTA DE ABREVIATURAS E SIGLAS

ADACORAD	Aptive holonic CONTROL aRchitecture for distributed manufacturing systems
AS-I	Actuator Sensor Interface
CAEX	Computer Aided Engineering Exchange
CAN	Control Area Network
CIFASIS	Centro Internacional Franco Argentino de Ciencias de la Información y de Sistemas
CLP	Controlador Lógico Programável
CNC	Controlador Numérico Computadorizado
CNI	Confederação Nacional da Indústria
COLLAD	COLLABorative Design Activity
CPS	Cyber-Physical Systems
DAML	DARPA Agent Markup Language
DCOM	Distributed Component Object Model
DDE	Dynamic Data Exchange
DL	Description Logic
DOLCE	Descriptive Ontology for Linguistic and Cognitive Engineering
DP	Decentralized Peripherals
ERP	Enterprise Resource Planning
FIPA	Foundation for Intelligent Physical Agent
FMS	Fieldbus Message Specification
HART	Highway Addressable Remote Transducer
HMS	Holonic Manufacturing System
HSE	High Speed Ethernet
IA	Inteligência Artificial
ICAM	Integrated Computer Aided Manufacturing
IDEF	ICAM DEFinition

IEC	International Electrotechnical Commission
IHC	Interação Humano-Computador
IHM	Interface Homem-Máquina
IP	Internet Protocol
ISA	International Society of Automation
KBSI	Knowledge Based Systems, Incorporation
MAS	Multi-Agent System
MASON	MANufacturing's Semantics ONtology
MES	Manufacturing Execution System
MFI	Multisensor Fusion Integration
NSF	National Science Foundation
ODP	Ontology Design Pattern
OIL	Ontology Inference Layer
OLE	Object Linking and Embedding
OPC	OLE for Process Control
OWL	Web Ontology Language
PA	Process Automation
PC	Personal Computer
RDF	Resource Description Framework
SCADA	Supervisory Control And Data Acquisition
SDCD	Sistema Digital de Controle Distribuído
SSN	Semantic Sensor Network
TA	Tecnologia da Automação
TIC	Tecnologias da Informação e Comunicação
TCP	Transmission Control Protocol
TOVE	Toronto Virtual Enterprise
URI	Universal Resource Identifier
W3C	World Wide Web Consortium
XML	EXtensible Markup Language

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Contextualização	15
1.2	Motivação	18
1.3	Objetivo geral	18
1.3.1	Objetivos específicos	18
1.4	Estrutura do texto	19
2	FUNDAMENTAÇÃO TEÓRICA	20
2.1	Estrutura de um sistema de automação moderno	20
2.2	Tecnologias de automação de sistemas de produção	22
2.2.1	Fusão de sensores	23
2.2.2	Sistemas de controle industrial	28
2.2.3	Visão da Indústria 4.0	31
2.3	Representação do conhecimento	39
2.3.1	Ontologias	41
2.3.2	Ontologias para manufatura	44
2.3.3	Metodologias para construção de ontologias	52
2.3.4	Tecnologias semânticas	56
2.3.5	SWRL - Semantic Web Rule Language	64
3	TRABALHO PROPOSTO	68
3.1	A ontologia	68
3.1.1	Formalização dos elementos da planta	69
3.1.2	Formalização da fusão de sensores	73
3.1.3	Formalização dos parâmetros	74
3.1.4	Formalização das divergências	75
4	VALIDAÇÃO DA PROPOSTA	77
5	CONCLUSÃO	92
5.1	Trabalhos Futuros	93
	REFERÊNCIAS	95
	ANEXO A GRÁFICO DA ONTOLOGIA	103

1 INTRODUÇÃO

1.1 Contextualização

Desde a Revolução Industrial e a conseqüente troca dos sistemas de produção inteiramente artesanais pela utilização de máquinas, que as indústrias de manufatura buscam maior flexibilidade e rapidez na produção, além de oferecer produtos com qualidade a fim de atender as exigências dos consumidores e, também, a redução dos custos de produção. A automação dá suporte a todas essas necessidades e também possibilita o desenvolvimento de sistemas mais eficientes devido ao uso das informações recebidas dos meios de produção.

Por automação entenda-se, em um conceito mais abrangente, como sendo a integração de conhecimentos substituindo a observação, os esforços e as decisões humanas por dispositivos e *softwares* concebidos por meio de especificações funcionais e tecnológicas, através da utilização de metodologias [80].

Durante a década de 1980 o aumento na concorrência internacional obrigou a mudança de foco por parte dos fabricantes, tendo início a flexibilização dos sistemas de produção, a qual foi possível através da utilização de células robotizadas programáveis e controladores numéricos programáveis, que permitiam a produção de diferentes produtos em uma mesma linha de produção, sem alteração dos equipamentos [51]. Dessa forma, os sistemas de manufatura tornaram-se mais dinâmicos e rápidos, tanto na produção, como também nas etapas de tomada de decisão, desenho, projeto, execução e controle. Para [38], neste período houve ampliação do grau de automação das instalações industriais através do desenvolvimento de métodos de identificação, otimização e controle avançado dos processos, multiplexação nos sistemas de comunicação, além do aumento da capacidade de processamento dos computadores e componentes microeletrônicos, o que gerou a expansão das aplicações de sistemas de controle de processo automáticos.

Todavia, [31] considera que os avanços tecnológicos na área da indústria de produção nas últimas décadas foram apenas incrementais comparados com os avanços nas áreas da TI - Tecnologia da Informação, nas comunicações móveis e no comércio eletrônico. Mesmo assim, a evolução da TA - Tecnologia de Automação tornou-se cada vez mais

complexa. A abordagem da utilização de máquinas individuais se encerra com a abordagem de integração de máquinas. Sistemas inteiros de máquinas automatizadas são interligados para propósitos mais abrangentes, tornando-se comuns nas indústrias [86]. Dessa forma, em uma visão mais moderna, além da flexibilização da produção é preciso que haja integração entre os diferentes níveis ligados a produção, e para que isso ocorra de forma plena é preciso que os equipamentos e dispositivos, já existentes e os novos, de diferentes fornecedores, comuniquem-se entre si.

Porém, o reflexo das exigências do mercado consumidor não está apenas no problema da comunicação entre os dispositivos heterogêneos das plantas industriais. Há também a necessidade de que os dados gerados pelos dispositivos que compõem estas plantas tenham o maior nível de integridade e confiabilidade possível, principalmente aqueles que atuam sobre um mesmo meio de forma combinada, como é o caso dos sensores, pois podem existir determinadas situações em que eles forneçam, ao mesmo tempo, dados diferentes relativos a um estado ou ação de um determinado processo na linha de produção. Esse tipo de problema pode fazer com que o sistema responsável pelo controle da planta não faça a interpretação correta da realidade dos processos produtivos, o que pode trazer não só consequências à qualidade dos produtos, como também riscos de segurança aos trabalhadores da linha de produção.

Tome-se como exemplo a seguinte situação: em uma determinada linha de produção existe um tanque para armazenamento de água, sendo que essa é bombeada para o tanque através de uma bomba d'água simples. Nesse tanque encontram-se quatro sensores de nível, onde dois deles atuam em fusão para controle do nível máximo de água no tanque, e os outros dois atuam em fusão para controle do nível mínimo. O nível da água no tanque deverá sempre estar entre os dois níveis (mínimo e máximo). Existe uma variável controlada para a altura do nível e outra variável manipulada indicando o estado da bomba (ligada/desligada). Dessa forma, quando a água atingir o nível mínimo, um dos sensores designados para isso detecta essa condição e o outro não, qual informação está correta? Como o controlador vai atuar, aciona ou não a bomba? Em relação aos outros dois sensores, que são responsáveis pelo controle do nível máximo, o problema é o mesmo, ou seja, se um deles detecta que a água atingiu o nível máximo, o outro sensor não, a bomba deve ser desligada? Em qual deles confiar? Esse exemplo é um exemplo simples, porém retrata uma realidade a respeito da utilização da fusão de sensores e sua importância no controle dos processos produtivos.

O outro problema enfrentado pela indústria manufatureira, está relacionado mais especificamente aos engenheiros de *software* que são responsáveis pela concepção dos sistemas de controle que fazem o gerenciamento dos processos de produção. Isto porque o aumento no número de tecnologias presentes nas plantas industriais, algumas com seus próprios *softwares* embutidos, traz como consequência o crescimento da complexidade do ambiente de produção e das aplicações desenvolvidas pelos engenheiros de *software*,

as quais exigem cada vez mais esforços à sua concepção pois é imprescindível que estas ofereçam integridade e confiabilidade à tomada de decisões nos sistemas de controle e supervisão.

Sendo assim, é possível observar que os problemas aqui apresentados estão basicamente relacionados à integração entre os novos equipamentos instalados nas linhas de produção assim como entre os novos *softwares* embutidos neles e os sistemas supervisórios das plantas industriais. A esse respeito, [55] afirma que o barateamento dos equipamentos de campo, os quais estão cada vez mais inteligentes, traz como resultado a geração de uma grande variedade de dados originados a partir desses equipamentos e que são disponibilizados em tempo real para os sistemas responsáveis pelo controle e supervisão ou para outras aplicações.

Para este panorama descrito, [89] afirma que em domínios onde há aplicação intensa e variada de TIC - Tecnologias da Informação e Comunicação e também de TA, as tecnologias semânticas são cruciais. Ainda segundo o autor, a manufatura auxiliada por ontologia pode: facilitar novas funcionalidades, estruturas e processos organizacionais, simplificar ou habilitar a integração intra e inter-organizacional de diferentes camadas, funções, domínios e processos, além de poder facilitar as soluções orientadas ao conhecimento.

As pesquisas na área da automação da manufatura apontam como solução para o problema da integração entre os diferentes dispositivos e níveis dos sistemas o uso de ontologias já que elas possibilitam representar, não só o conhecimento sobre os processos de produção, como também sobre os equipamentos e produtos através de uma máquina de interpretação responsável pelo controle da linha de produção, sendo que o conhecimento adquirido pode ser usado como ferramenta para administração do sistema de produção [51] [15]. Nesse sentido [80] afirma que caminhamos para, ou já transitamos, em uma era da informação, onde as indústrias para obterem vantagem competitiva, devem cada vez mais capacitarem-se para gerar, partilhar, integrar e gerenciar conhecimentos.

O uso de ontologia objetiva agilizar e otimizar a produção na manufatura, assim como oferecer tratamento adequado aos dados gerados pelos dispositivos utilizados no ambiente de produção, visando também o aumento da qualidade e confiabilidade do *software* de controle. Sobre os dados, [55] diz que o problema não é a quantidade extraída de informação, mas sim a sua qualidade, pois pode acontecer de não serem necessariamente traduzidos em informação de boa qualidade, sendo que uma forma de trabalhar com eles com maior eficiência é a partir da estruturação do conhecimento sobre as plantas industriais, a fim de permitir a extração de informações semânticas, facilitando o processamento dos dados.

1.2 Motivação

As transformações que as indústrias manufatureiras têm enfrentado com os avanços tecnológicos nos últimos anos, fizeram crescer não somente a importância dos dados gerados a partir do chão de fábrica, como também a importância do tratamento às informações que deles resultam. Dessa forma, muitas pesquisas na área de automação industrial buscam soluções que permitam o gerenciamento e tratamento adequado aos dados e informações a fim de torná-los íntegros e confiáveis no suporte as tomadas de decisões envolvendo controle de processos de produção.

Sendo assim, é apresentado aqui um estudo que visa propor uma solução baseada na utilização de uma ontologia para auxílio ao controle de plantas industriais, mais especificamente no que se refere a formalização, captura e reutilização do conhecimento sobre a aplicação de fusão de sensores nos processos produtivos, possibilitando a identificação das estruturas físicas onde este tipo de arquitetura é utilizado, assim como as possíveis divergências de dados nas medições realizadas pelos sensores em tais processos.

1.3 Objetivo geral

Propor uma ontologia para manufatura a fim de servir de suporte à tomada de decisão em sistemas de controle de plantas industriais, disponibilizando para isso o conhecimento sobre a aplicação de fusão de sensores, mapeando os conceitos, sistema de controle, dispositivos e recursos a ela relacionados, possibilitando ainda a identificação de ocorrências de divergência de dados entre os sensores utilizados, permitindo nestes casos também identificar aqueles que possam apresentar falhas de funcionamento. Essa camada deve ser viabilizada por uma especificação no formato XML - *EXtensible Markup Language*.

1.3.1 Objetivos específicos

São definidos os seguintes objetivos específicos:

- estudo sobre metodologias para desenvolvimento de ontologias para o domínio da automação e manufatura
- identificação e classificação dos componentes envolvidos na aplicação da fusão de sensores em plantas industriais
- descrições destes componentes por meio da ontologia
- identificação, através da ontologia, das divergências de dados entre sensores que são utilizados em fusão dentro do ambiente da manufatura

1.4 Estrutura do texto

Para uma melhor compreensão dos aspectos abordados neste trabalho o texto encontra-se organizado da seguinte forma: o capítulo 2 apresenta uma abordagem sobre a importância da integração entre os diferentes níveis da pirâmide de automação industrial e descreve as tecnologias de automação industrial tradicionais e modernas, como as da Indústria 4.0, que têm relação com a abordagem proposta. Neste capítulo também são descritas as diferentes formas de representação do conhecimento através da Web Semântica. O capítulo 3 apresenta o desenvolvimento do trabalho, com a descrição das diferentes etapas envolvidas, sendo que os detalhes da implantação das regras com a SWRL e a demonstração dos resultados obtidos na ontologia são apresentados no capítulo 4. Por fim, no capítulo 5 são apresentadas as conclusões e perspectivas de trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

A abordagem proposta neste trabalho envolve diversos conceitos, tecnologias e ferramentas que integrados visam proporcionar uma solução que possibilite aos sistemas de controle e supervisão utilizados na manufatura, capacidade para suportar as exigências atuais e futuras impostas aos sistemas de produção das indústrias. Sendo assim, aspectos como o tratamento das informações do sistema de produção e a interoperabilidade entre os dispositivos e sistemas automatizados são abordados a seguir, assim como uma ontologia que permite a utilização do conhecimento adquirido sobre os processos de fabricação e sobre os agentes presentes no ambiente de manufatura, favorecendo a tomada de decisões baseadas em informações confiáveis, além de possibilitar rapidez e agilidade na configuração/reconfiguração da linha de produção.

2.1 Estrutura de um sistema de automação moderno

Atualmente, para uma indústria ser competitiva, seja na produção ou no gerenciamento do ciclo de vida do produto, além de contar com controle lógico programável, sistemas de execução de manufatura ou empreendimento no planejamento de recursos, sua estrutura deve estar fundamentada na pirâmide da automação industrial [72], mostrada na figura 1.

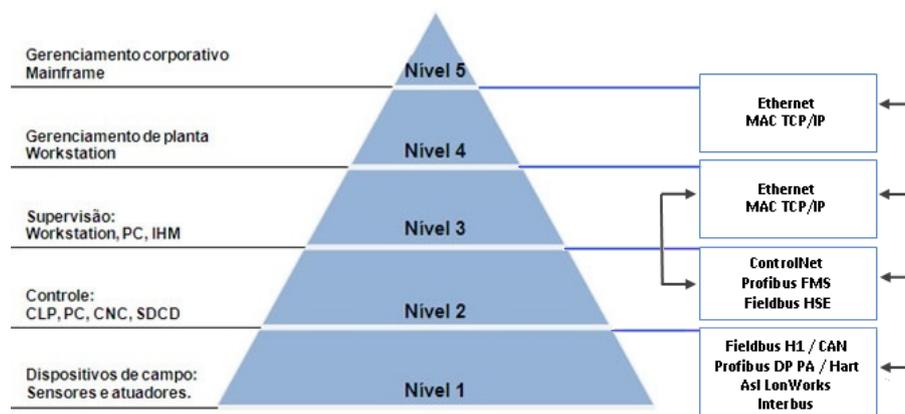


Figura 1: Pirâmide de automação - Disponível em: <http://http://2.bp.blogspot.com>

Os níveis que compõem a pirâmide de automação são os seguintes:

- Nível 1: é o nível mais baixo na hierarquia e inclui atuadores, sensores e componentes de *hardware*.
- Nível 2: inclui os equipamentos responsáveis pelo controle automatizado das atividades da planta industrial. Esses equipamentos são controladores digitais, dinâmicos e lógicos os quais são responsáveis pela execução das etapas de produção na sequência correta e de forma adequada.
- Nível 3: é o nível de supervisão do processo produtivo. Inclui computadores pessoais, IHM - Interface Homem-Máquina, servidores e bancos de dados para o armazenamento dos dados pertinentes a produção. Dado esses que fornecem informações, tais como: índices de qualidade e de produtividade, e também, relatórios e estatísticas.
- Nível 4: onde é feito o planejamento e a programação da produção com base nas informações do sistema corporativo, e onde é realizado o controle dos processos produtivos e a logística de suprimentos.
- Nível 5: é o nível mais alto sendo responsável pela gestão e gerência da produção industrial e por administrar os recursos da empresa, utilizando-se para isso de *softwares* de gestão empresarial que são ferramentas de auxílio a tomada de decisões.

Os níveis 1, 2 e 3 estão diretamente ligados a operação da planta industrial e os níveis 4 e 5 estão relacionadas com ações gerenciais e de suporte à produção. A troca de dados entre os cinco níveis é possível através de diferentes tipos de redes, cada um adequado ao tipo de atividade executada em cada um dos níveis da pirâmide de automação.

No nível 1, chamado de chão de fábrica, do qual fazem parte os sensores e atuadores, conhecidos como dispositivos de campo, são utilizados os seguintes tipos de redes industriais para conexão com o próximo nível: Fieldbus, CAN - *Control Area Network*, Profibus DP - *Decentralized Peripherals*, Profibus PA - *Process Automation*, AS-Interface (*Actuator Sensor Interface*) ou simplesmente AS-I e HART - *Highway Addressable Remote Transducer*.

Na camada de controle (nível 2), composta por CLPs - Controladores Lógicos Programáveis, CNCs - Controladores Numéricos Computadorizados, PCs - *Personal Computers* (industriais e SDCD - Sistemas Digitais de Controle Distribuído, os tipos de conexão com o nível seguinte são: Controlnet, EthernetIP, Foundation-HSE - *High Speed Ethernet*, OLE - *Object Linking and Embedding*, OPC - *OLE for Process Control*, Profibus FMS - *Fieldbus Message Specification*, Profibus DP, Modbus e Profinet.

No nível da supervisão (nível 3), das *Workstations*, PCs e IHM - Interface Homem-Máquina, é onde também estão localizados os bancos de dados com os dados referentes

aos processos e os sistemas supervisórios como o SCADA - *Supervisory Control And Data Acquisition*. Na conexão com o próximo nível são utilizadas redes dos tipos Ethernet, TCP/IP - *Transmission Control Protocol/Internet Protocol*, OPC, DDE - *Dynamic Data Exchange* e DCOM - *Distributed Component Object Model*.

O nível 4 é responsável pelo gerenciamento da planta industrial, e nele também são utilizadas *Workstations* e ferramentas de auxílio ao gerenciamento da produção como o MES - *Manufacturing Execution System*. Para a conexão com o próximo nível são utilizados os mesmos tipos de redes descritas anteriormente no nível 3.

Finalmente o nível 5, onde é realizado o gerenciamento corporativo, através do uso de *Mainframes* e com apoio de *softwares* que realizam o planejamento de negócios e de logística como o ERP - *Enterprise Resource Planning* e que, logicamente, utiliza-se do mesmo tipo de rede do nível anterior.

Na composição da estrutura da pirâmide, há de se considerar a incorporação de novas fontes de informação que surgiram a partir da ampliação da oferta de recursos tecnológicos voltados à produção industrial e, em contrapartida, gerou o aumento do nível de complexidade dos sistemas de produção e das tarefas para sua administração.

Sendo assim, a integração entre os diferentes níveis da pirâmide de automação é fator fundamental na produção industrial, e para que ela ocorra de maneira efetiva e sem prejuízos à automação, não bastam *softwares* dedicados e *hardwares* se eles não tiverem capacidade de superar as barreiras geográficas que separam os recursos dentro de um ambiente de produção distribuído. Sob este panorama, há de se permitir a interoperabilidade entre os novos recursos e aqueles já existentes e, também, que o sistema de manufatura seja reconfigurável de forma rápida possibilitando produzir mais de um produto em uma mesma linha de produção. Ou seja, para esse tipo de sistema de produção há necessidade de interação e colaboração intensa entre diversos e diferentes dispositivos, especializados ou não, máquinas e *softwares* [50].

A necessidade que as indústrias atualmente têm de modernizar e modificar seus métodos de produção a fim de alcançarem a agilidade e flexibilidade (facilidade no reuso de componentes) necessárias para aumentar a sua competitividade no mercado, assim como as dificuldades para que os componentes de um sistema de manufatura sejam interoperáveis, fazem com que pesquisas sejam desenvolvidas objetivando encontrar uma solução que atenda a esses requisitos.

2.2 Tecnologias de automação de sistemas de produção

Inúmeras são as tecnologias atualmente utilizadas nos processos de manufatura sendo que as soluções geradas para esse ambiente normalmente são compostas pelo uso combinado de várias delas. No caso deste trabalho, as tecnologias utilizadas são descritas a seguir.

2.2.1 Fusão de sensores

O aumento do nível de automação na produção busca dar qualidade e segurança aos processos produtivos, sendo que tarefas complexas são executadas por máquinas que independentemente da natureza do processo ou do produto, geram dados sobre o meio em que atuam. Dessa forma, a utilização de sensores, independentemente da aplicação, serve para prover o sistema de controle com informações úteis sobre os recursos e o ambiente sobre o qual atuam [60].

Há muito tempo os sensores são utilizados pelas indústrias de manufatura para o controle de sistemas e para o monitoramento do ambiente de produção. Com o passar do tempo, não só foram desenvolvidos sensores com mais tecnologia embarcada e de diferentes tipos, assim como também novas estratégias para sua utilização com maior confiabilidade e integridade nos dados por eles gerados. Atualmente, muitos dispositivos de natureza diferente (redes de sensores, robôs, câmeras de vídeo,...) são utilizados conjuntamente em diversas tarefas de monitoramento de uma planta industrial.

Muitas são as nomenclaturas encontradas na literatura que referem-se ao uso de conjuntos de sensores que atuam de forma integrada no monitoramento dos ambientes e linhas de produção na manufatura. Para [81] [26], Fusão de Sensores (*Sensor Fusion*), Fusão de Dados (*Data Fusion*), Fusão de Dados Multissensor (*Multi-sensor Data Fusion*) ou Integração Multissensor (*Multi-sensor Integration*) são nomenclaturas com uma única definição, ou seja, trata-se de um processo de combinação dos dados originados em múltiplos sensores, sejam eles de mesma natureza ou não, o qual tem como objetivo fornecer dados com maior qualidade a fim de reduzir falhas nos processos decisórios dos sistemas de controle.

Já [57] [59], assim como [81] [26] também chamam de Integração Multissensor, mas também referem-se a Fusão Multissensor (*Multi-sensor Fusion*), sendo que o termo Integração refere-se ao uso sinérgico da informação fornecida por vários dispositivos sensoriais para auxiliar na realização de uma tarefa por um sistema, enquanto que o termo Fusão está ligado a qualquer uma das fases do processo de integração, isto é, onde existir uma combinação (ou fusão) real de diferentes fontes de informação sensorial em um formato representacional, definição essa que [46] também compartilha. Em [58], as duas nomenclaturas e suas definições são unificadas surgindo a MFI - *Multisensor Fusion Integration* e, segundo os autores, requer conhecimento interdisciplinar na teoria do controle, processamento de sinal, inteligência artificial, probabilidade e estatística, etc.

No que diz respeito a escrita deste trabalho, o termo adotado é o de Fusão de Sensores, visto tratar-se de expressão mais genérica e que representa melhor a proposta aqui apresentada, pois cabe lembrar que a fusão dos dados, definida em [46] como sendo a combinação de dados de múltiplos sensores em um formato comum de representação, ou a fusão das informações propriamente ditas não fazem parte do escopo. Além disso, a escolha por **Fusão de Sensores** é reforçada pela interpretação apresentada por [57] [59],

descrita no parágrafo anterior, a qual expressa que o termo pode ser referido à qualquer fase da combinação de informações de diferentes fontes. Sendo assim, tal afirmação contempla o que aqui está sendo proposto.

A utilização de Fusão de Sensores, assim como de fusão de dados, conforme [58], tem a vantagem de proporcionar percepções complementares, como também informações mais oportunas disponibilizadas por meio do processamento paralelo de dados sensoriais. Segundo [59], a Fusão de Sensores tem como benefício principal fornecer ao sistema de controle informações de maior qualidade, as quais possivelmente referem-se a determinados aspectos do ambiente que não podem ser detectados diretamente por qualquer sensor que opere independentemente.

Além das vantagens apontadas no parágrafo anterior, a Fusão de Sensores possibilita a tomada de decisões com relação a produtividade da planta industrial, como também detectar falhas em tempo real para que os engenheiros de produção venham a ter condições de intervir em qualquer processo para fazer as correções necessárias.

Na visão de [46], na qual são comparadas as características básicas de seres inteligentes, como o sensoriamento e interpretação de dados feito pelos humanos através de seus cinco sentidos (visão, audição, olfato, tato e paladar), com as características da fusão de sensores, e afirma o autor que para obter-se êxito no desenvolvimento de uma "máquina inteligente", com capacidade de interação com o ambiente e de forma semiautônoma ou autônoma tomar decisões, seria fundamental o uso de fusão de sensores e a interpretação de seus dados. Porém, os autores ressaltam a dificuldade em fundir e interpretar os dados, motivo esse que incentiva um grande número de pesquisas. Dessa forma, a fusão de sensores passou a ser componente obrigatório em sistemas inteligentes com um universo amplo de aplicações [57].

Algumas dessas pesquisas [60] [65] [22] [57] [58], adotaram a proposta de [59] que classificaram a Fusão de Sensores e o uso de um único sensor, ao longo do tempo, em níveis diferentes de abstração, conforme figura 2. Para [60], normalmente grande parte dos sensores fornecem dados que podem ser enquadrados em mais de um ou apenas um dos níveis, sendo que informações podem ser fornecidas a um sistema com variados propósitos. Os níveis são os seguintes:

- **Nível baixo** ou *fusão a nível sinal e pixel* - os dados dos sensores ao nível de sinal (medição) são a simples representação da saída ou do estado da planta. São dados brutos oriundos das entradas e que são combinados gerando dados mais precisos (ruído reduzido) do que as entradas individuais [65]. Com relação aos dados brutos, [57] [58] dizem que há necessidade deles serem sincronizados e adaptados antes do processo de fusão, isso em razão dos diferentes tipos de sensores e suas propriedades de amostragem, tais como a taxa de amostragem. Já a fusão a nível de pixel possibilita a melhora da informação associada a cada pixel de uma imagem, que resulta da associação de múltiplas imagens.

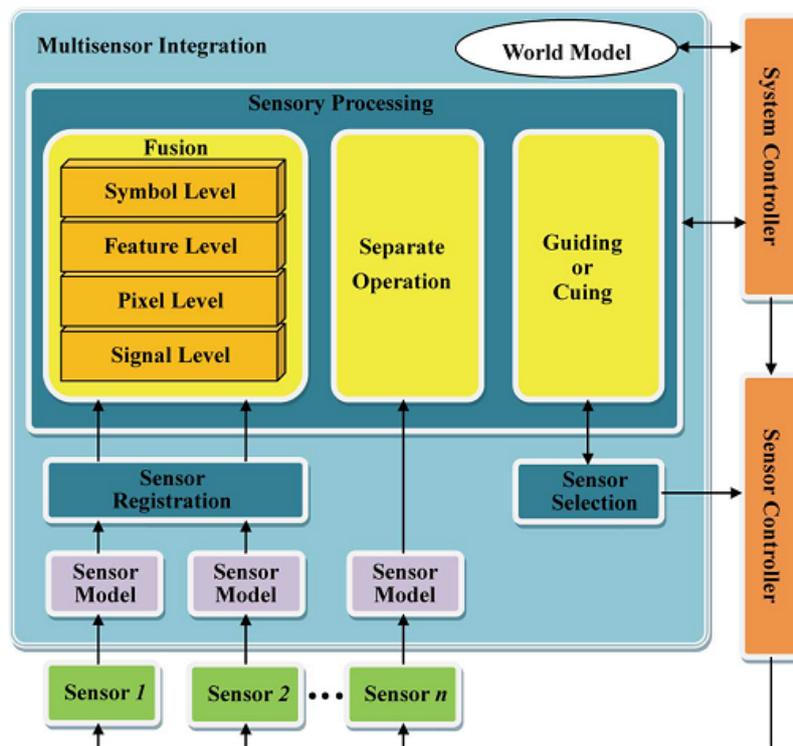


Figura 2: Diagrama funcional da fusão e integração multissensor na operação de um sistema. Fonte: [58]

- **Nível médio** ou *fusão a nível de características* - os dados fundidos neste nível são características extraídas de sinais e imagens. A fusão é realizada através da concatenação dos pontos de características obtidas de diferentes fontes, resultando em uma característica com maior discriminação
- **Nível alto** ou *fusão ao nível de símbolo* - a informação fundida neste nível é uma representação simbólica dos parâmetros do processo, como as descrições humanas. Em geral, a fusão de alto nível procura processar decisões locais de múltiplos sensores para alcançar uma decisão conjunta, o que pode ser alcançado por sistemas experientes, rede neural adaptativa e lógica difusa. Este tipo de fusão também é referido como fusão de decisão [57] [58].

Em seu trabalho, [57] aponta o uso para cada um dos níveis de abstração dos dados, são eles:

- a fusão em nível de sinal pode ser usada em aplicações em tempo real e pode ser considerada apenas uma etapa adicional no processamento geral dos sinais;
- a fusão a nível de pixel pode ser usada para melhorar o desempenho de muitas tarefas de processamento de imagem, como a segmentação;
- a fusão em nível de característica e a fusão em nível de símbolo podem ser usadas para fornecer um sistema de reconhecimento de objetos com características adi-

cionais que podem ser usadas para aumentar suas capacidades de reconhecimento [59].

Ainda segundo [59], a mudança de níveis de abstração, de baixo para cima, na medida que a informação move-se através da estrutura de reconhecimento, é algo comum em grande parte dos processos de integração de sensores, com a transformação do sinal em informação, algo que ocorre em diferentes etapas de fusão, resultando em representações numéricas e simbólicas cada vez mais abstratas. Esse fenômeno de transformar "sinais em símbolos" também é comum na visão computacional e na inteligência artificial. A figura 3 ilustra a relação dos níveis de fusão com os níveis de abstração.

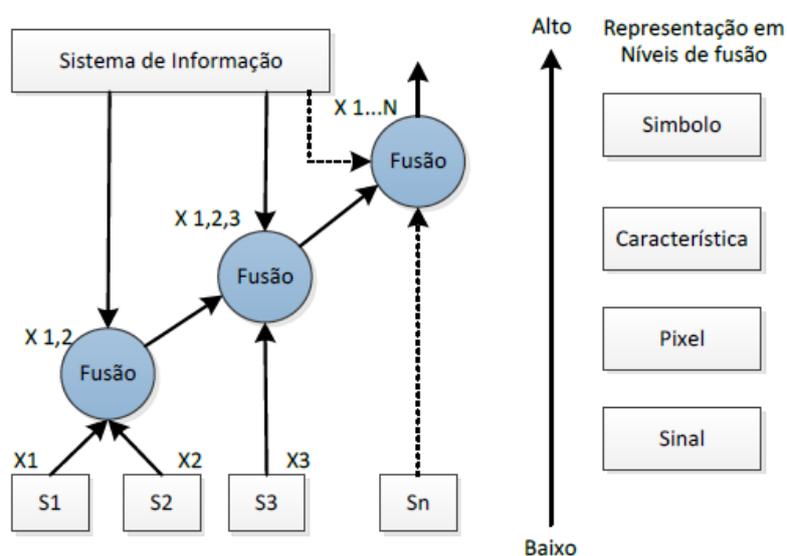


Figura 3: Relação dos níveis de fusão com os níveis de abstração - Adaptado de [28]

A utilização de fusão de sensores tem crescido muito em diversas áreas de aplicação, e também tem evoluído muito ultimamente em razão das pesquisas e das indústrias [29]. O objetivo é aperfeiçoar as técnicas de fusão para tornar as medições mais precisas e consequentemente os processos de produção mais confiáveis e seguros. Porém, a fusão de sensores tem os seus problemas, e não são poucos como afirmam [47], sendo que muitos desses problemas aparecem quando os dados são fundidos, ou a partir da diversidade e imperfeições nas tecnologias ou ainda no domínio da aplicação. Esses problemas devem-se a diferentes razões relativas aos dados dos sensores, tais como incertezas, imperfeições, conflitos [29] [74], imprecisão, incompletude, inconsistência e ambiguidade dos dados [49] [3].

Com relação as imperfeições e a falta de confiabilidade nos dados gerados pelos sensores, de uma maneira geral, têm origem em fatores técnicos e de ruído (ruído ambiental, presença de objetivos desconhecidos, condições meteorológicas, etc.) [29]. Para [49] a imprecisão e os ruídos dos sensores não são os únicos responsáveis pelas incertezas nas suas medições. Há outras razões para isso, como as ambiguidades e inconsistências que

fazem parte do ambiente e a incapacidade que eles têm de distinguir. Já [74], afirma que a fusão de sensores traz consigo distúrbios eletrônicos e mudanças térmicas que influenciam no estado do sistema e que, em virtude disso, a estrutura deve ser capaz de garantir estabilidade e suavidade mesmo que sob influências externas ou não confiáveis, pois sistemas onde há uso de vários sensores o ganho de conhecimento é maior do que aqueles que fazem uso de um único sensor.

Em muitas áreas são utilizados sistemas autônomos onde a fusão de sensores é empregada, sendo que esses sistemas devem possuir capacidade de percepção e interação com o ambiente físico onde se encontram e a ocorrência desses problemas podem resultar em entendimentos errados sobre o comportamento do sistema, assim como sobre o estado do ambiente, o que pode resultar em tomadas de decisões erradas [3].

Conforme [47], mesmo com a identificação destes problemas nas pesquisas realizadas, ainda não existe um algoritmo de fusão sensorial capaz de atender a todos os problemas aqui mencionados. A figura 4 exibe a taxonomia das metodologias de fusão sensorial apresentada em seu trabalho, onde os algoritmos são classificados com base em um dos quatro problemas apontados pelo autor: imperfeição de dados, correlação de dados, inconsistência de dados, e disparates em formulários de dados.

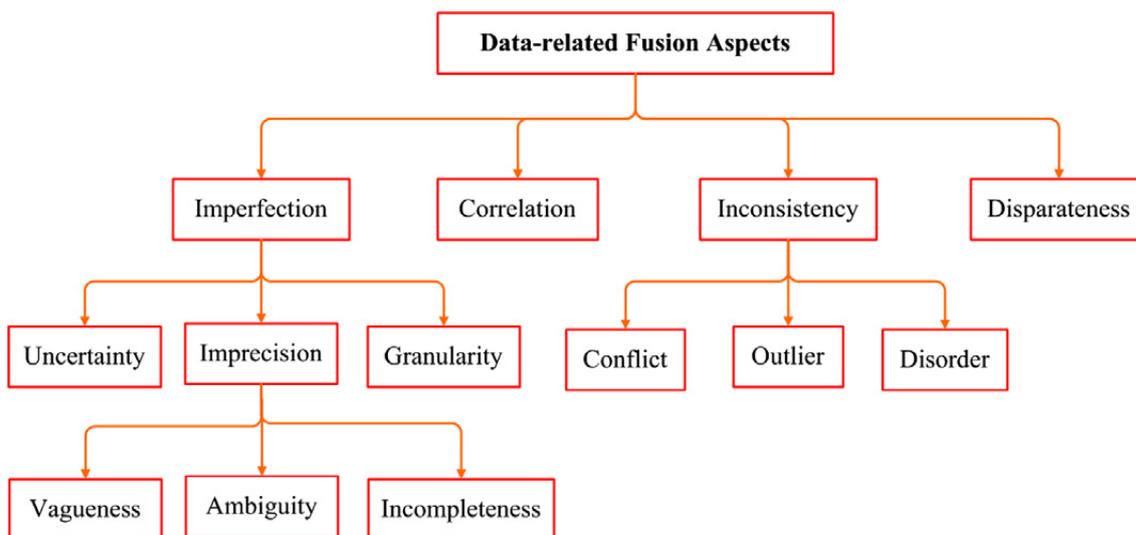


Figura 4: Taxonomia apresentada por [47]

Duas destas categorias principais estão divididas em sub-categorias de problemas mais específicos, são elas:

- **Imperfeição** - dividida em três sub-categorias, a incerteza que é quando o grau associado de confiança sobre o que é indicado pelos dados é menor do que 1, o dado impreciso que refere-se a vários objetos e não apenas um, e a granularidade de dados que refere-se a capacidade de distinguir entre os objetos, os quais são descritos pelos dados, sendo dependentes do conjunto de atributos fornecidos. No que diz respeito a imprecisão, ela ainda é dividida em outras sub-classes: ambiguidade

(referem-se a dados exatos e bem definidos mas imprecisos), incertezas (ou vagos são aqueles com atributos mal definidos, isto é, o atributo é mais de um e não um conjunto ou intervalo bem definido) ou incompletude dos dados (são os que tem falta de alguma informação).

- **Inconsistência** - são dados de entrada os quais são divididos em outras três sub-classes: conflitantes, espúrios ou fora da sequência.

Existem vários mecanismos de fusão de sensores cujo o objetivo principal é a melhora na autenticidade dos dados ou na sua disponibilidade. A discussão sobre esses diferentes mecanismos, de acordo com [47], é dependente de uma taxonomia centrada em dados, e em explorar cada método baseado em dados específicos relacionados a aspectos desafiadores.

Cabe ressaltar que este trabalho não propõe desenvolver uma solução para acabar com os problemas pertinentes ao uso da fusão/integração dos dados ou informações e nem propor algum raciocínio matemático capaz de minimiza-lo, pois para isso já existe a *Bayesian fusion* e a *Dempster-Shafer Evidence Theory*, assim como outras soluções que visam tornar mais confiáveis tanto os dados quanto as informações, como mostra a bibliografia sobre o assunto. Em razão disso, e também para não haver relação com a taxonomia desenvolvida por [47], a qual é mais específica para esses problemas, e a partir das considerações aqui apresentadas a respeito da fusão de sensores, as quais estão baseadas nas pesquisas realizadas para o seu desenvolvimento, foi adotado na taxonomia da ontologia aqui apresentada o termo **Divergência** para fazer referência as situações em que os dados oriundos de diferentes sensores de mesmo tipo, que estejam atuando em fusão, não estão em concordância de valor em suas medições. A palavra divergência na língua portuguesa refere-se a desacordo, discordância (ausência de compatibilidade; divergência de opiniões), entre outras [1].

A divergência, no escopo deste trabalho, é a identificação de uma situação pré-fusão, e que permite a identificação de possíveis problemas já na origem, permitindo assim o levantamento de informações estatísticas a respeito do comportamento dos sensores já que as metodologias de fusão apenas fundem os dados e as informações.

2.2.2 Sistemas de controle industrial

A partir da década de 80 o SDCD - Sistema Digital de Controle Distribuído passou a ser largamente utilizado nas plantas industriais para controle dos processos produtivos a fim de reduzir os esforços humanos e aumentar a estabilidade operacional. Desde então, muitas mudanças ocorreram no ambiente da manufatura, seja nos processos de produção, nas novas tecnologias e equipamentos, como também no comportamento do mercado consumidor. Esses fatores fizeram com que as indústrias adotassem soluções visando proporcionar uma integração mais ampla entre o chão de fábrica e os níveis de gerência

da planta e também proporcionar a troca de informações entre os diferentes níveis a fim de obter melhores condições à tomada de decisões. Assim como todas as outras soluções empregadas na manufatura, o SDCD também possui suas vantagens e desvantagens. Para [34] as vantagens do uso do SDCD são:

- pode ser instalado a partir de uma configuração bem básica e posteriormente melhorado e expandido conforme as necessidades;
- facilidade na execução multitarefa em paralelo por ser composto por muitos computadores;
- também devido a seus vários computadores, tem redundância embutida;
- cabeamento de controle reduzido em comparação a uma configuração de controle por computador central;
- as redes oferecem informações sobre o processo em toda a empresa para um gerenciamento mais eficiente da fábrica e do processo.

Já [76], relacionou as seguintes desvantagens apresentadas pelo SDCD, são elas:

- custo relativamente alto;
- sistema proprietário com problemas de falta de padrão nos protocolos de comunicação, tornando incompatível seu uso com outras aplicações;
- falta padronização de linguagem;
- falta padronização do sistema de interligação;
- sistema grande, pesado e pouco flexível;
- sua aplicação só é justificada em complexos industriais de grande porte e com forte suporte econômico.

Na visão de [93], a conceituação do SDCD foi baseada na distribuição espacial das plantas industriais da época da sua implantação, exigindo uso de redes que possibilitem a conexão entre uma unidade de controle centralizado e os sensores, atuadores e reguladores locais, flexibilizando a manutenção do *hardware* sem flexibilizar a produção.

Além disso, conforme [76], o SDCD não é considerado a melhor solução para os problemas principais do controle de processos, e por isso sua adoção pelos novos e grandes projetos de instrumentação tem diminuído muito em virtude das mudanças mundiais.

Há também que se considerar que com o passar dos anos, o aumento da quantidade de componentes utilizados na automação da produção resultou no aumento da complexidade para controlá-los através de um *software* que não para de crescer de tamanho e proporcionalmente cada vez mais complexo também.

Esse cenário fez com que os fabricantes de componentes eletrônicos passassem a oferecer produtos de alto desempenho, de baixa potência, e com capacidade de comunicação e computacional avançadas, proporcionando que sistemas heterogêneos tenham melhora na integração entre seus dispositivos, com plataforma independente, além de atenderem a outros requisitos de importância, tais como processamento em tempo real, robustez, segurança e soluções estáveis [7]. [93] diz se tratar de componentes mecatrônicos inteligentes, que são compostos por controladores e que possibilitam uma melhora na flexibilização dos sistemas produtivos, podendo ainda serem agregados a sistemas e máquinas com maior facilidade que os tradicionais componentes mecânicos, facilitando o projeto e a reconfiguração automatizada do sistema de produção através do reuso do conhecimento gerado sobre o sistema e seus componentes.

As novas tecnologias de automação apresentam como sua mais importante característica o poder avançado de comunicação aliado aos recursos da informática. Dessa forma, é possível o manejo de grandes quantidades de dados entre dispositivos, máquinas e células, assim como dos produtos utilizados, desenhados, fabricados e comercializados. Esse poder de comunicação entre máquinas, sistemas e sensores, com o consequente compartilhamento de informações, vai trazer no futuro uma maior eficiência na produção, com flexibilidade e rapidez necessárias para atender ao mercado consumidor.

Porém, essa grande quantidade de dados requer uma estrutura que permita a sua manipulação de forma adequada a fim de permitir a extração das informações que irão gerar o conhecimento sobre o domínio e que servirão às futuras tomadas de decisões. Dentro dessa ideia, as unidades de produção tendem a se tornar completamente autônomas.

A partir das considerações aqui feitas a respeito das características apresentadas por um SDCD e sobre as novas perspectivas propostas para o aumento da automação nas plantas industriais no futuro, é possível observar que da forma como foi concebido, o SDCD não terá principalmente capacidade para autonomamente extrair conhecimento do domínio de sua aplicação e a partir dele exercer o controle sobre o sistema produtivo. Além disso, não possui uma plataforma aberta que permita absorver diferentes tecnologias e sua forma de controle é centralizada, condição essa que é análoga à ideia de controle distribuído apresentada em pesquisas baseadas nos conceitos da Indústria 4.0.

Para [7], os grandes sistemas da indústria formam um conjunto complexo, e de grande potencial, de sistemas multidisciplinares, conectados e heterogêneos que funcionam como um sistema distribuído complexo, sendo que os sistemas combinados resultantes são capazes de abordar problemas que os componentes individuais não teriam capacidade de realizar, além de fornecer funcionalidades de controle e automação que só estão presentes porque resultam da criação de fontes de informação novas e "emergentes" e de resultados de composição, agregação de índices de monitoramento existentes e emergentes baseados em características e modelos.

Sendo assim, as pesquisas recentes têm proposto diversas soluções que visam pos-

sibilita a interoperabilidade entre equipamentos, a flexibilização, agilização e rapidez na configuração/reconfiguração das linhas de produção, como também para armazenar e inferir conhecimento sobre os processos produtivos a fim de reutilizá-lo sempre que necessário.

Entre essas novas soluções propostas está a da NSF - *National Science Foundation* a qual sugere o uso de CPS - *Cyber-Physical Systems* que, conforme [27], fornecem técnicas de *design* e análise de dados em escala integrada fazendo a integração do dinamismo dos processos físicos com *softwares* e ferramentas de comunicação. [9] diz que os CPS são sistemas computacionais que atuam e controlam sistemas físicos, podendo interagir com humanos, e expandir as capacidades do mundo físico por meio da computação, comunicação e controle, tornando-se elemento chave para futuros desenvolvimentos tecnológicos. Ainda com relação aos humanos, além de interagir com eles, estes sistemas inteligentes possuem capacidades semelhantes, como por exemplo a tomada de decisão [16]. Ou seja, o CPS é a tecnologia moderna que apresenta-se como sucessora do SDCD no papel de gerenciamento da planta. Mais adiante neste trabalho o CPS será melhor apresentado.

Porém, sozinhos os CPS não são solução completa para atender a complexidade das futuras plantas industriais e absorver o conhecimento sobre as tecnologias nelas embarcadas. Outras soluções e conceitos necessitam ser aplicados e integrados em uma solução conjunta que permita futuramente a concepção de um ambiente produtivo inteligente, autônomo, interoperável, flexível e rápido de ser (re)configurado. Na seção a seguir são apresentados os conceitos e tecnologias que juntamente com um sistema ciberfísico integram e fundamentam a proposta apresentada neste trabalho.

2.2.3 Visão da Indústria 4.0

A popularização da Internet a partir do início deste século e o consequente aumento de seu uso voltado à comunicação entre os usuários fez com que surgissem novas tecnologias que passaram a ser estudadas a fim de avaliar a aplicação de seus conceitos no ambiente das indústrias. Muitos desses estudos dizem respeito a busca de uma solução para permitir que as máquinas dentro de um ambiente de produção comuniquem-se entre si, ou seja, que possam através de uma linguagem comum aos diversos equipamentos e dispositivos de diferentes fornecedores, trocar informações a fim de oferecer uma maior integração entre eles para atender as necessidades de produção, além de possibilitar a inferência de conhecimento para que as linhas de produção tenham agilidade, flexibilidade e autonomia para tomar decisões sem necessidade de interferência humana. Essas são algumas das propostas da nova revolução industrial, que é um projeto estratégico do governo alemão em parceria com as empresas de seu país e que recebeu o nome de **Indústria 4.0**, tendo como objetivo a implementação de fábricas inteligentes através da informatização, caracterizando-se pela capacidade de adaptação, eficiência e integração em todos os níveis

por meio de sistemas físicos/cibernéticos [23].

A entidade que representa as indústrias no Brasil, a CNI - Confederação Nacional da Indústria, afirma que o conceito da Indústria 4.0 tem como característica integrar e controlar a produção por meio de redes interligando sensores, atuadores e outros dispositivos, possibilitando criar sistemas ciberfísicos, os quais usam de inteligência artificial fundindo o mundo real com o virtual [19]. A figura 5 exibe a representação da integração proposta pela Indústria 4.0.

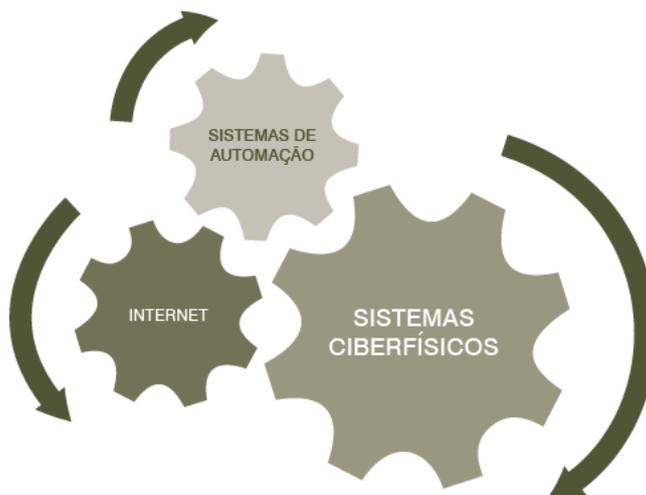


Figura 5: Proposta de integração da Indústria 4.0. Fonte: [19]

[39] afirma que a Indústria 4.0 trata-se de um conceito abrangente que envolve tecnologias e conceitos de organização de uma cadeia de valores que envolve integração de estruturas modulares com sistemas ciberfísicos, criando um mundo físico virtual e com descentralização da tomada de decisão. Uma visão das tecnologias envolvidas no conceito Indústria 4.0 é mostrada na figura 6.

No ambiente industrial tais tecnologias quando aplicadas conjuntamente agregam valor e ampliam as possibilidades, porém isoladamente tornam-se mais limitadas [72]. Dessa forma, faz parte da realidade atual e futura das indústrias a busca pela integração das tecnologias envolvidas em todos os níveis de produção a fim de tornar as linhas de produção autônomas e inteligentes, e também possibilitar que as informações sejam trocadas entre eles, desde o nível mais baixo (chão de fábrica) até o nível gerencial, em qualquer plataforma.

No entanto, para atingir a total integração da planta industrial é necessário que, além do controle sobre todos os processos, sejam realizadas avaliações e melhoras nas configurações de produção através de análises e de simulações a partir dos dados atualizados gerados no chão de fábrica, usando para isso sistemas de sensores atuando em conjunto e fornecendo dados em tempo real e, além disso, haja garantia que as análises realizadas representem o estado atual do sistema de fabricação [69].

Essa ampliação da automação, que tem como objetivo tornar as linhas de produção

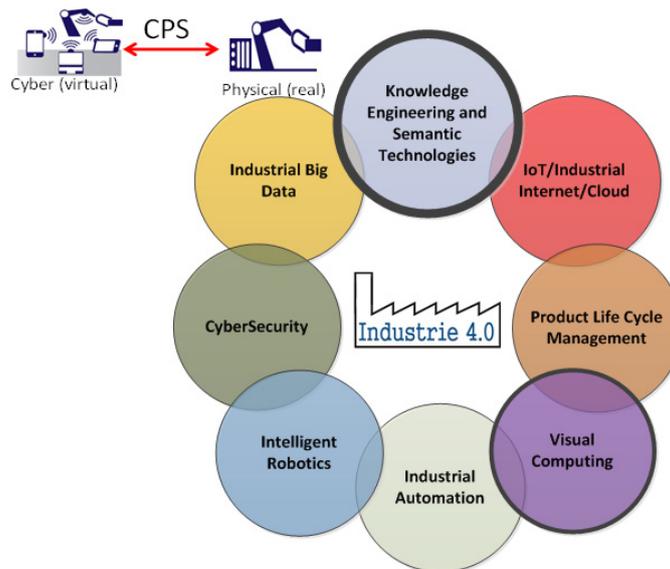


Figura 6: Tecnologias da Indústria 4.0. Fonte: [91]

totalmente autônomas e inteligentes, traz consigo o aumento do número de dispositivos que nelas são utilizados, os quais em suas versões mais atuais embarcam tecnologias que permitem o processamento em tempo real dos dados, robustez e segurança, além da potencialidade de comunicação. Esse é o caso dos sensores, que passam a incorporar inteligência e, conforme [69], estão mais baratos e ganhando uma maior significância, sendo que sua integração com a IA - Inteligência Artificial possibilita visão de máquina, consciência e inteligência.

Pode-se observar pelo que foi exposto a respeito do paradigma da Indústria 4.0, que fatores como flexibilidade, integração, comunicação, inteligência e produção autônoma são alguns dos itens que fazem parte do conceito. No entanto, para que esse conceito seja efetivo nas indústrias é necessário a adoção de novas tecnologias voltadas a informação e a automação industrial. Sendo assim, muitas tecnologias têm sido pesquisadas e desenvolvidas visando atender a essa tendência de tornar as linhas de produção cada vez mais inteligentes a fim de se tornarem autônomas em sua própria gestão.

Com relação a integração dos processos e da diversidade de elementos que compõem ou que irão compor os modernos ambientes industriais, [15] afirma que as ontologias apresentam-se como uma solução para atingir uma integração plena, pois permitem a representação das várias fontes de conhecimento e, além da modularidade, possibilitam sua reusabilidade. Ainda segundo o autor, a utilização de ontologias abrange diferentes áreas de pesquisa já que podem ser aplicadas na análise, modelagem e implementação do conhecimento do domínio, assim como para criar semanticamente vocabulários e metadados para descoberta e expressão de informações.

O paradigma da Indústria 4.0, como já mencionado, propicia o desenvolvimento e a aplicação de tecnologias diversas voltadas ao ambiente fabril, dessa forma fica impossível,

mesmo que de forma sucinta, descrever todas elas nesse trabalho. Sendo assim, serão descritas aqui apenas aquelas que em conjunto com a ontologia proposta servem de base à concretização deste trabalho.

2.2.3.1 CPS - Cyber-Physical Systems

É possível perceber que as transformações já ocorridas nos meios de produção industrial e as novas mudanças que ainda estão por vir, fazem com que as indústrias busquem meios para aplica-las em seus ambientes. Dessa forma há ultimamente um crescente interesse por aspectos como colaboração autônoma, reconfiguração e integração de sistemas heterogêneos, só para citar alguns. Com referência as mudanças que já aconteceram, [40] diz que o refinamento dos padrões das TIC a partir da ampliação de sua capacidade computacional visando o desenvolvimento e integração de sistemas em rede de grande escala propiciou o projeto, implementação e sistemas integrados em rede, ou seja, o CPS propriamente dito.

No domínio da manufatura, os recentes avanços implicaram no crescimento da importância dada às informações geradas pelo ambiente fabril, trazendo como reflexo a ampliação da aplicabilidade e acessibilidade de sensores, sistemas voltados a aquisição de dados, além das redes de computadores. Neste panorama, onde faz-se necessário o gerenciamento e controle dos dados gerados nos processos de produção a partir do chão de fábrica até o nível superior, o uso do CPS tende a se consolidar como solução.

Uma das características dos CPS é que por meio de variadas tecnologias eles permitem integrar sistemas embarcados e suas capacidades, assim como a capacidade de comunicação, sendo que isso possibilita a aplicação de tais sistemas em diferentes segmentos [27] e não somente na manufatura.

Em seu trabalho, [54] coloca que a utilização da comunicação para proporcionar reconfigurabilidade e escalabilidade, além da inteligência agregada em sensores e atuadores, são características que diferenciam o CPS dos sistemas de controle comuns e dos sistemas embutidos, porém é mais complexo, podendo ser mais instável e com fortes restrições no que diz respeito ao desempenho.

Os CPS representam a integração entre elementos de computação, comunicação e processos físicos. [70] diz que tais sistemas tendem a ser híbridos e distribuídos, visto que de maneira geral eles são compostos por uma diversidade de *hardwares*, como sensores, atuadores, unidades de processamento de controle e por dispositivos de comunicação interligados em rede, suportando arquiteturas, protocolos e interfaces diferentes. Nessa visão, os componentes das linhas de produção não só podem ser controlados independentemente, como também podem agir com autonomia e inteligência, trocando informações, tomando decisões e executando ações. Na figura 7 é exibido o esquema de integração com CPS.

Mesmo que a figura 7 seja uma representação básica de um CPS, é possível observar

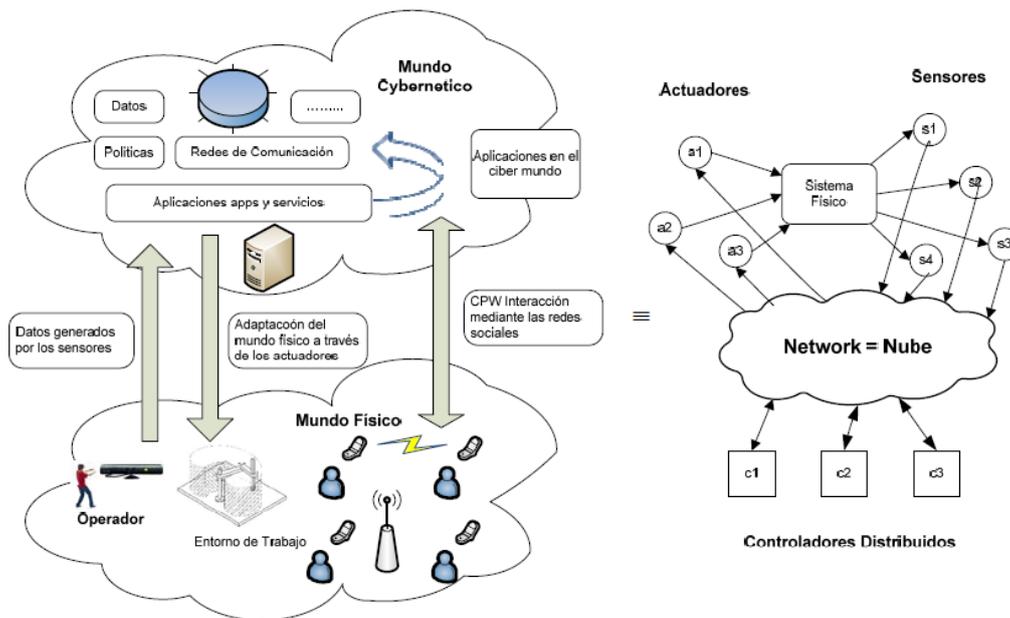


Figura 7: Esquema de integração com CPS. Fonte: [70]

que há uma diversidade de conceitos e estruturas e, pode-se deduzir, que inúmeras são também as fontes de dados em sistemas tão complexos. Mas, como afirma [54], existem alguns desses elementos que são básicos na troca desses dados, dentre eles está a aplicação de sensores em grande número com a finalidade de coletar as informações importantes para o controle do sistema. Dentro dessa visão, [27] afirma que a utilização de sensores para obtenção de dados em determinado domínio integra o conceito e também a implantação de sistemas ciberfísicos.

Para[30], apesar de interessante, o desenvolvimento do conceito de sistema ciberfísico apresenta problemas referentes ao conhecimento do domínio, mais especificamente na estrutura e gerência deste. Ainda aponta o autor que tal conhecimento é essencial para conciliar o variado número de elementos indiferenciados que integram uma arquitetura CPS. E, vai mais além, afirmando que para tratar destas questões ligadas ao conhecimento a solução apropriada é o uso de ontologias, as quais possibilitam que ele seja compartilhado com sistemas automáticos, desde que desenvolvidas em linguagens apropriadas.

Dessa forma, fica evidenciada a razão da utilização dos conceitos relacionados a sistemas ciberfísicos no contexto deste trabalho, mais especificamente naquilo que diz respeito ao uso de fusão de sensores e de ontologias, já que a ideia é apresentar uma solução que possa tanto ser aplicada em ambientes de manufatura com sistemas de controle tradicionais quanto naqueles que já se valem dos conceitos de sistemas ciberfísicos.

2.2.3.2 AutomationML

Conforme [7], no paradigma dos novos sistemas industriais, as diferentes unidades que compõem suas estruturas são vistas como um conglomerado distribuído com carac-

terísticas como autonomia, inteligência, pró-atividade, tolerância a falhas e que ainda possibilitam o reuso dessas unidades. Com tantas características envolvidas e com um grande número de componentes que podem ser desenvolvidos e aplicados nas plantas industriais, faz surgir a necessidade da adoção de padrões que possibilitem a troca de dados entre eles. Fato esse que já é vivenciado há algum tempo na indústria, haja visto que atualmente as linhas de produção já são compostas por grande número de componentes (sensores, atuadores, CLPs, ...), porém ainda não há consenso quanto a padronização da maneira como os dados serão trocados entre os diferentes dispositivos utilizados nas linhas de produção, os quais originam-se de diferentes fornecedores.

Em razão dessa falta de padrão, e a necessidade de encontrar uma solução para o problema da comunicação em um ambiente tão diversificado surgiu a **AutomationML**, a qual segundo [91] [4], a nível de taxonomia, é a ferramenta mais conhecida voltada à construção de vocabulários padronizados para uso no domínio da indústria. De acordo com [11], no que se refere a troca de informações, a AutomationML é uma ferramenta que permite estabelecer um padrão que possibilite que os diferentes sistemas de engenharia produzidos por também diferentes fornecedores, atuem de forma integrada. [5] [91] acrescentam que a AutomationML oferece condições para, através de objetos que encapsulam os diferentes aspectos do domínio, fazer a descrição de seus componentes, sendo que tais objetos podem ser compostos por outros objetos, e todos fazendo parte de uma estrutura mais ampla.

A AutomationML - *Automation Markup Language*, ou simplesmente AML, é um projeto lançado na Alemanha em 2006 e que envolveu governo, indústrias e universidades na sua concepção, objetivando a interconexão entre as diferentes ferramentas da engenharia moderna utilizadas nos mais diferentes domínios através de um formato de dados que permite transferir informações entre elas de forma consistente e que baseia-se na norma IEC 62714 [5]. Isso é possível porque o formato de dados da AutomationML possibilita que as exportações e importações ocorram sem perdas de dados, sendo que ela também permite que sejam feitas simulações e testes com esses dados a fim de propiciar melhorias na sua qualidade bem como reduzir tempo e custos [18]. Essa ferramenta oferece recursos que habilitam a sua utilização na modelagem de sistemas automatizados como os exigidos nas plantas industriais modernas, podendo ainda ser usado para especificar, não só processos como também parâmetros de configuração de CLPs.

Ainda segundo [5], a AML possibilita descrever componentes mecatrônicos de forma única, como também descrever sistemas completos de automação de produção com diferentes níveis de detalhamento, sendo que nesse modelo a informação é dividida em topologia, geometria, cinemática e lógica (sequenciamento, comportamento e controle), além de integrar diversos padrões baseados em XML. Dentro desta características, essa ferramenta viabiliza a construção de um modelo topológico com base em uma estrutura hierárquica, o qual pode ser relacionado com modelos tridimensionais e que posterior-

mente permite a geração de um modelo semântico por meio do seu relacionamento com o conhecimento sobre o domínio, o qual pode ser disponibilizado através de uma ontologia.

As trocas de dados no nível superior da AML são feitas através da utilização do formato CAEX - *Computer Aided Engineering Exchange* que define uma estrutura para armazenamento de informações, propriedades e bibliotecas hierárquicas de objetos, além de possibilitar a representação das informações topológicas das plantas (células, componentes, atributos, interfaces, relações e referências), COLLADA - *COLLABorative Design Activity* para geometria e informações cinemáticas e XML PLCopen para informações de lógica [4] [48] [82] [66]. Para [83], a utilização e a conexão inteligente dos formatos existentes através de *links* e referências é a tarefa principal da AML, sendo que isso faz com que os dados não percam seus formatos e sejam separados quando armazenados e, além disso, simplifica a troca e a adaptação de modelos de plantas industriais complexas.

Baseada no paradigma da orientação a objetos a AML oferece suporte a técnicas fundamentais de relacionamentos, como classe-instância e herança hierárquica [94]. A arquitetura da AML tem como base quatro conceitos CAEX principais, são eles:

- *InterfaceClasses* e *InterfaceClassLib* - definem relações entre objetos AML. A AML prevê algumas interfaces abstratas para sistemas de automação geral, que podem ser estendidas pelo usuário. Uma classe de interface pode ser utilizada para duas finalidades: a) definir relações entre objetos em um arquivo CAEX, isto é, entre objetos de uma topologia de planta (isto inclui todo tipo de relações, por exemplo, de natureza mecânica ou sinais e variáveis relacionadas ao código do PLC); e b) para definir referências de informações que são armazenadas fora do arquivo CAEX (por exemplo, uma descrição 3D para um robô) [48] [82].
- *RoleClasses* e *RoleClassLib* - os conceitos de domínio específico são modelados como *RoleClasses* e organizados como taxonomia na biblioteca *RoleClass*. A AML fornece um conjunto predefinido de bibliotecas *RoleClass* para o domínio de automação e fabricação que abrange conceitos abstratos como produto, processo e recurso, além de componentes específicos como robô ou transportador. Estes podem ser estendidos para cobrir os requisitos individuais do domínio do aplicativo. A classe de função tem a finalidade de descrever objetos físicos ou lógicos de maneira abstrata, dando assim um significado a esses objetos, permitindo dessa forma a interpretação semântica automática por uma ferramenta, como por exemplo uma ontologia.
- *SystemUnits* e *SystemUnitClassLib* - modelos de componentes do sistema de produção podem ser modelados como *SystemUnitClasses*. Essas são classes da AML especificamente definidas pelo usuário e normalmente dependentes do fabricante. Uma *SystemUnitClass* pode referenciar várias *RoleClasses* por meio da *SupportedRoleClass* para demonstrar sua semântica.

- *Instances e InstanceHierarchy* - instâncias concretas de modelos de componentes podem ser armazenadas na *InstanceHierarchy*. Essas instâncias carregam dados do projeto e representam a instalação real da planta no mundo digital. Possibilita a descrição da topologia da planta, incluindo a definição de equipamentos concretos em um projeto real (os dados da instância). A hierarquia da instância contém todos os dados, incluindo propriedades, interfaces, classes de função, relações e referências.

Uma visão geral sobre os conjuntos de dados AutomationML é mostrada na figura 8.

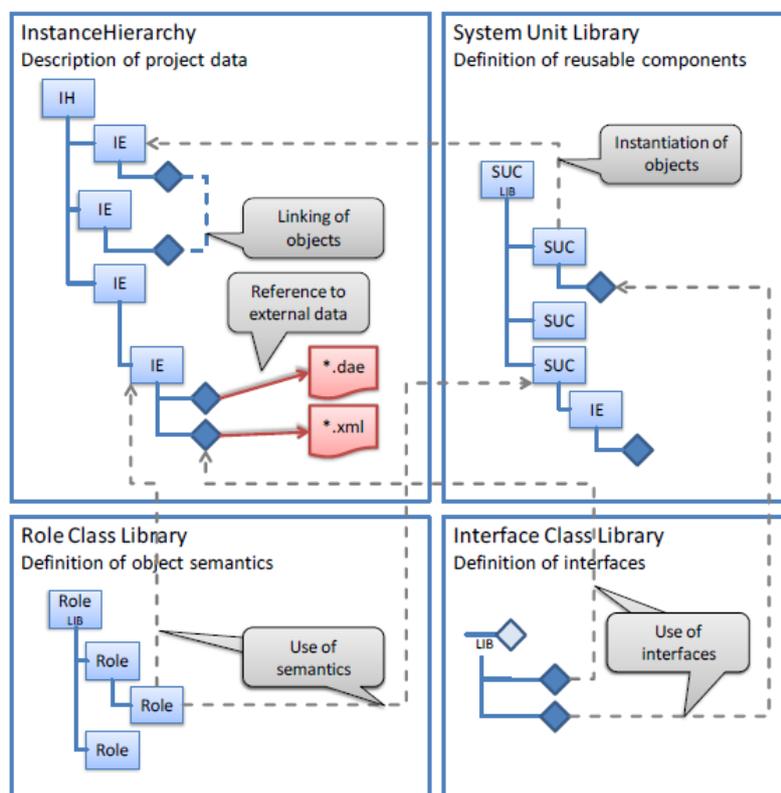


Figura 8: Dados AutomationML. Fonte: [10]

Todavia, conforme [94], a interpretação exata dos dados só é possível por intermédio de um *software* especial que tenha o conhecimento codificado e lógica aplicada, pois nenhum dos elementos AML possui semântica legível por máquina e a troca de dados é meramente uma serialização e desserialização de arquivos XML o que resulta somente em interoperabilidade sintática. [4] acrescenta que, em razão disso, o CAEX não suporta facilmente a validação de modelos de plantas industriais.

Com relação ao problema da semântica na AML, [4] [48] [82] apontam que uma solução seria a utilização de tecnologias da Web Semântica, pois oferecem suporte a integração de dados heterogêneos e distribuídos que possuem *links* entre eles. Além da integração, elas possuem mecanismos que dão sentido aos dados acessados por meio de

um navegador web e também possibilitam a construção de consultas formais habilitadas para o raciocínio sobre os modelos de domínio expressos em formas de ontologias [4] [82]. Estas, segundo [83] são utilizadas na introdução de semântica nos dados modelados, simplificam a troca de conhecimento e estabelecem conceitos que dependem do domínio, sendo possíveis de processar tanto pelos humanos quanto por sistemas informatizados.

Nesta seção foi abordado o problema relacionado a diversidade de componentes nas linhas de produção industrial e a inexistência de um padrão de comunicação que permita que esses componentes interajam uns com os outros a fim de proporcionar agilidade, flexibilidade e autonomia a estas linhas de produção. Foi apresentada a AutomationML que é uma ferramenta que visa facilitar o intercâmbio uniforme de dados entre ferramentas de engenharia e entre os diferentes dispositivos utilizados nos ambientes de produção. Outra razão para justificar a escolha sobre a AutomationML é que, como citam [48] [82], trata-se de uma solução largamente usada na indústria e que atende as especificações da Indústria 4.0.

No âmbito deste trabalho, a AML foi empregada para o desenvolvimento da interface utilizada na formatação dos dados importados do sistema de controle da linha de produção, e que posteriormente são gravados como instâncias no arquivo XML da ontologia proposta para que o motor de inferência possa extrair o conhecimento sobre o domínio da aplicação.

2.3 Representação do conhecimento

Quando foi criada a Web, um dos seus objetivos originais era permitir que os usuários trocassem informações entre si, sendo que os computadores, além de realizarem a comunicação, serviriam também para auxiliar esses usuários na realização de outras tarefas. Na atual realidade da Web, os computadores servem somente para realizar o direcionamento e entrega de informações, sem ter acesso ao conteúdo das páginas acessadas. A razão disso está na diversidade de formas utilizadas na Web para representação das informações, são elas: uso de linguagem natural (por exemplo, Inglês), gráficos, multimídia e *layout* de página. Porém, essa estrutura atende às necessidades humanas, sendo de difícil entendimento pelas máquinas devido a fatores como, por exemplo, ambiguidade dos dados e formatos de dados sem tipos. A representação da estrutura atual da Web é mostrada na figura 9.

Em 2001, Tim Berners-Lee pesquisador do W3C - *World Wide Web Consortium* propôs a Web Semântica com o objetivo de melhorar a exploração do conhecimento contido na Web [90]. Na visão apresentada, os computadores não têm apenas que mostrar os dados armazenados na Web, mas também utiliza-los em automação, integração e reuso entre diferentes aplicações. Mas, para isso ser possível há necessidade que os computadores acessem coleções estruturadas de informações e também a conjuntos de regras de

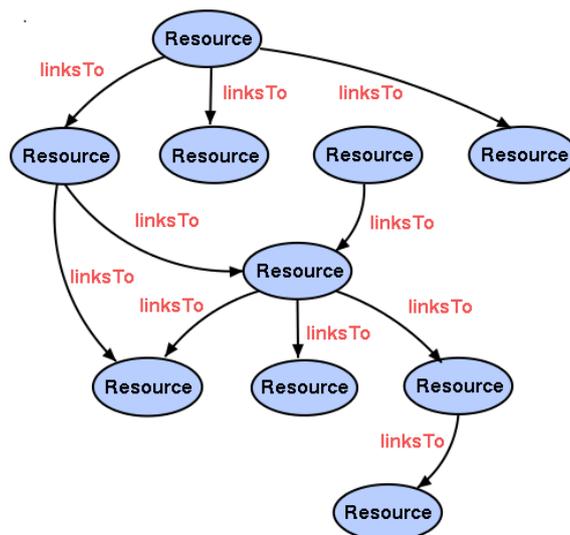


Figura 9: Visão atual da Web. Disponível em: <https://www.w3.org/Talks/2002/10/16-rdf/currentweb.png>

inferência permitindo assim que desenvolvam seu raciocínio de forma automatizada, o qual é a representação do conhecimento.

Na figura 10 é mostrada a visão da Web com Web Semântica.

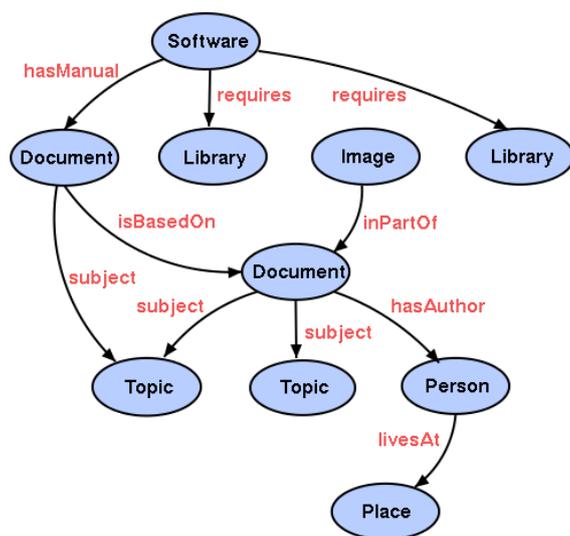


Figura 10: Visão com Web Semântica. Disponível em: <https://www.w3.org/2002/Talks/www2002-w3ct-swintro-em/semanticweb.png>

Ao lançar sua proposta, [90] colocou que a Web Semântica não se tratava de uma outra Web, mas apenas uma extensão desta. [44] refere-se a Web Semântica como sendo uma camada adicional no topo da Web, sendo que a sua construção está baseada em representações explícitas de conceitos de informação e suas relações, como as ontologias e taxonomias. Ainda segundo os autores, a valorização das tecnologias de Web Semântica não ocorre somente para uso na Internet, mas também em sistemas fechados como os encontrados em ambientes industriais.

Conforme [75], no que diz respeito ao domínio da Web Semântica, considera-se que algo expressa semântica quando é interpretado e processado por meio de computadores, dessa forma a Web Semântica, apesar do termo semântica, está se referindo ao tratamento de um subconjunto da linguagem natural, o qual é chamado de semântica formal, só que os modelos com ela desenvolvidos mostram-se incapazes de tratar a linguagem natural usada pelos humanos para se comunicarem. Isso faz com que a Web Semântica esteja restringida a um subconjunto da linguagem natural e formada por sentenças similares a esta, adequadas para máquinas, mas sem a sua capacidade de comunicação. Ou seja, os computadores fazem a leitura do conteúdo, porém sem processá-lo.

Como já citado neste trabalho, [90] aponta como parte da solução para o problema de fazer com que computadores entendam a linguagem natural, permitir que eles acessem coleções estruturadas de informações, que nada mais são do que dados e metadados (dados que descrevem outros dados). Isso porque os metadados agregam semântica, facilitando assim a indexação dos dados a fim de melhorar as tarefas de pesquisa realizadas por motores de busca.

Para [24], a importância do uso de metadados está no oferecimento de suporte à codificação, ao transporte, como também à interoperabilidade semântica, sintática e estrutural para uma diversidade de metadados desenvolvidos de maneira independente. Pois, dentro dessa variedade de tecnologias voltadas à Web Semântica, existem algumas consideradas básicas por serem usadas na representação de dados elementares por meio da padronização da sintaxe. Entre elas encontram-se as ontologias que através da Web Semântica atribuem sentido e significado ao conteúdo dos documentos, ou seja, representam o conhecimento.

2.3.1 Ontologias

As ontologias possibilitam a definição de conceitos (entidades, objetos, eventos, processos,...), dando destaque as suas propriedades, relações e restrições expressas através de axiomas. Elas permitem o compartilhamento do conhecimento sobre um determinado domínio, o qual é representado pelo vocabulário que elas fornecem, dessa forma podem resolver problemas de comunicação entre seus componentes, mesmo que não tenham o mesmo conhecimento sobre esse domínio.

De acordo com [51], as ontologias surgem como uma solução que possibilita a representação das informações geradas durante a produção como uma forma de conhecimento sobre processos de fabricação, sobre os equipamentos e sobre os produtos, através de uma máquina de interpretação. Permite-se, assim, que esse conhecimento seja utilizado para a configuração e reconfiguração do sistema de manufatura como também para possibilitar a tomada automática de decisões com base no conhecimento adquirido pelo sistema de controle. Este compartilhamento de conhecimento, segundo [21], é possível para aqueles que têm necessidades parecidas já que ao criar uma ontologia, também pode-

se compartilhar seus conceitos.

O termo ontologia tem origem na filosofia, porém é utilizado nas áreas de computação e automação com um sentido diferente. No caso deste trabalho, uma definição talvez mais apropriada é dada por [13], que refere-se à ontologia como uma especificação formal, através de uma convenção (no sentido de conformidade) de conceitos.

Porém, a conformidade de conceitos no que se refere a automação da manufatura ainda está longe de ser alcançada visto que, segundo [5], os engenheiros de *software* não conseguem obter informações dos fornecedores de componentes mecatrônicos a respeito da geometria e comportamento de seus produtos, que no caso são úteis para apoiarem e acelerarem o desenvolvimento das aplicações voltadas ao controle da produção. Além disso, não há concordância entre os fornecedores de equipamentos às indústrias de manufatura quanto a maneira que seus produtos irão interagir. Assim, não será possível o reuso de componentes e muito menos do conhecimento sobre os processos e situações que possam ocorrer em uma planta industrial, o que impossibilita a agilidade e flexibilidade necessárias ao aumento da competitividade por parte das indústrias.

Dessa forma, as ontologias apresentam-se como uma solução, já que possibilitam o compartilhamento de um vocabulário, o qual facilita o conhecimento do domínio modelado, permitindo que diversos usuários comuniquem-se entre si, e também possibilitando que diferentes bases de conhecimento se associem [6].

Porém, no contexto da Web Semântica, para representar conhecimento através de uma ontologia é necessário utilizar uma linguagem de ontologia para a Web. Há várias linguagens de ontologias, sendo que cada uma delas oferece diferentes facilidades. Dentre estas linguagens destaca-se a OWL - *Web Ontology Language*, que é a linguagem utilizada na construção da ontologia proposta neste trabalho e que será melhor apresentada mais adiante neste capítulo.

O tamanho do domínio a ser representado na estrutura de uma ontologia é que determinará a quantidade de componentes envolvidos. No entanto, existem alguns componentes básicos que fazem parte de um número expressivo de ontologias, são eles:

- **Classes** - são representações abstratas que refletem alguns aspectos do domínio (ex: objetos físicos, ideais, tarefas, pessoas, etc...). As classes contêm indivíduos e podem ser organizadas em hierarquias de superclasses e subclasses, conhecidas como taxonomia. A OWL-DL tem uma característica que possibilita ao motor de inferência (*reasoner*, em inglês) computar automaticamente os relacionamentos superclasse-subclasse.
- **Indivíduos** - são instâncias das classes.
- **Propriedades** - são relações binárias entre indivíduos, sendo que podem ser inversas de outras propriedades já existentes. As propriedades expressam uma interação entre as entidades do domínio (classes ou indivíduos).

- **Axiomas** são sentenças verdadeiras que permitem capturar as regras do domínio.

No caso das propriedades, a OWL possui dois tipos principais:

- **Object Properties** - as propriedades de objeto conectam um indivíduo a outro indivíduo.
- **Data Type Properties** - as propriedades de tipos de dados conectam um indivíduo a um valor do *XML-Schema Datatype* ou a um literal do RDF - *Resource Description Framework*.

Além das duas propriedades descritas acima, a OWL possui mais uma propriedade chamada *Annotation Property* (propriedade de anotação) e que pode ser utilizada para adicionar metadados a classes, indivíduos, propriedades de objeto e as propriedades de tipos de dados.

A OWL possibilita atribuir características as propriedades para dar um maior significado a elas. A seguir são descritas as características mais comuns de propriedades utilizadas nas ontologias desenvolvidas com OWL.

- **Funcional** - se para um determinado indivíduo "A", pode existir até no máximo um indivíduo "B" relacionado a ele (indivíduo "A"). Ex: a propriedade *temMaeBiologica*.
- **Funcional Inversa** - se a propriedade é Funcional Inversa, significa que a sua propriedade inversa é Funcional. Neste caso, para o indivíduo "B" pode existir no máximo um indivíduo "A" relacionado a ele (indivíduo "B").
- **Transitiva** - se a propriedade relaciona um indivíduo "A" ao indivíduo "B", e também relaciona o indivíduo "B" ao indivíduo "C", é possível inferir que o indivíduo "A" está relacionado ao indivíduo "C" através desta propriedade. Um exemplo é o uso da propriedade *temAntecessor*, se o prefeito "X" da cidade tem um antecessor que era o "Y", e esse por sua vez tem um antecessor que era o "Z". Então neste caso, é possível inferir que o "Z" também é antecessor de "X".
- **Simétrica** - se a propriedade relaciona um indivíduo "A" ao indivíduo "B", então o indivíduo "B" também está relacionado ao indivíduo "A" através da mesma propriedade.

Há ainda outras características que podem ser utilizadas e que aparecem na interface da aba *Object Properties* do Protégé, são elas: Assimétrica, Reflexiva e Irreflexiva.

Toda propriedade possui um *domain* (domínio) e uma *range* (faixa ou escopo), sendo que as propriedades conectam indivíduos de um domínio com indivíduos de um escopo.

As propriedades podem ser usadas para criar restrições aos indivíduos de uma classe. As restrições são divididas em três categorias:

- **Quantifier Restrictions** - as restrições de quantificador são compostas por um quantificador, uma propriedade e uma classe nomeada que contém indivíduos que atendem a restrição. Existem dois tipos de quantificadores:
 - Quantificador existencial (\exists) - a restrição existencial (axioma *someValueFrom*) aplicada a uma propriedade, descreve a classe de indivíduos que tem relação com pelo menos um, ou algum indivíduo da classe definida como *range* da propriedade.
 - Quantificador universal (\forall) - a restrição universal (axioma *allValuesFrom*) aplicada a uma propriedade, descreve a classe de indivíduos que cada vez que participam da relação especificada, o fazem somente como membros da classe definida como *range* da propriedade.
- **Cardinality Restrictions** - com estas restrições é possível determinar a quantidade mínima, máxima ou exata do número de vezes que cada membro da classe participa da relação dada.
- **Restrictions has Value** - o axioma *hasValue* descreve o conjunto de indivíduos que possui pelo menos um relacionamento através dessa propriedade com outro indivíduo específico. Dessa forma, serão considerados membros desta classe todos os indivíduos que tenham ao menos um valor da propriedade igual ao definido pelo axioma *hasValue*.

De acordo como forem utilizadas estas restrições quando da definição de uma classe, a OWL faz uma diferenciação entre:

- **Classe Primitiva** - são as classes que somente possuem como condições restrições necessárias, as quais são especificadas pelo axioma *subClassOf*, também chamado de axioma de inclusão de classe.
- **Classe Definida** - são as classes que possuem ao menos uma restrição como condição suficiente e necessária, também chamada de classe equivalente.

2.3.2 Ontologias para manufatura

Como não existe consenso entre os autores sobre os tipos de ontologias, é adotada neste trabalho a classificação realizada por [37], o qual define quatro categorias: de alto-nível (descrição de conceitos gerais), de domínio (descrevem domínios genéricos), de tarefas (descrevem tarefas e domínios genéricos) e de aplicação (descrição de conceitos dependentes de domínios e tarefas específicas). Com base nessa classificação, as ontologias aqui abordadas são de domínio, mais especificamente ao da manufatura e automação. Porém, ressalta-se que apesar de serem voltadas a esses domínios os objetivos específicos

delas tem diferenças, então foram selecionadas aquelas que de alguma forma têm alguma relação com a proposta apresentada.

Em seu trabalho, [89] faz uma revisão de literatura sobre o uso de ontologias para manufatura com foco no gerenciamento da produção e de suprimentos, onde o autor afirma que o número de ontologias publicadas para a manufatura e logística é limitado. Ainda segundo ele, das 26 ontologias que foram pesquisadas, muitas referem-se a segmentos de áreas específicas e poucas às operações de gerenciamento de endereços, controle de manufatura e logística.

Entre os vários segmentos que direcionam os estudos sobre a aplicação de ontologias na manufatura encontram-se aqueles voltados aos sensores e redes de sensores, como os que são apresentados no trabalho de [84] que é uma revisão de literatura correspondente a estes dois temas, onde são analisadas 9 ontologias, porém com o foco específico para sensores de percepção utilizados nos processos de produção.

A partir das revisões de literatura realizadas por [89] e [84], foi feita uma avaliação para identificar quais das ontologias apresentadas nos dois trabalhos estavam de alguma forma ligadas a fusão de sensores. Isso foi necessário para identificar se alguma destas ontologia já existentes poderia servir para a compor este projeto ou se seria preciso desenvolver uma ontologia nova. Das ontologias apresentadas em ambos os trabalhos foram escolhidas cinco para serem abordadas neste projeto de pesquisa.

Destas, a MASON - *MAnufacturing's Semantics ONtology*, citada em [89] foi escolhida por ser umas das mais abordadas nos trabalhos pesquisados sobre o uso de ontologias na manufatura e direcionada para controle. Já em [84] foram identificadas outras três ontologias que estão relacionadas mais especificamente a utilização de sensores na manufatura.

Além das ontologias já citadas, foram selecionadas mais duas que resultaram de outras pesquisas realizadas. As duas são para controle na manufatura, uma delas é a ADACOR - *ADAPtive holonic COntrol aRchitecture for distributed manufacturing systems* que juntamente com a MASON é amplamente abordada em pesquisas sobre o uso de ontologias na manufatura. E a outra foi proposta por um grupo de pesquisadores do CIFASIS - *Centro Internacional Franco Argentino de Ciencias de la Información y de Sistemas*, a qual foi desenvolvida para supervisão de processos em indústrias de produtos químicos, onde os sensores são muito utilizados, e que mostrou ter características que atendem aos interesses do trabalho aqui proposto.

A seguir são apresentadas algumas características das ontologias citadas, sendo que as três primeiras estão ligadas ao controle na manufatura e as demais aos sensores.

ADACOR

É uma ontologia baseada em MAS - *Multi-Agent System* [51] e HMS - *Holonic Manufacturing System*, usada para controle distribuído objetivando agilidade e flexibilidade

na produção [12]. Ela surgiu propondo quatro tipos de *holons* (partes independentes que se comunicam entre si): *product holon*, *task holon*, *operational holon* e *supervisor holon* [45]. Segue as normas da FIPA - *Foundation for Intelligent Physical Agent* e serve para agendar e despachar dinamicamente ordens de serviço, que são destinadas as estações de processamento apropriadas [51].

A arquitetura da ontologia ADACOR é mostrada na figura 11.

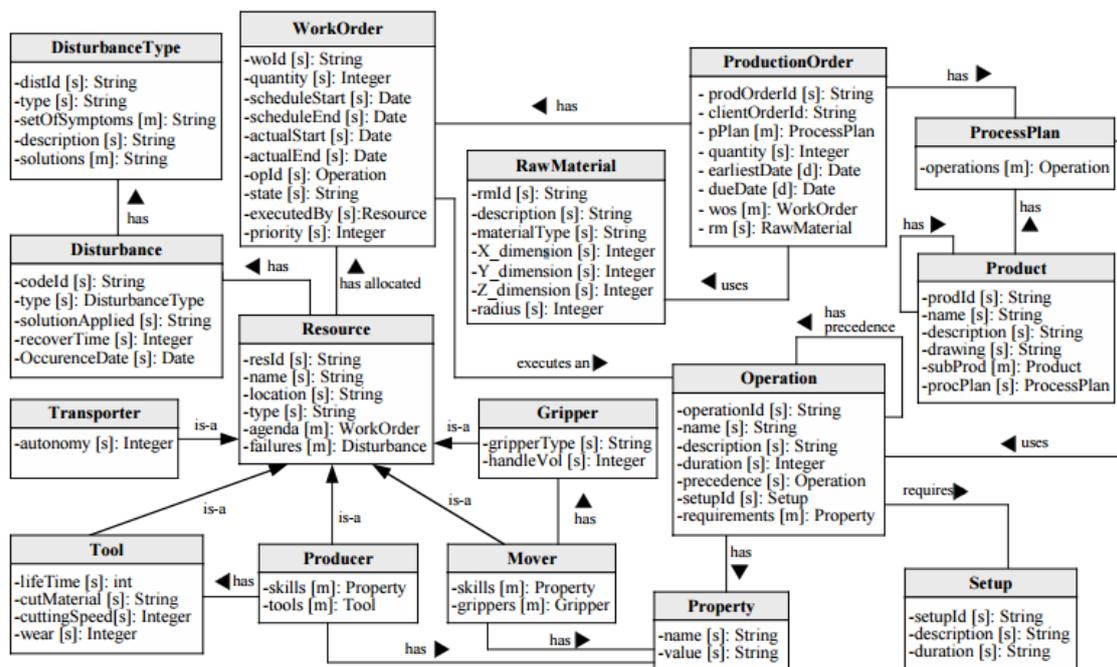


Figura 11: Visão geral da arquitetura da ontologia ADACOR. Fonte: [52]

MASON

É uma proposta construída sobre três principais conceitos: *entities*, *operations*, e *resources* [53]. Ela tem como objetivo oferecer uma semântica comum para compartilhar o entendimento sobre o ambiente de manufatura e foi idealizada como de alto-nível, mas sem necessariamente estar ligada ao domínio da manufatura, porém tem limitações referentes à modelagem de serviços. Na figura 12 é possível observar a estrutura da arquitetura MASON.

Ontologia do CIFASIS

Diferentemente das propostas de outros autores que apresentam soluções de mediação semântica, [64] propõem a exploração das capacidades dedutivas do motor de inferência semântica através da integração dirigida pelo conhecimento (*knowledge-driven approach*). Além disso, o referido trabalho utilizou-se de conceitos da engenharia de processos definidos nos padrões e tecnologias propostas pelo W3C - *World Wide Web Consortium* para construção de Web Semântica. Esta proposta abrange dois enfoques: a incorporação

visto que um equipamento pode ser composto por outros equipamentos também (a recursividade está representada na figura pela linha laranja).

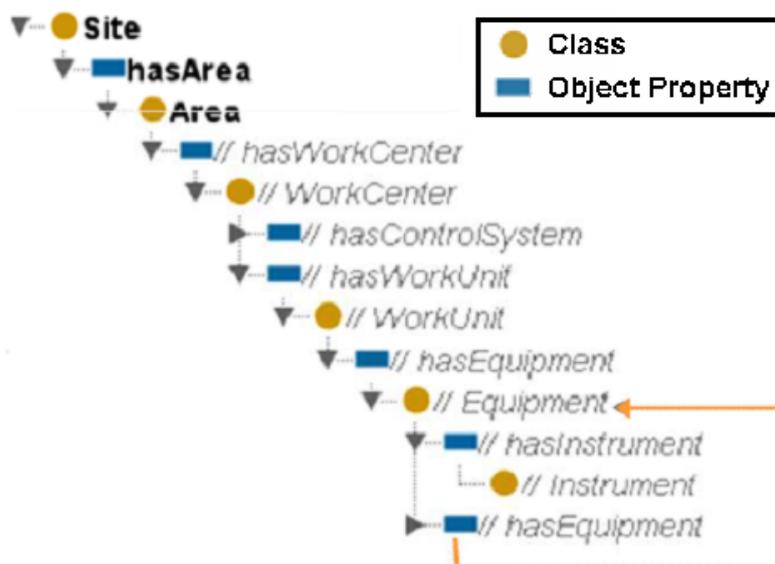


Figura 13: Hierarquia dos equipamentos da planta (a linha laranja mostra a recursividade da propriedade representando uma estrutura de agregação). Fonte: [64]

Na figura 14 é exibida a hierarquia das propriedades de relação *hasPhysicalElement*.



Figura 14: Hierarquia das propriedades dos elementos físicos. Fonte: [64]

Dentre os instrumentos que os equipamentos podem possuir, estão os sensores e atuadores que servem para o monitoramento e controle da planta. Estes dispositivos são utilizados pelo sistema de controle para manter as variáveis de processo em seus valores de referência.

- **Módulo de controle**

Fornece os conceitos necessários à configuração dos sistemas de controle incluindo os componentes para o controle clássico e em cascata. Os conceitos foram organizados através de uma hierarquia de classes enlaçadas aos módulos de equipamentos e de eventos, como mostra a figura 15.

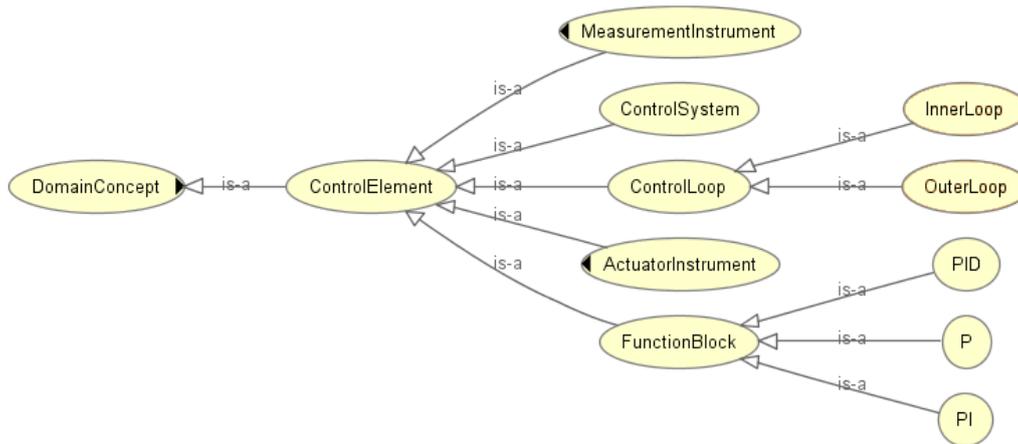


Figura 15: Hierarquia das classes dos elementos de controle. Fonte: [64]

No topo da hierarquia está a classe *ControlElement*, onde estão agrupados todos os conceitos que servem para configuração do sistema de controle da planta. Os sistemas de controle, representados pela classe *ControlSystem* foram definidos como um conjunto de laços de controle (classe *ControlLoop*).

• Módulo de supervisão

Neste módulo é feita a captura dos eventos ou medições que os dispositivos de campo realizam, incluindo as variáveis envolvidas e um registro de seus desvios a fim de detectar possíveis falhas. Cabe ressaltar que a maior quantidade de dados tratada pelo sistema diz respeito aos eventos da planta industrial.

Para representar os eventos, com seus respectivos valores, *time-stamps* e qualidade da medição foi criada a classe *Event*. Também foi criado o axioma *propertychain* o qual especifica a associação entre eventos e os dispositivos através de suas variáveis.

A representação dos desvios é feita através da classe *Deviation*, a qual descreve tanto possíveis falhas quanto a normalidade do processo. Para identificar os desvios é utilizada a variável dinâmica *Parameter* e uma palavra-guia (*GuideWord*), sendo que os outros atributos indicam os limites superior e inferior admitidos por cada desvio.

Ontologia de Pedigree Formal para Fusão de Sensor de Nível 1

Apesar de ser desenvolvida para fins militares, mais especificamente para operações navais, a ontologia de [63] pelas suas características pode ser também aplicada na manufatura, pois trata da fusão a nível de sinal a qual pode ser aplicada a qualquer área onde os sensores são empregados. Conforme [84], essa ontologia é destinada a manter a proveniência de dados de múltiplos sensores a fim de permitir que o usuário possa classificar e interpretar os dados dos sensores, isto é, os usuários decidem em quais sensores confiar.

Para [95], esta ontologia trata em alto nível conceitos relacionados a sensores, sistemas, humanos, configurações, *software*, *InfoSource*, dados de relatórios e outros. Os

autores ainda realçam a importância a ser dada ao pedigree dos dados usados na fusão de nível um, ou seja, a sua procedência, e afirmam que ela faz a descrição da maneira como são coletados e para o que eles contribuem.

Na figura 16, é mostrada a arquitetura da Formal Pedigree Ontology for Level-One Sensor Fusion.

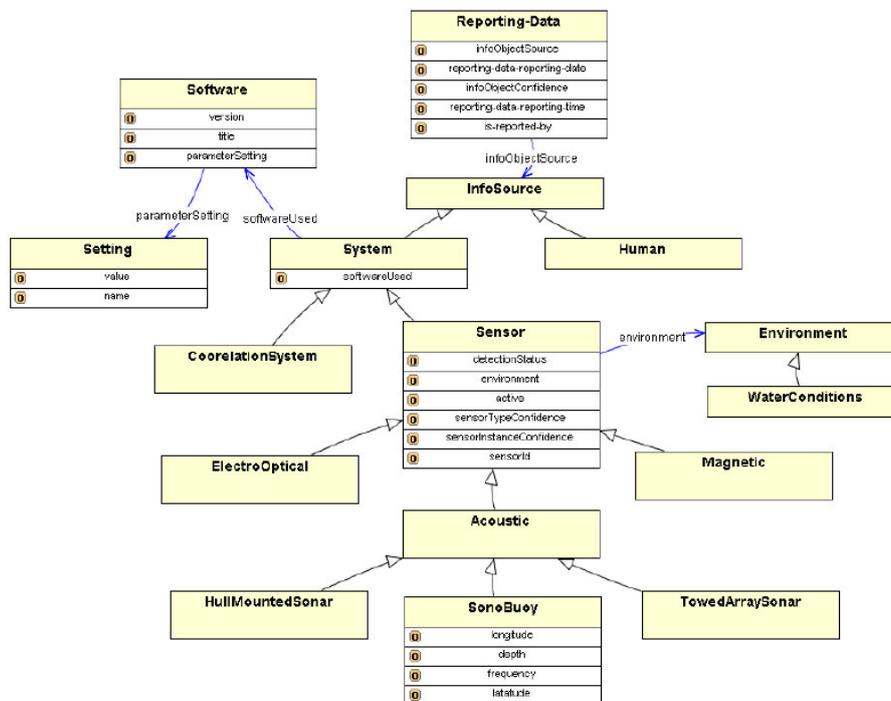


Figura 16: Arquitetura da Formal Pedigree Ontology for Level-One Sensor Fusion. Fonte: [63]

Semantic Sensor Network ontology

A SSN - *Semantic Sensor Network ontology* é uma ontologia OWL 2 desenvolvida pelo grupo *W3C Semantic Sensor Network Incubator (SSN-XG)* [17], a qual faz a descrição de sensores e das observações a eles associadas, porém sua organização é somente conceitual, não existe fisicamente. Constituída por dez módulos, contendo 41 conceitos e 39 propriedades de objeto, ela herda 11 conceitos e 14 propriedades de objetos DOLCE - *Descriptive Ontology for Linguistic and Cognitive Engineering*.

Conforme [84], a ontologia SSN possibilita a descrição de sensores, com sua precisão e capacidades, assim como das observações e métodos utilizados para a detecção, sendo que foi construída levando em consideração o ODP - *Ontology Design Pattern* com descrição das relações entre sensores, estímulos e observações. Na figura 17 é mostrada a estrutura da ontologia SSN.

Nesta seção foram apresentadas as ontologias pesquisadas cujo o foco estivesse voltado à produção industrial ou que apresentavam características que pudessem ser aproveitadas neste domínio. A pesquisa teve a finalidade de avaliar ontologias existentes com

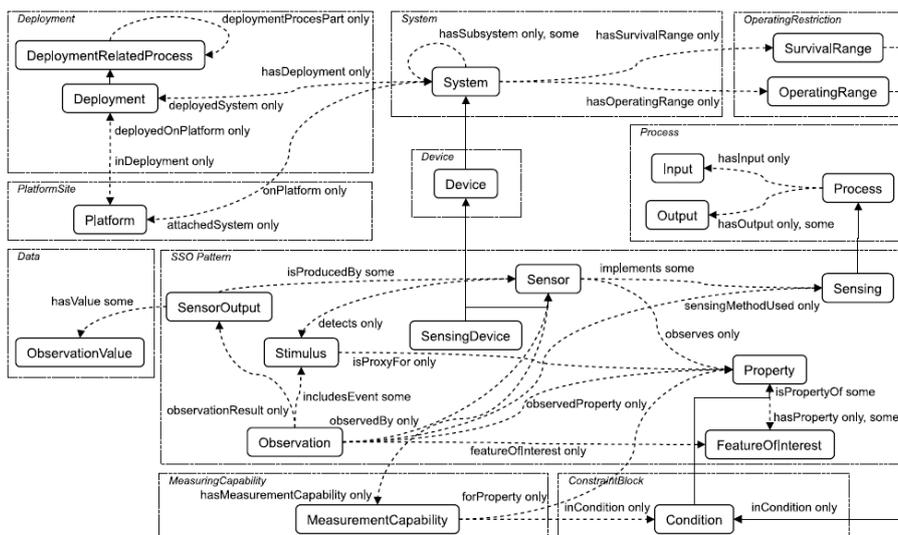


Figura 17: Estrutura da ontologia SSN. Fonte: [17]

o objetivo de analisar seu potencial para atender total ou parcialmente à proposta deste trabalho ou, caso contrário, se haveria necessidade de desenvolver uma nova ontologia a partir do zero.

Dentre as ontologias referenciadas neste trabalho, a ADACOR, mostrou-se inapropriada para o trabalho proposto, por ser fortemente acoplada a uma arquitetura de controle holônica, fato esse que restringe o seu uso, além de não estar disponível em outro formalismo semântico mais geral. Outro fator que também atua em desfavor a essa ontologia é que, segundo [45] tem a limitação de abordar controle adaptativo, considerando a evolução através de duas configurações pré-definidas que são ao mesmo tempo um dos seus principais pontos fortes e sua principal fraqueza. Por um lado, este mecanismo de equilíbrio permite ao sistema combinar prontamente a reação a perturbações e a otimização quando é adquirida a função normal. No entanto, ao mesmo tempo, o sistema é aprisionado com estes dois estados pré-definidos, não sendo capaz de dinamicamente descobrir novos pontos de melhor funcionamento, o que implica na utilização de mecanismos de auto-organização.

Com relação a ontologia MASON, mesmo sendo uma ontologia aberta, ou seja, passível de novas implementações, apresenta limitações à modelagem de serviços, fator esse que acaba por tornar inadequada sua utilização neste projeto visto que os ambientes modernos de manufatura são baseados em serviços. Além disso, ela também não possui classes em sua estrutura que possam ser aproveitadas e também não oferece nenhum tratamento para fusão de sensores.

A ontologia do grupo de pesquisadores do CIFASIS, dentre as cinco ontologias analisadas, foi considerada a que mais atendeu às necessidades do projeto, pois além de ser aberta, permitindo assim agregar as implementações exigidas pela proposta aqui apresentada, mais especificamente ao tratamento da fusão sensorial, mostrou estar mais adaptada

às necessidades atuais dos sistemas de produção, visto que seus conceitos contemplam plenamente a descrição dos agentes básicos presentes nas plantas industriais (equipamentos, processos e supervisão). Outro fator que favorece o uso desta ontologia é que sua estrutura contempla a descrição de laços de controle simples e em cascata, além de oferecer um modelo que permite a representação das variáveis dos processos como também os eventos capturados por sensores, assim como para possíveis desvios que normalmente ocorrem durante os processos produtivos. Apesar de ser bastante abrangente a ontologia do CIFASIS não oferece condições ao tratamento específico das questões envolvendo fusão de sensores, como identificação de ocorrência de fusão em elementos físicos ou tratamento para os dados originados da fusão, porém ela possibilita todo o tipo de monitoramento quanto ao comportamento individual dos sensores em uma planta industrial, descrevendo suas falhas, os riscos e consequências a elas associados, características essas que podem ser aproveitadas em um trabalho futuro que possibilite a integração dessa ontologia com a ontologia aqui apresentada.

Já as duas ontologias diretamente relacionadas aos sensores não contemplam os requisitos definidos neste trabalho como necessários para uma ontologia direcionada ao gerenciamento de fusão de sensores em uma planta industrial. A Ontologia de Pedigree Formal para Fusão de Sensor de Nível 1, mesmo tratando da fusão a nível de sinal, a qual pode ser aplicada a qualquer área onde os sensores são empregados, sua estrutura é totalmente voltada aos sensores em si, não considerando os demais elementos físicos que constituem as plantas industriais (área, linhas de produção, equipamentos,...) e também não possui tratamento para divergências de dados entre sensores em fusão. O mesmo acontece com a SSN, a qual é citada em diversos trabalhos relacionados aos sensores, porém como está no seu próprio nome, é direcionada ao gerenciamento de redes de sensores sem oferecer nenhum tratamento sobre divergência de dados entre eles quando utilizados em fusão.

Dessa forma, foi feita a opção pelo desenvolvimento de uma nova ontologia especificamente para atender aos requisitos definidos neste trabalho, como possibilitar a identificação de todos os elementos de uma planta industrial direta ou indiretamente envolvidos em fusão de sensores, bem como identificação e tratamento de divergência nos dados gerados por esses sensores. De acordo com a proposta do trabalho, a ontologia deverá integrar os conceitos e tecnologias que a seguir serão abordados, a fim de possibilitar a construção de uma solução que sirva de apoio ao controle de plantas industriais, disponibilizando para isso o conhecimento sobre o domínio da fusão de sensores.

2.3.3 Metodologias para construção de ontologias

O primeiro passo necessário à construção de uma ontologia é definir qual a metodologia mais adequada ao universo que se quer modelar. Pois, segundo [43], sistematicamente a metodologia define as atividades de gerência, construção e suporte necessárias à ontologia.

As primeiras metodologias para construção de ontologias surgiram na área da Inteligência Artificial, particularmente no eixo ligado a representação do conhecimento. Depois a Engenharia de Software inspirou a construção de ontologias baseadas no desenvolvimento de *software* e, com o passar do tempo, surgiram metodologias baseadas em IHC - Interação Humano-Computador [43].

Como existem diversas metodologias, e não há como definir qual delas é a mais correta, existem alguns aspectos que auxiliam na sua escolha, como saber a que se propõe a ontologia e qual seu grau de descrição, detalhado ou geral [67], ou seja, definir o domínio e o escopo. Ainda segundo o mesmo autor, provavelmente será necessário rever a ontologia durante seu ciclo de vida através de um processo iterativo de construção.

Grüninger e Fox

Desenvolvida por [33] sua construção teve como base a experiência adquirida no Laboratório de Integração de Empresas da Universidade de Toronto, mais especificamente na participação no projeto TOVE - *Toronto Virtual Enterprise*. Fazem parte de sua abordagem:

- **Cenários de motivação:** objetivam identificar problemas no ambiente atual, os quais servem de motivação para construir ontologias. Trata-se de uma fase de pré-desenvolvimento da ontologia;
- **Questões de competência informal:** formuladas em linguagem natural, têm como objetivo a especificação dos requisitos que a ontologia tem a atender;
- **Especificação da terminologia:** concepção da terminologia formal, onde são especificados (geralmente em lógica de primeira ordem) os objetos, atributos e as relações da ontologia;
- **Questões de competência formal:** a terminologia definida é usada para formalizar os requisitos da ontologia;
- **Especificação de Axioma:** especificação de axiomas formais, que restringem a interpretação dos termos envolvidos nas questões de competência formal;
- **Teorema da completude:** é a definição das condições em que as questões de competência são realizadas.

Uschold e King

[92] propuseram esta metodologia abrangente que teve como base o conhecimento adquirido a partir de suas participações no desenvolvimento da *Enterprise Ontology*, a qual oferece uma coleção de termos e definições com relevância a negócios empresariais, sendo construída pelo *Enterprise Project del Artificial Intelligence Applications Institute*

da Universidade de Edimburgo em parceria com a IBM [56]. Ela é dividida em quatro fases:

- **Identificação do propósito:** consiste em identificar as razões para o desenvolvimento da ontologia e o nível de formalismo para descrevê-la. Também são identificadas as classes de usuários da ontologia;
- **Construção da ontologia:** esta fase está dividida em outras três:
 - Captura - captura ou concepção da conceitualização da ontologia
 - Codificação - codificação ou implementação da ontologia através de uma linguagem que represente ontologias
 - Integração - integração com outras ontologias existentes
- **Avaliação da ontologia:** é realizada através dos requisitos especificados;
- **Documentação:** definida segundo o propósito e o tipo da ontologia. Trata de suas pretensões e das primitivas utilizadas para expressar suas definições.

IDEF5

Segundo [32], IDEF - *ICAM (Integrated Computer Aided Manufacturing) DEFinition* - é uma família de métodos, desenvolvidos pela KBSI - *Knowledge Based Systems, Incorporation*, sendo que o primeiro surgiu para atender a uma solicitação da força aérea norte-americana. IDEF5 tem o objetivo de apoiar a criação, como também a modificação e manutenção de ontologias. As diretrizes do IDEF5 são as seguintes:

- **Organização e escopo:** estabelecimento do propósito, do ponto de vista e do contexto do projeto de construção da ontologia. O propósito fornece um conjunto de "critérios de conclusão" para a ontologia, incluindo os objetivos e exigências. O escopo define os limites da ontologia e especifica as partes que devem ser incluídas ou excluídas do sistema;
- **Coleta de dados:** aquisição dos dados necessários para o desenvolvimento da ontologia. Esses dados são obtidos através de entrevistas com especialistas no domínio, análise de documentos e através da observação introspectiva de certos fenômenos ou atividades de uma organização;
- **Análise de dados:** a partir dos resultados da coleta de dados é possível construir uma caracterização inicial da ontologia. Primeiramente, são listados os objetos de interesse no domínio, seguido da identificação dos objetos sobre os limites da ontologia. E finalmente, podem ser identificados os sistemas internos dentro dos limites da descrição;

- **Desenvolvimento de uma ontologia inicial:** é um protótipo da ontologia no qual são feitas as avaliações preliminares. Ele contém proto-conceitos ou seja, descrições iniciais de tipos, relações e propriedades;
- **Refinamento e validação da ontologia:** os proto-conceitos são refinados e testados. O refinamento é um processo de validação dedutiva, onde as estruturas da ontologia são "instanciadas" com dados reais, e o resultado da instanciação é comparado com a estrutura da ontologia.

METHONTOLOGY

A metodologia Methontology [61] é uma das mais conhecidas metodologias para construção de ontologias e foi desenvolvida por pesquisadores do Laboratório de Inteligência Artificial da Universidade Politécnica de Madri. Ela permite a construção de ontologias completamente novas como também a reutilização de ontologias existentes.

Nesta metodologia a evolução dos protótipos é a base do ciclo de vida da ontologia, além disso, ela também oferece técnicas que auxiliam na realização das atividades relativas ao planejamento, desenvolvimento e suporte. Outra característica desta metodologia é o conjunto de estágios de desenvolvimento que ela propõe. São eles:

- **Especificação:** consiste na elaboração de um documento, em linguagem natural, contendo o objetivo da ontologia e seus propósitos;
- **Aquisição de conhecimento:** estágio que se repete nos outros estágios. Trata da busca por fontes de conhecimentos, que podem ser entrevistas com especialistas do domínio, consulta a livros e documentação, ontologias existentes, etc;
- **Conceitualização:** organização e representação do conhecimento útil sobre um domínio em um modelo conceitual. É o principal estágio da metodologia e está baseado no vocabulário adquirido nos estágios anteriores, tendo como objetivo descrever os problemas e suas soluções possíveis;
- **Formalização:** transformação do modelo conceitual, definido no estágio anterior, em um modelo formal representado em linguagem formal;
- **Integração:** trata da integração da ontologia que se está desenvolvendo com ontologias já existentes;
- **Implementação:** trata da codificação da ontologia em linguagem formal como: Prolog, Ontolingua, OWL - *Web Ontology Language*, etc;
- **Avaliação:** é a avaliação da própria ontologia considerando os processos de verificação e validação;

- **Documentação:** realizada em todo o processo de construção da ontologia, sendo importante no suporte a possíveis manutenções e facilita a reutilização da ontologia.
- **Manutenção:** trata das alterações que objetivam melhorias ou correções.

O desenvolvimento de uma ontologia é um trabalho bastante complexo, porém ele pode ser facilitado se parâmetros forem adotados durante sua evolução. O primeiro deles é a definição de uma metodologia que contemple as exigências estabelecidas para a construção da ontologia, as quais são determinadas conforme a abrangência do domínio modelado. Dessa forma, com base nas pesquisas realizadas para definir uma metodologia que permitisse o desenvolvimento da ontologia proposta neste trabalho, optou-se pela Methontology, que apesar de não ser específica da área de automação e manufatura ela contempla a definição de seus requisitos básicos. Também possui um conjunto de atividades que devem ser aplicadas durante o desenvolvimento, especificando o ciclo de vida da ontologia ao longo deste e, além disso, sugerindo técnicas a serem realizadas durante as fases do ciclo de vida do projeto. Outros fatores que determinaram a escolha da Methontology é que ela além de atender a todos os itens definidos nas pesquisas de avaliação, oferece uma proposta para o gerenciamento do projeto, algo que é fundamental na construção de qualquer ontologia.

A tabela 1, adaptada de [20] apresenta um comparativo entre as metodologias descritas neste trabalho.

2.

2.3.4 Tecnologias semânticas

Como já dito neste trabalho, as novas tecnologias empregadas na manufatura trazem consigo, entre outras coisas, um aumento considerável no volume de dados disponibilizados aos sistemas de controle das plantas industriais. Nesse sentido muitas pesquisas são desenvolvidas visando a busca de soluções que eliminem ou diminuam o problema relacionado ao tratamento de quantidades grandes de dados e de como, a partir deles, extrair informações confiáveis e seguras para ajudar na tomada de decisões, sendo que parte dessas pesquisas são direcionadas ao desenvolvimento de métodos de representação e organização de informações que objetivam a extração do conhecimento. Entre as pesquisas desenvolvidas nessa área, destaca-se o papel do W3C, não apenas pelos trabalhos de pesquisa desenvolvidos, assim como pelo estabelecimento de padrões, como é o caso da Web Semântica.

O surgimento da Web Semântica e a intenção de através dela prover inteligência às máquinas, fez com que aumentasse o interesse por soluções cuja estrutura envolva uma ou mais tecnologias com capacidade semântica. A esse respeito [79] afirma que as tecnologias semânticas possibilitam a utilização dos dados com maior eficiência através da aplicação de ontologias que descrevem o conteúdo e as relações entre os dados a fim

Tabela 1: Quadro comparativo entre metodologias avaliadas.

CICLO DE VIDA	METODOLOGIAS			
	Gruninger e Fox	Uschold e King	IDEF5	Methontology
Especificação de Requisitos	Questões de competência informal	Determinar o propósito da ontologia	Determinar o propósito da ontologia e seus limites	Determinar o propósito, grau de formalização e escopo da ontologia
Modelagem Conceitual	Concepção da terminologia formal	Construção do vocabulário consensual	Coleta e análise de dados de entrevistas, documentos, etc	Métodos para construção da conceitualização da ontologia
Formalização	Transforma o modelo conceitual em um modelo formal para definir de forma precisa seu significado	Ausente	Desenvolvimento de uma ontologia inicial	Não obrigatória, mas pode ser realizada em linguagem formal
Implementação	Linguagem Prolog	Linguagem Prolog e Ontolingua	IDEF5 Schematic Language (gráfica) e IDEF5 Elaboration Language (textual)	Critérios para escolha de ferramentas, podendo ser: Ontolingua, Prolog, ...
Integração	Integração a ontologias de núcleo comum	Integração com ontologias existentes	Somente com outros métodos IDEF	Documento de integração com meta ontologias
Avaliação	Através de teoremas completos	Questões de competência	Refinamento e validação da ontologia	Verificação e validação da ontologia
Documentação	Ausente	De acordo com o propósito e o tipo da ontologia	Ausente	Recomendada em cada fase do ciclo de vida da ontologia

de que possam ser compreendidos pelas máquinas. Dessa forma, faz-se necessária aqui a apresentação de diferentes tecnologias semânticas que foram objeto de estudo para o desenvolvimento da ontologia proposta neste trabalho.

A seguir são descritas as linguagens padronizadas que são utilizadas no desenvolvimento de pesquisas envolvendo tecnologias semânticas. Estas linguagens permitem descrever conteúdos por meio de metadados com a finalidade de facilitar o entendimento e a troca de informações entre humanos e máquinas. Na figura 18, é mostrada a arquitetura da Web Semântica onde é possível observar a camada a qual pertence cada uma destas linguagens.

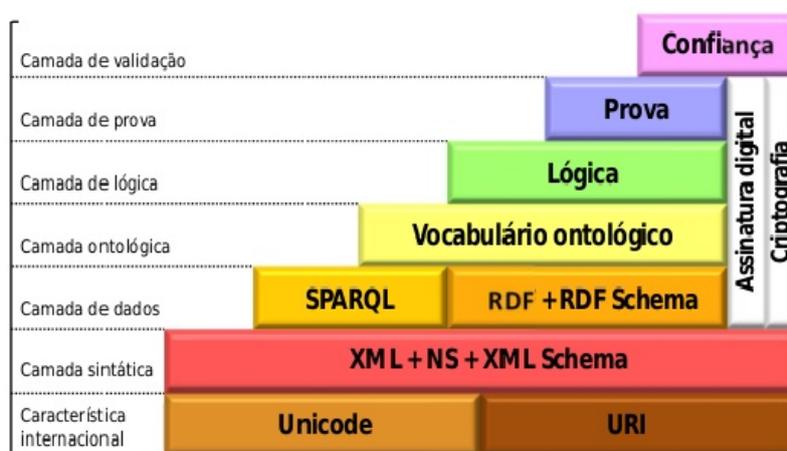


Figura 18: Declaração RDF simples. Fonte: Elaborado pelo autor

2.3.4.1 XML (eXtensible Markup Language)

Desenvolvida com o objetivo de atender as necessidades dos projetos de páginas web com marcação para uso exclusivamente industrial, com formato neutro para troca de dados em um ambiente heterogêneo sem depender de fornecedores, editoração independente do meio, marketing direto, gerenciamento de *workflow* em ambientes autorais colaborativos, e o processamento de documentos web por navegadores inteligentes [21].

XML oferece uma sintaxe de estruturação de documentos sem imposição de regras semânticas, mas não permite a representação do significado da estrutura. Sua funcionalidade está em nível semântico. Tem amplo uso na representação de ontologias, apesar de não ter a expressividade da OWL.

2.3.4.2 XML Schema

Baseada em XML, esta linguagem permite restringir a estrutura de documentos XML através da definição de regras de validação, ou seja, ela define a estrutura de documentos XML. As finalidades destas regras são:

- restringir estrutura e conteúdo de documentos
- habilitar herança de elemento, atributo e definições de tipos de dados
- restrições e descrições específicas da aplicação
- habilitar integração de esquemas estruturais com tipos primitivos de dados
- tipificação de dados primitivos (*byte, data, integer,...*)
- criar tipos de dados definidos pelo usuário

2.3.4.3 RDF (Resource Description Framework)

Modelo padrão para intercâmbio de dados na Web, facilitando a fusão de dados, mesmo de esquemas diferentes, e amplia a estrutura de ligação da Web através do uso de URI -*Universal Resource Identifier* para nomear a relação entre coisas, bem como as duas extremidades do *link*, a chamada "tripla", o que permite a mistura de dados estruturados e semi-estruturados, assim como sua exposição e compartilhamento entre diferentes aplicativos [36]. O RDF foi projetado para situações onde as informações necessitam ser processadas por aplicações, e não por humanos.

A modelagem RDF pode ser feita de duas formas, uma delas é na forma de grafos onde os nós e arestas são identificados por URIs, sendo que um nó representa o Recurso, a aresta é a Propriedade do recurso que liga este ao Valor. A outra forma de modelagem é por meio de uma tripla, onde Sujeito, Predicado e Objeto possuem respectivamente as mesmas definições que Recurso, Propriedade e Valor. Na figura 19 é exibida uma declaração simples em RDF.

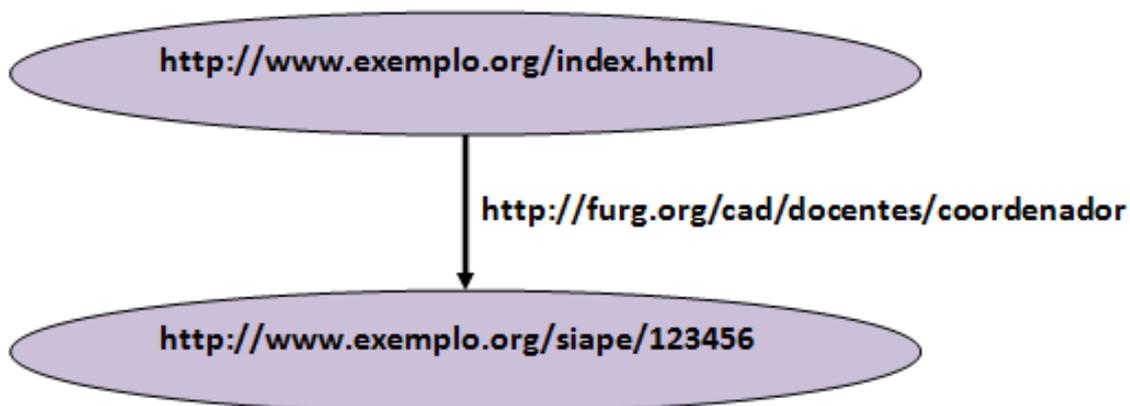


Figura 19: Declaração RDF simples. Fonte: Elaborado pelo autor

Um modelo baseado em RDF pode ser representado usando sintaxe XML, isso visa facilitar o processamento executado pelas máquinas. Além disso, é bastante expressiva, pois permite a representação de conceitos, taxonomias de conceitos e relações binárias.

O RDF também define uma linguagem de marcação XML específica chamada de RDF/XML. É nesta linguagem que é criado o arquivo com o código da ontologia.

2.3.4.4 *RDF Schema*

Essa linguagem oferece um vocabulário para descrever propriedades e classes de recursos RDF e, juntamente com semânticas, criar hierarquias para essas propriedades e classes.

2.3.4.5 *OWL (Web Ontology Language)*

Desenvolvida como uma extensão de vocabulário do RDF a *Web Ontology Language*, ou simplesmente OWL, é uma linguagem de marcação semântica para publicação e compartilhamento de ontologias na Web e tem como origem as linguagens (DAML + OIL) DAML - *DARPA Agent Markup Language* e OIL - *Ontology Inference Layer* [85].

A OWL possibilita criar vocabulários para descrever: classes e propriedades, relacionamentos entre classes, cardinalidade, igualdade, variados tipos de propriedades, características de propriedades e classes numeradas. Assim como o RDF, sua funcionalidade está em nível semântico.

Uma das características da OWL é proporcionar um conjunto mais amplo de operadores para serem usados na especificação de entidades na ontologia. Outra característica é oferecer uma formalização semântica que permite o raciocínio sobre os dados. A razão disso, é que grande parte da definição da OWL tem base na DL - *Description Logic*, ou Lógica Descritiva, só que essa não suporta algumas construções em OWL [79]. Dessa forma, surgiram outras variantes da OWL, tais como:

- **OWL-Lite** - sintaticamente mais simples, essa sub-linguagem deve ser utilizada em situações onde há necessidade apenas de restrições simples e uma hierarquia de classes simples também. Suporta cardinalidade, mas só permite os valores 0 e 1. É mais fácil de ser implementada com outra ferramenta e é mais rápida na transição de outros modelos de vocabulários e taxonomias para OWL.
- **OWL-DL** - sub-linguagem baseada em lógica descritiva e passível de raciocínio automático, o que permite verificar inconsistências na ontologia. Além disso, sua expressividade é maior que a OWL-Lite, sendo que dá garantia ao sistemas de que todas as conclusões serão executadas (completude) e que todos os cálculos terminarão em tempo finito (decidibilidade). A OWL DL possui todas as potencialidades da OWL, porém com algumas restrições quanto à sua utilização.
- **OWL-Full** - A mais expressiva das sub-linguagens OWL, que pode ser usada quando a alta expressividade é mais significativa do que a garantia da decidibili-

dade ou completude da linguagem. Uma das características dessa sub-linguagem é que não é possível efetuar inferências em ontologias nela baseadas. Sua finalidade é ser uma extensão do vocabulário do RDF *Scheme*, mas sem capacidade de raciocínio.

Considera-se a OWL-Full como uma extensão do RDF, enquanto a OWL-Lite e OWL-DL são extensões de uma visão delimitada do RDF. Dessa forma, todo documento OWL é um documento RDF e o contrário é verdadeiro também, só que apenas determinados documentos RDF serão válidos em OWL-Lite ou OWL-DL.

Atualmente, está em desenvolvimento uma extensão e revisão da versão publicada em 2004 (OWL 1), trata-se da OWL 2 que assim como a OWL 1, é projetada para o desenvolvimento e compartilhamento de ontologias via Web, tendo como objetivo principal fazer com que as máquinas tenham acesso facilitado ao seu conteúdo, assim como também possam adquirir conhecimento sobre ele, como já era a ideia da primeira versão [35]. Por essa razão, pela expressividade e também pela capacidade de raciocínio a OWL 2 é a versão utilizada neste trabalho.

2.3.4.6 *Lógica de Descrição*

DL - *Description Logic* ou lógica de descrição é uma linguagem formal para representação do conhecimento aplicada na modelagem de ontologias [62], a qual também serve para raciocinar sobre esse conhecimento, além de servir de base à OWL. Ela é fundamentada em um fragmento de lógica de primeira ordem, o que permite o raciocínio automático que possibilita computar não só a hierarquia de classes, como também verificar possíveis inconsistências na ontologia.

Fazendo uma relação com a programação orientada a objetos, um conceito em DL equivale a uma classe de objeto, as propriedades seriam os métodos e as relações entre as classes de objetos existentes, o domínio é o sistema que está sendo desenvolvido, e finalmente as regras (propriedades) representam relações binárias conectando indivíduos.

A representação do conhecimento com base na DL fornece métodos para definir bases de conhecimentos a fim de inferir consequências lógicas sobre o seu conteúdo e também para manipular estas bases. Uma base de conhecimento é composta por três componentes:

- **RBox** - é responsável pela descrição das *roles* e suas propriedades.
- **TBox** - responsável pela terminologia do sistema, ou seja, por todo o vocabulário existente no domínio de uma aplicação.
- **ABox** - contém afirmações sobre os indivíduos no que diz respeito ao vocabulário da TBox. Por outras palavras, a ABox contém o conhecimento sobre os indivíduos do domínio em questão. Estas afirmações indicam como se relacionam os objetos e a que conceitos pertencem.

No caso da ontologia aqui apresentada, a linguagem baseada em Lógica de Descrição usada em seu desenvolvimento foi a OWL-DL.

2.3.4.7 SPARQL

SPARQL é um protocolo de acesso e linguagem de consulta a dados em RDF, e é recomendada pela W3C. O nome SPARQL é um acrônimo recursivo cujo o significado é *SPARQL Protocol and RDF Query Language* [25].

Por ser uma linguagem de consulta, a SPARQL é orientada a dados, ou seja, não existe inferência nela. Uma de suas características é a aplicação com outras tecnologias de web semântica, tais como o RDF, para representação de dados e OWL para construção de vocabulários, só para citar algumas.

Uma consulta SPARQL é constituída de três partes [73]:

- **Pattern matching** - inclui várias características interessantes de combinação de padrões de gráficos, como peças opcionais, união de padrões, aninhamento, valores de filtragem de possíveis correspondências e a possibilidade de escolher a fonte de dados para ser correspondida por um padrão.
- **Solution modifiers** - uma vez que a saída do padrão foi calculada (na forma de uma tabela de valores de variáveis), permitem modificar esses valores aplicando operadores clássicos como PROJECTION, LIMIT, DISTINCT, ORDER e OFFSET.
- **Output** - podem ser de diferentes tipos como: booleanas, seleções, gráficos, seleções dos valores das variáveis que correspondem aos padrões, construção de novos dados RDF desses valores e descrições de recursos.

Abaixo são apresentadas as principais cláusulas que compõem a linguagem SPARQL:

- **SELECT** - permite selecionar quais informações serão retornadas como resultado da consulta. As informações são armazenadas em variáveis que são identificadas por um ponto de interrogação (?).
- **WHERE** - permite especificar restrições quando da realização de consultas. As restrições seguem o formato de tripla <sujeito, predicado, objeto> que podem ser formadas tanto por um objeto quanto por um valor literal.
- **FILTER** - Restringe o conjunto de soluções de acordo com uma ou mais expressões. As expressões podem ser funções e operações construídas sintaticamente. Os operandos e os operadores dessas funções são um subconjunto dos tipos de dados do XML Schema (*xsd:string*, *xsd:decimal*, *xsd:double*, *xsd:dateTime*) e tipos derivados de *xsd:decimal*.

- **ORDER BY** - Captura uma sequência de solução e aplica sobre ela condições de ordenação. Uma condição de ordenação pode ser uma variável ou a chamada de uma função. A direção de ordenação é ascendente por padrão. Porém, é possível explicitamente informar a direção de ordenação em ascendente e decrescente, através das abreviaturas ASC e DESC respectivamente.
- **LIMIT** - limita o número de soluções retornadas. Se o número de soluções reais é maior do que o limite, então no máximo o número limite de soluções será retornado. A cláusula FROM é opcional, visto que a consulta é realizada sobre a ontologia ou modelo no qual se está trabalhando.

2.3.4.8 Ferramenta semântica

Como citado no início desta seção, as linguagens aqui descritas serão utilizadas no desenvolvimento da ontologia proposta, em virtude disso fez-se necessária a escolha de uma ferramenta que oferecesse suporte a elas. Dessa forma, para auxílio ao desenvolvimento da ontologia proposta neste trabalho foi utilizada a ferramenta Protégé, a qual foi desenvolvida pelo *Biomedical Research Informatics* da Universidade de Stanford. Ela possui código aberto, uma linguagem baseada em lógica descritiva (OWL-DL) e é composta por um editor de ontologia e uma biblioteca de *plugins* com várias funcionalidades. Outra vantagem desta ferramenta é que ela faz a importação/exportação de/para diversas linguagens. Além disso, o Protégé foi a ferramenta adotada na maioria dos trabalhos avaliados nas pesquisas realizadas para o desenvolvimento do trabalho aqui proposto.

Há um ano atrás, quando iniciaram-se os trabalhos de pesquisa para o desenvolvimento da proposta aqui apresentada, a versão utilizada do Protégé foi a 4.3. Na época já havia a versão beta 5.0, porém no computador utilizado para as pesquisas esta versão não funcionou a contento. Já no início deste ano, quando teve início o desenvolvimento da ontologia, foi feita uma nova tentativa de atualização da versão, agora com sucesso. Dessa forma, no desenvolvimento da ontologia e realização dos testes foi utilizado o Protégé 5.2.

A plataforma Protégé suporta duas principais formas de modelar ontologias:

- **Protégé-Frames editor:** habilita usuários a construir e instanciar ontologias que são baseadas em *frames*. Neste modelo, uma ontologia consiste de um conjunto de classes organizadas em uma hierarquia para representar um domínio de conceitos, um conjunto de *slots* associados às classes descrevendo suas propriedades e relacionamentos, e o conjunto de instâncias destas classes.
- **Protégé-OWL editor:** habilita usuários a construir ontologias utilizando a OWL. Uma ontologia OWL pode incluir descrições de classes, propriedades e suas instâncias.

2.3.4.9 Motor de inferência

Para que seja possível extrair o conhecimento de um domínio a partir de uma ontologia é necessário o uso de um motor de inferência ou raciocinador (*reasoner* em inglês) automático. Segundo [14], um dos principais usos da representação de conhecimento é combiná-la com um raciocinador automático, para realizar inferências e obter novos fatos. Ainda segundo o autor, trata-se de um *software* capaz de inferir consequências lógicas a partir de um conjunto de fatos declarados ou axiomas. Essas regras de inferência são definidas através da ontologia [8].

Um motor de inferência é um *software* que tem a tarefa de realizar inferências lógicas, para isso segue as regras definidas na ontologia, o que possibilita a identificação de inconsistências, dependências escondidas e eventuais redundâncias. Além disso, o motor de inferência extrai novos conhecimentos a partir daqueles já descritos nas ontologias, permitindo que sejam agregados e apresentados de forma transparente.

O Pellet é um motor de inferência OWL DL baseado em algoritmos de Tableau, suporta total expressividade OWL DL incluindo raciocínio sobre nominais (classes enumeradas). Ele garante a integridade e completude incorporando o processo de decisão recentemente desenvolvido para SHOIQ.

Funcionalidades suportadas pelo Pellet:

- serviços de raciocínio
- restrições de cardinalidade
- axiomas complexos
- propriedades disjuntas
- propriedades reflexivas, irreflexivas, simétricas e assimétricas
- partilha de vocabulário entre indivíduos, classes e propriedades
- definição de propriedades negativas

Apesar do Protégé 5.2 oferecer outros motores de inferência, tais como FaCT++ e Hermit, o Pellet, que já está na sua versão 2.2.0, foi o escolhido para ser aplicado a ontologia proposta. Os motivos da escolha é porque o Pellet suporta a OWL 2, oferece explicação sobre a ocorrência de erros na ontologia e também por ser citado na maior parte das bibliografias estudadas, fato que indica um grande número de usuários, que é um facilitador para obtenção de suporte técnico quando necessário.

2.3.5 SWRL - Semantic Web Rule Language

Nos últimos anos a ampliação do uso da Web Semântica [78] e de Ontologias [71], no intuito de fazer com que os conteúdos da Web tenham significado tanto para pessoas

quanto para máquinas [90], fez crescer o interesse dos pesquisadores por essas tecnologias que possibilitam a comunicação e compartilhamento de conhecimento [78].

O uso de ontologias como forma de representar o conhecimento possibilita que os dados gerados sejam anotados semanticamente, algo que pode ser feito através da SWRL - *Semantic Web Rule Language* no formato de regras com base nos conceitos OWL e por meio de afirmações condicionais, representar, organizar e compartilhar o conhecimento de um domínio específico [77].

Conforme [2], a SWRL é uma linguagem que resulta da união da lógica Horn e da DL, com o objetivo de alcançar uma maior expressividade e capacidade de raciocínio sobre o domínio em comparação com o uso somente de OWL, pois ela possui regras expressivas e é projetada para permitir asserções declarativas usando conceitos OWL.

Uma regra SWRL é constituída de duas partes, a parte antecedente e a parte consequente. Quando os conceitos atômicos da parte antecedente são verdadeiros, então a consequência também é verdadeira [88]. As duas partes podem ser a conjunção de zero ou mais átomos, porém não permite disjunção e negação. Já os átomos possuem um predicado e um ou mais argumentos.

Na figura 20 é exibida uma regra SWRL (representada no formato para humanos), onde a seta (\rightarrow) separa o antecedente do consequente, o acento circunflexo (\wedge) indica a conjunção entre os átomos e o ponto de interrogação (?) serve para distinguir nomes de variáveis de nomes de indivíduos

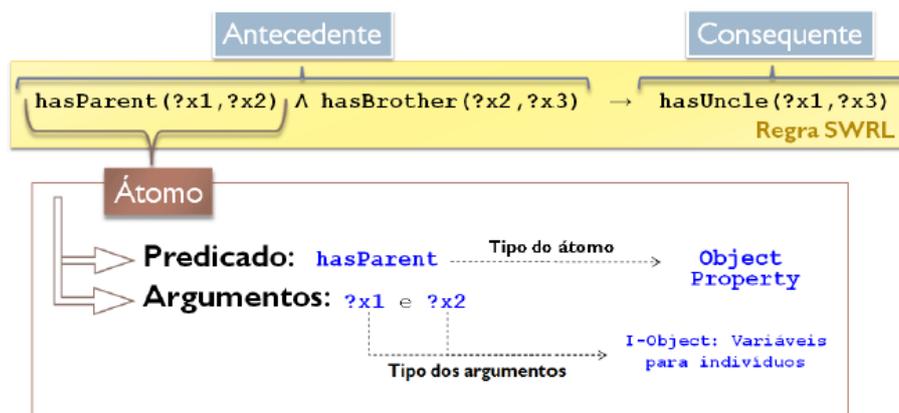


Figura 20: Regra SWRL

A regra exibida na figura 20 tem o seguinte significado: se um indivíduo **?x2** é pai do indivíduo **?x1** e o indivíduo **?x3** é irmão do indivíduo **?x2**, então o indivíduo **?x3** é tio de **?x1**.

A especificação SWRL não impõe restrições sobre como o raciocínio deve ser executado com as regras SWRL. Assim, é livre usar uma variedade de mecanismos de regras SWRL para raciocínio com as regras armazenadas em uma base de conhecimento OWL, podendo ainda implementar suas próprias instalações de edição para criar regras SWRL

[68] [71].

[77] afirmam que a SWRL faz com que a expressividade das ontologias OWL seja ampliada, isso é conseguido através de afirmações condicionais (regras) que possibilitam a inferência de novos conhecimentos sobre os indivíduos da ontologia. A este respeito, [87] acrescenta que a representação do domínio é feita pela ontologia, enquanto que a função da SWRL é representar, através de regras, condições que com a OWL é impossível de expressar. Depois de criadas, as regras SWRL ficam armazenadas em formato OWL na própria ontologia [77], ou seja, elas passam a fazer parte das linhas de código da ontologia.

Porém, a OWL-DL tem sua expressividade limitada em razão de não oferecer suporte a regras [87] [2], fazendo com que as vezes seja preciso expressar definições que somente são possíveis de criar com regras, para isso o autor dá como exemplo uma situação na qual não é possível de ser expressa com a OWL-DL. É o caso da sentença "o irmão do pai de uma pessoa é tio desta pessoa". A SWRL foi desenvolvida para suprir tal limitação e é uma solução recomendada pelo W3C.

Um aspecto que merece atenção é o aumento da complexidade da estrutura a partir da integração de novas regras, podendo trazer problemas de gerenciamento, pois um número grande de regras torna o entendimento difícil, e com maiores possibilidades de ocorrência de erros, ainda mais se o controle é feito por uma só pessoa, o que faz surgir a necessidade do desenvolvimento de ferramentas para criar, visualizar e gerenciar regras, e que possibilitem principalmente [77]:

- adquirir o conhecimento sem inconsistências, ambiguidade e regras duplicadas
- visualização de regras e conjuntos de regras de forma a facilitar o entendimento e conhecimento das mesmas

Em seu trabalho, [2] apresenta algumas vantagens e desvantagens no uso da SWRL. As vantagens são:

- capacidade de usar nomes de classe ou suas descrições como predicados
- uso de igualdades e desigualdades
- permitir conjunções de átomos no antecedente e no conseqüente da SWRL
- os axiomas SWRL podem ser usados na transferência de características de uma classe ou propriedade para outra sem sub-classificar - o que ultrapassa os poderes expressivos da OWL
- as regras SWRL permitem a inferência de novos indivíduos usando operadores existenciais

- a SWRL sendo uma extensão semântica da OWL tem sido objeto de interesse de especialistas, além do apoio da comunidade de pesquisa da Web Semântica.

E as desvantagens são:

- limitação na modelagem de conhecimento de domínio impreciso. A natureza evolutiva e às vezes inconsistente do conhecimento humano requer a representação de informações de domínio vagas ou imprecisas
- falta de construções não monotônicas. A modelagem baseada em regras de um domínio de conhecimento envolve expressar fatos de domínio ou situações de objetos de domínio, através do uso de declarações "if-then". Esta representação ordenada de fato, reflete basicamente a natureza ordenada do conhecimento humano - que geralmente inclui ordenar ou estabelecer uma precedência de atividades, levando à adição de novos fatos a bases de conhecimento existentes. Além disso, demonstrou-se que as formas dedutivas de inferência, quando os novos fatos são inferidos, não podem ser representados pelo formalismo SWRL
- necessidade de fatos de "desencadeamento" em bases de conhecimento, ou seja, remoção de fatos. As construções necessárias para modelar e alcançar tais cenários encontrados não tem definição SWRL clássica
- falta de suporte para modelagem de conhecimento científico complexo. Outras limitações incluem a incapacidade da SWRL de modelar eficientemente os domínios do conhecimento científico e de engenharia, especialmente aqueles que envolvem fórmulas e restrições matemáticas complexas

A apresentação da SWRL neste trabalho deve-se não só pelas características positivas que ela apresenta, sendo que talvez a mais importante é a capacidade de aumentar a expressividade da linguagem OWL, mas principalmente porque o projeto da construção da ontologia previa que as características que determinariam a existência de fusão de sensores nas instalações de uma planta industrial deveriam ser implementadas através de uma linguagem de regras.

3 TRABALHO PROPOSTO

Neste capítulo é feita a descrição do desenvolvimento da ontologia proposta, apresentando os conceitos definidos, a modelagem do domínio a partir da identificação dos componentes que podem estar integrados a estrutura onde a fusão de sensores é aplicada. Além disso, é apresentada a forma como será feita a identificação das ocorrências de divergência de dados entre os sensores utilizados em fusão.

Será apresentada também uma solução desenvolvida com a AutomationML para permitir a integridade na troca de dados entre sistemas de controle industriais e a ontologia desenvolvida. E finalmente, é mostrada a validação da proposta.

3.1 A ontologia

Dentro da sequência definida na metodologia utilizada (Methontology) para o desenvolvimento da ontologia, foi preciso transformar o conhecimento adquirido nas pesquisas realizadas em conceitos sobre o domínio estudado, mais especificamente ao uso da fusão de sensores na manufatura. Dessa forma foi feita a identificação dos componentes envolvidos neste tipo de estrutura e a relação entre eles, para que fosse definida a taxonomia da ontologia. Também foi preciso definir a quem a ontologia se destina, ou seja, quais seus usuários. Esta definição está representada na figura 21, onde é possível observar os níveis da pirâmide de automação com seus respectivos tipos de usuários e quais deles fazem parte do escopo da abrangência da ontologia.

Na ontologia foram definidas 26 classes, 32 propriedades de objetos e apresenta ainda 4 módulos distintos, mas relacionados (figura 21): (a) componentes físicos, (b) fusão de sensores, no qual podem ser identificados os elementos relacionados a fusão de sensores, (c) parâmetros das medições para cada fenômeno a ser medido (pressão, fluxo,...) para cada um dos recursos e (d) ocorrências de divergências nas medições dos sensores em fusão.

Para definição da taxonomia foram identificadas três opções: ou usava-se a de uma ontologia existente, ou criar uma própria ou utilizar os conceitos definidos na AML, já que a mesma foi objeto de estudo nas pesquisas realizadas e está inserida no contexto da

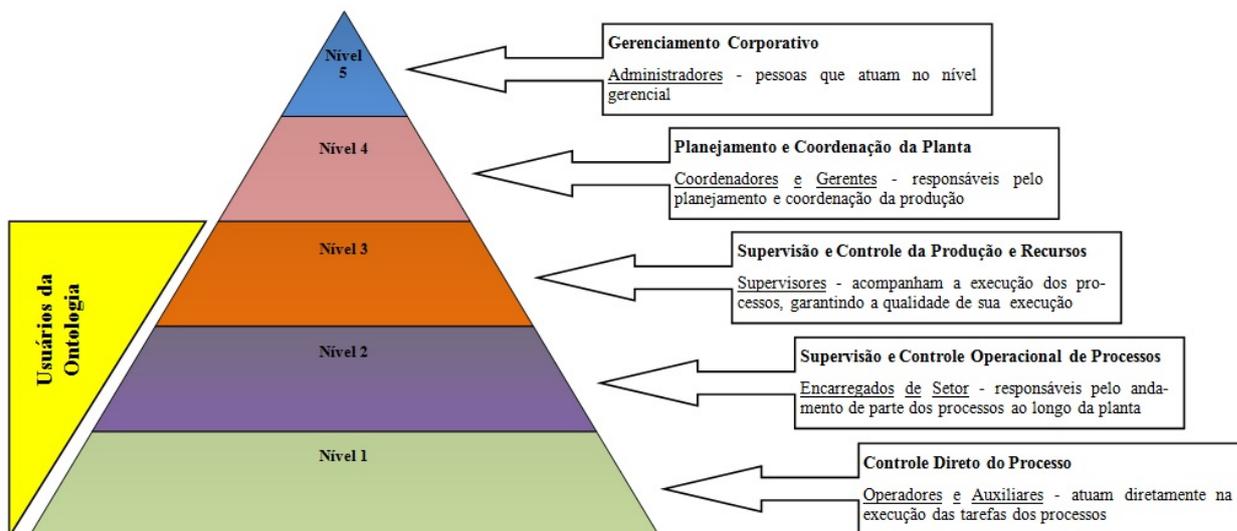


Figura 21: Abrangência da ontologia. Fonte: Elaborado pelo autor

proposta aqui apresentada.

Como grande parte dos termos utilizados pela AML são fundamentados em padrões da IEC, fez-se então a opção de escolher em um primeiro momento, dentro do escopo oferecido pela AML, aqueles termos cuja a semântica melhor identificasse as classes que compõem a ontologia, e nas situações onde não foi possível esse aproveitamento foram definidos termos mais apropriados.

3.1.1 Formalização dos elementos da planta

A conceituação dos elementos físicos da planta é apresentada no módulo **Element** definido com uma taxonomia hierárquica das estruturas físicas (edifícios, armazéns, depósitos,...), linhas de produção, recursos (equipamentos, máquinas,...) e, por fim os tipos de sensores divididos em sub-classes. A definição deste módulo foi baseada na descrição hierárquica definida na norma IEC 62264-1:2013, a qual é exibida na figura 22.

A seguir são mostradas as definições das classes que fazem parte do módulo **Element**, todas as subclasses deste módulo foram declaradas como disjuntas, isso significa que um indivíduo somente pode estar vinculado a uma delas. Por exemplo, um prédio qualquer não pode ser um site e uma área ao mesmo tempo, ou é um ou é outro.

Classe SITE

A classe **Site** deve ser usada para determinação da posição de um site ou como elemento de uma organização hierárquica.

A IEC 62264-1: 2013, 5.3.3 define SITE como: "Um site é um agrupamento físico, geográfico ou lógico determinado pela empresa, podendo conter áreas, linhas de produção, células de processo e unidades de produção". Ex: planta, instalação de fabricação.

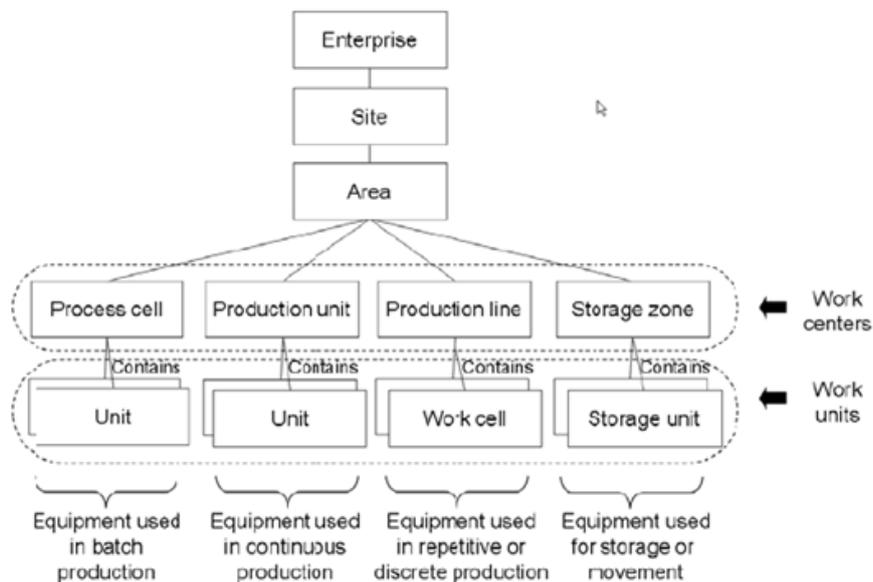


Figura 22: Estrutura de Recurso. Fonte: [18]

No caso da ontologia, um site pode ser composto por várias áreas.

Classe **AREA**

A classe de **Area** deve ser utilizada para edificações de produção e suas subdivisões (estrutura/salão). Também é usada como elemento de uma organização hierárquica.

A IEC 62264-1: 2013, 5.3.4 define **Area** como: "Uma área é um agrupamento físico, geográfico ou lógico determinado pelo site. Pode conter centros de trabalho como células de processo, unidades de produção, linhas de produção e zonas de armazenamento." Ex: um *hall*

No caso da ontologia, uma área só pode estar relacionada a um único site, isso é possível através da definição da propriedade inversa *isAreaOf* declarada como funcional, e de seu *domain* e *range*. Já uma área pode ser composta por várias linhas de produção. Cabe ressaltar que poderia também ter sido considerado que na composição de uma área poderiam existir outras áreas ou outras sub-divisões como existem nos conceitos da AML. Porém, este é outro caso que só serviria para aumentar o nível de detalhes mas sem interferência significativa com relação ao que aqui se quer provar.

A atribuição de propriedades inversas nos relacionamentos das classes do módulo **Element**, tem a intenção de fazer com que a relação ocorra nos dois sentidos, isto é, a propriedade relaciona o indivíduo "A" ao indivíduo "B", então a propriedade inversa relacionará "B" com "A".

Classe **PRODUCTIONLINE**

A classe **ProductionLine** deve ser usada para definir o papel básico do equipamento na hierarquia definida na IEC 62264-1: 2013, 5.3.7: "As linhas de produção e as células

de trabalho são os níveis mais baixos de equipamentos ... para processos de fabricação discreta. As células de trabalho são geralmente identificadas apenas quando há roteamento de trabalho dentro de uma linha de produção. As linhas de produção e as células de trabalho podem ser elementos de nível inferior A principal atividade de processamento geralmente identifica a linha de produção.”

No caso da ontologia, uma linha de produção só pode estar relacionada a uma única área, isso é possível através da definição da propriedade inversa *isProductionLineOf* declarada como funcional, e de seu *domain* e *range*. Mas, pode ser composta por vários recursos.

Classe **RESOURCE**

A classe **Resource** é um tipo básico abstrato da AML, ou seja, não é padrão IEC. Ela serve para descrever plantas, equipamentos ou outros recursos de produção.

No caso da ontologia aqui apresentada, são considerados recursos todos os equipamentos (tanque, tubulação,...) onde os sensores em fusão estão diretamente instalados. Além disso, um recurso só pode estar relacionado a uma única linha de produção, isso é possível através da definição da propriedade inversa *isResourceOf* declarada como funcional, que na prática significa que o recurso não poderá pertencer a duas linhas de produção diferentes ao mesmo tempo. Para esse tipo de restrição funcionar é necessário definir corretamente o *domain* e *range* da propriedade, que no caso da ontologia são a classe *Resource* e a classe *ProductionLine*, respectivamente. Já o recurso, por sua vez, pode estar relacionado a vários sensores de vários tipos, porém a propriedade usada não é a mesma.

Classe **SENSORTYPE**

A classe **SensorType** deve ser usada para sensores. Assim como **Resource**, ela também é um tipo abstrato AML. No caso da ontologia, foi necessário alterar o nome original de *Sensor* para **SensorType** a fim de poder representar os diferentes tipos de sensores que podem ser utilizados. As subclasses representando os tipos de sensores foram declaradas como disjuntas, para evitar que um indivíduo seja atribuído a dois tipos de sensores diferentes.

No caso da ontologia, um sensor, independente do tipo, só pode estar relacionado a um único recurso, isso é possível através da definição das propriedades inversas de cada um dos tipos de sensor (figura 23), como funcional (da mesma forma como descrito na classe **Resource**). Só que no caso dos sensores, como há diferenciação pelo tipo, o *domain* de cada uma delas será a classe correspondente ao tipo de sensor que será utilizado na relação, mas a *range* é a mesma para todas elas, é a classe **Resource**.

Como já dito, a AML oferece outros termos que poderiam ser inseridos na ontologia, porém optou-se por construir um modelo enxuto, com menos complexidade, pois de nada adiantaria criar uma ontologia com muitas classes, como por exemplo um número

maior de tipos de sensores, já que este trabalho não visa apresentar uma solução robusta para suportar um número grande de componentes. Aqui, o que se quer provar é que a estrutura da ontologia proposta tem condições de representar basicamente um ambiente fabril e nele identificar os elementos de produção envolvidos (direta ou indiretamente) na aplicação de fusão de sensores para o controle dos processos produtivos. Além disso, provar também que ele tem condições de identificar a ocorrência de divergências de dados entre os sensores de mesmo tipo operando em fusão. A figura 23 exhibe as classes do módulo **Element**.

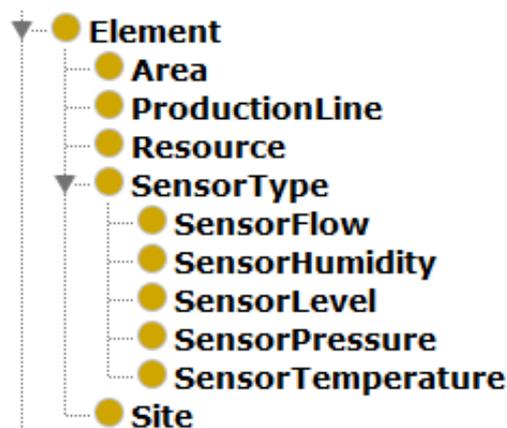


Figura 23: Módulo Element. Fonte: Elaborado pelo autor

As subclasses deste módulo foram declaradas como disjuntas, isso significa que um indivíduo somente pode estar vinculado a uma delas. Por exemplo, um prédio qualquer não pode ser um site e uma área ao mesmo tempo, ou é um ou é outro.

Na figura 24 são mostradas todas as propriedades do módulo **Element**.

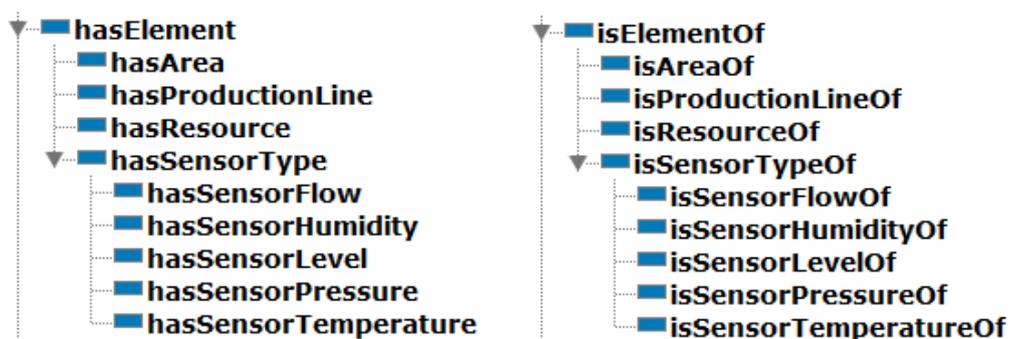


Figura 24: Propriedades de objetos do módulo Element, as do lado direito são inversas das do lado esquerdo. Fonte: Elaborado pelo autor

A tabela 2 mostra a lista das propriedades de objetos ligadas ao módulo **Element** com seus respectivos domínios (*domain*) e escopo (*range*):

Na tabela 2, foi mostrado apenas o exemplo de uso da propriedade relativa aos sensores de fluxo, para os demais tipos segue-se o mesmo raciocínio.

Tabela 2: Propriedades de objeto - Módulo Element

Data Property	Domain	Range
hasArea	Site	Area
hasProductionLine	Area	ProductionLine
hasResource	ProductionLine	Resource
hasSensorFlow	Resource	SensorFlow

Após a definição das propriedades o próximo passo foi declarar todas as classes do módulo como primitivas, ou seja, definiu-se as restrições para os indivíduos que irão compor cada uma das classes. Para isso foram implementados os seguintes axiomas:

- classe **Site** - um site é um *Element* que tem pelo menos uma *Area*:

$$\text{Site} \equiv \text{Element} \cap \exists \text{hasArea.Area}$$

- classe **Area** - uma área é um *Element* que tem pelo menos uma *ProductionLine*:

$$\text{Area} \equiv \text{Element} \cap \exists \text{hasProductionLine.ProductionLine}$$

- classe **ProductionLine** - uma linha de produção é um *Element* que tem pelo menos um *Resource*:

$$\text{ProductionLine} \equiv \text{Element} \cap \exists \text{hasResource.Resource}$$

- classe **Resource** - um recurso é um *Element* que tem pelo menos um *SensorFlow*:

$$\text{Resource} \equiv \text{Element} \cap \exists \text{hasSensorFlow.SensorFlow}$$

Importante ressaltar que para cada tipo de sensor foi definido um axioma seguindo o modelo acima apresentado.

3.1.2 Formalização da fusão de sensores

O módulo **SensorFusion** relativo a fusão de sensores propriamente dita, é composto por classes relacionadas ao tipo de fusão, o qual é determinado pelo tipo de sensor utilizado, servindo para identificar os indivíduos (sites, áreas, linhas de produção, recursos e sensores) envolvidos com a fusão de sensores. Todas as classes deste módulo são classes de nível superior, pois representam a abstração da ocorrência de determinada condição, no caso a fusão de sensores. Ou seja, os indivíduos desta classe só farão parte dela de forma lógica e não física. Por exemplo, os indivíduos que pertencerem a uma das classes do módulo **Element**, quando criados devem ser instanciados nas classes a qual pertencem, se é uma área então deve pertencer a classe **Area**, se é sensor de fluxo deve pertencer a classe **SensorFlow** e assim por diante. No caso das classes superiores, os indivíduos são relacionados a elas através de regras SWRL, ou seja, é preciso acionar o motor de

inferência para que a condição definida para identificar a ocorrência de fusão de sensores (um recurso e dois sensores de mesmo tipo) tenha efeito. Sem acionar o motor de inferência estas classes não possuem indivíduo algum, diferentemente dos indivíduos relacionados com as classes do módulo **Element**, por exemplo, os quais podem ser consultados sem uso do motor de inferência. Isto deve-se ao fato dos indivíduos estarem fisicamente ligados as classes.

As subclasses desse módulo foram declaradas disjuntas para evitar que um indivíduo do tipo sensor não esteja envolvido em fusão com sensores de tipos diferentes.

Com a finalidade de facilitar o entendimento sobre este módulo, foi definido que indivíduos como sites, áreas, linhas de produção e recursos, quando da ocorrência de fusão, ficam instanciados logicamente na classe **SensorFusion**, enquanto que os sensores ficam instanciados logicamente na classe correspondente ao seu tipo. A estrutura deste módulo é apresentada na figura 25.

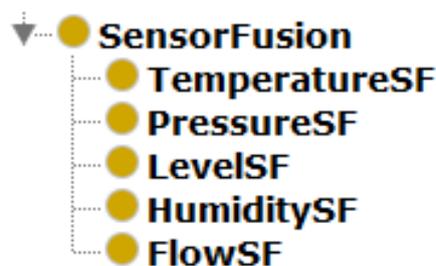


Figura 25: Módulo SensorFusion. Fonte: Elaborado pelo autor

Também, diferentemente das classes do módulo **Element**, as restrições das classes do módulo **SensorFusion** foram implementadas através de regras construídas com a SWRL, sendo assim não há nenhuma propriedade ligada a ela. Tais regras serão explicadas e mostradas na seção da validação da proposta, ainda neste capítulo.

3.1.3 Formalização dos parâmetros

O módulo **Parameter** é composto pelas classes que representam os indivíduos que contém os parâmetros de cada recurso utilizado na fusão de sensores para cada um dos tipos de sensor descritos na ontologia. Os indivíduos que compõem cada uma delas são na verdade parâmetros que possuem duas propriedades de dados (*DataProperties*) contendo os valores máximo e mínimo permitido em cada recurso para cada tipo de sensor utilizado em fusão. Isto é, para cada recurso que tenha algum tipo de sensor em fusão deverá existir um indivíduo (parâmetro) contendo os limites permitidos para cada um dos tipos de medições realizados, de acordo com o tipo de sensor utilizado. A figura 26 exhibe as propriedades de objeto do módulo **Parameter**.

As subclasses no módulo **Parameter** são todas disjuntas evitando assim que um parâmetro seja utilizado por dois recursos distintos.



Figura 26: Propriedades dos parâmetros. Fonte: Elaborado pelo autor

As propriedades ligadas ao módulo **Parameter** são específicas para cada tipo de sensor, porém todas tem o mesmo *Domain*, que é a classe **Resource**, mudando somente a *range* que deverá ser a classe relativa ao tipo de sensor empregado.

Na figura 27 são mostradas todas as propriedades de dados (*DataProperties*) relativas aos tipos de parâmetros.

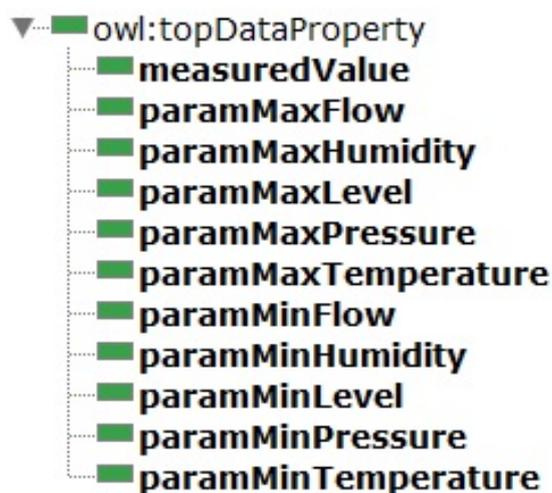


Figura 27: Propriedades de dados dos tipos de parâmetros. Fonte: Elaborado pelo autor

3.1.4 Formalização das divergências

O módulo **Divergence** é composto pelas classes **DivergenceDown** e **DivergenceUp**. Elas também são classes de nível superior, sendo que **DivergenceDown** representa a abstração das ocorrências de divergências abaixo do parâmetro mínimo definido para medição de um determinado fenômeno em um determinado recurso. Os parâmetros utilizados devem ser específicos para o tipo de sensor. Já a classe **DivergenceUp** representa a abstração das ocorrências de divergências acima do parâmetro máximo definido para medição de um determinado fenômeno em um determinado recurso.

Neste módulo, as restrições das duas classes também foram implementadas através de regras construídas com a SWRL, sendo assim não há nenhuma propriedade ligada a elas. Tais regras serão explicadas e mostradas na seção da validação da proposta, ainda neste

capítulo. A figura 28 mostra as classes do módulo **Divergence**.



Figura 28: Módulo Divergence. Fonte: Elaborado pelo autor

O resultado final da estrutura da ontologia, após as definições e implementações, pode ser visto no Anexo A. E, na figura 29, é mostrada uma visão geral da integração dos diferentes componentes da proposta apresentada.

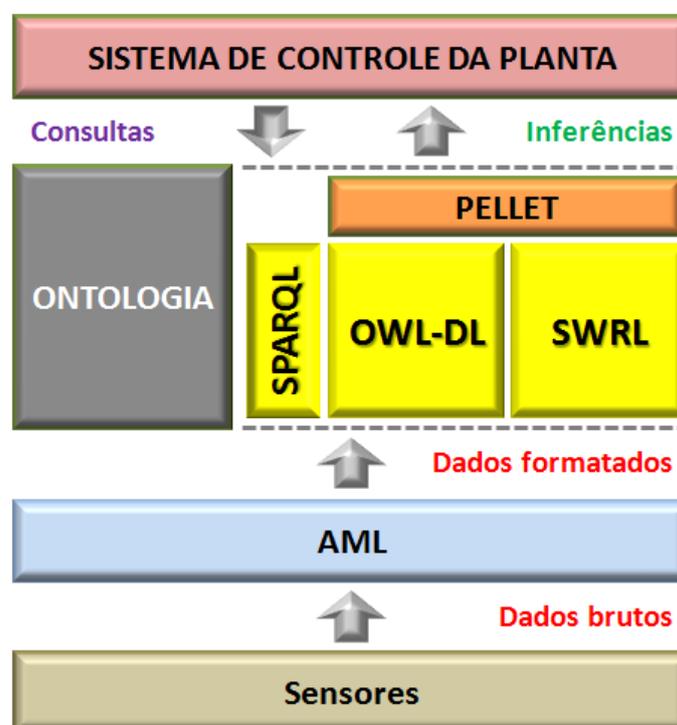


Figura 29: Visão geral da integração dos componentes da proposta. Fonte: Elaborado pelo autor

4 VALIDAÇÃO DA PROPOSTA

Para validar a proposta aqui apresentada foram realizados diferentes tipos de testes a fim de confirmar que a lógica de funcionamento aplicada sobre a estrutura oferece confiabilidade para ser utilizada em ambientes de produção. No primeiro momento foi definida a estratégia a ser utilizada na realização dos referidos testes. Foi estabelecido que haveriam duas plantas, cada uma com sua própria estrutura. Então foram definidas a *planta1* e a *plantaA* com as seguintes composições:

- Planta1 - Site: *planta1* / Area: *predio1* / ProductionLine: *smar1* / Resource: *tanque1* / *tanque1* contém os sensores de nível: *sl1*, *sl2* e *sl3* / *tanque1* possui um parâmetro de nível: *level1*
- PlantaA - Site: *plantaA* / Area: *predioA* / ProductionLine: *smarA* / Resource: *tanqueA* / *tanqueA* contém os sensores de nível: *slA*, *slB* / *tanqueA* possui um parâmetro de nível: *levelA*

Com a finalidade de facilitar a apresentação da proposta, bem como a construção deste texto, os exemplos apresentados na continuação são todos referentes ao uso de sensores de nível. Porém, cabe salientar que tais regras servem para todos os tipos, bastando somente trocar o nome do tipo de sensor em cada um das regras. Como os testes não foram executados dentro de um sistema real, não houve necessidade de implantar todas as regras para cada tipo de sensor.

O próximo passo foi definir os valores dos parâmetros para controle das divergências e instância-los na ontologia. Feito isso, foi testada então a entrada automática dos dados na ontologia, isto é, foi simulado um sistema de controle industrial que transfere seus dados para a ontologia a fim de que ela possa fazer suas inferências sobre eles e retornar os resultados. Então foi desenvolvido um aplicativo em linguagem C especialmente para fazer a simulação da transferência de dados para a ontologia.

Para a realização dos testes foi utilizado o arquivo XML gerado pela AML com a modelagem da planta didática SMAR, a qual está localizada no Laboratório de Infraestrutura do Centro de Ciências Computacionais da Universidade Federal do Rio Grande. A figura 30 apresenta o fluxograma da simulação realizada nos testes de validação.



Figura 30: Fluxograma da simulação. Fonte: Elaborado pelo autor

O uso do arquivo XML com a modelagem da planta foi para assegurar a integridade dos dados transferidos do aplicativo simulador para a ontologia. Sendo assim, se procedeu da seguinte maneira: os dados foram lançados no aplicativo simulador, que por sua vez fez a gravação dos mesmos no arquivo XML com a modelagem AML da SMAR. Depois o arquivo foi lido novamente e extraídos os dados (valores das medições) relativos aos sensores, os quais foram gravados no arquivo XML da ontologia para que finalmente ela pudesse ser testada. Essas ações de gravação e posterior leitura do arquivo XML foram para simular as duas etapas diferentes no processo, a saída dos dados do sistema de origem e sua gravação no arquivo AML, e a entrada dos dados na ontologia a partir deste arquivo. Na figura 31, é mostrada uma imagem parcial da modelagem AML da planta SMAR, a qual originou o arquivo utilizado na exportação/importação dos dados entre o sistema de controle da planta e a ontologia (os módulos referentes aos processos e produtos, bem como os relacionamentos entre os três módulos não puderam ser mostrados já que a imagem completa não caberia neste documento).

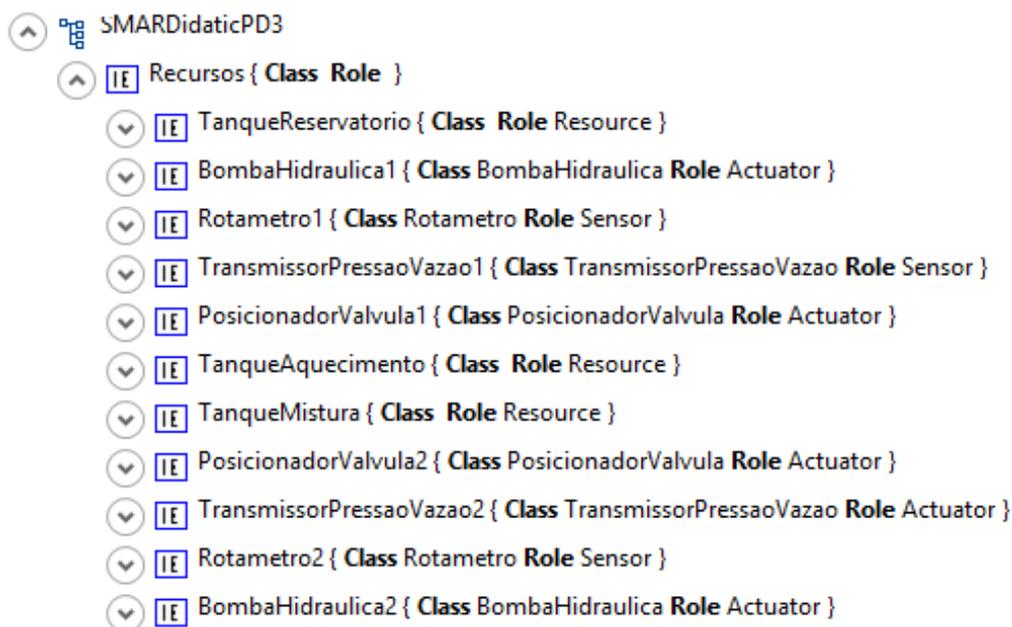


Figura 31: Modelagem AML da planta SMAR. Fonte: Elaborado pelo autor

Importante salientar que todos os dados que são transferidos para a ontologia ficam gravados no arquivo XML que contém a própria ontologia. Por esse motivo, deve-se

ter atenção com relação ao tamanho que esse arquivo tem, pois arquivos muito grandes podem não ser suportados pelo Protégé.

Após os dados serem gravados no arquivo XML da ontologia, foram realizados os testes de validação. Cabe salientar que existem muitas hipóteses a serem testadas na avaliação de um *software*, por esse motivo foram testadas apenas aquelas consideradas necessárias à comprovação da proposta apresentada neste trabalho. Então foram definidos cinco questionamentos para serem respondidos pela ontologia, são eles:

1. quais os recursos e quais os sensores que eles contém estão em fusão?
2. quais as linhas de produção que contém fusão de sensores?
3. quais as áreas que contém fusão de sensores?
4. quais os sites que contém fusão de sensores?
5. há divergência de medição entre os sensores que estão em fusão em determinado recurso?

A seguir são apresentadas as ponderações e respostas para cada um destes questionamentos.

- **Quais os recursos e quais os sensores que eles contém estão em fusão?**

Esse questionamento é o primeiro porque a resposta à ele é a base para responder aos demais questionamentos. Ele também é o único que é composto por duas situações de dúvida. A primeira é saber quais recursos contém sensores em fusão, e a segunda é saber quais são esses sensores e conseqüentemente seu tipo.

A razão por este questionamento ser composto é porque essas duas condições é que permitiram a definição de uma regra para caracterizar logicamente a ocorrência de fusão de sensores.

A regra definida foi a seguinte: há ocorrência de fusão de sensores quando um recurso for composto por no mínimo dois sensores do mesmo tipo. Essa situação é caracterizada pela seguinte regra SWRL, a qual é válida apenas para sensores de nível:

```
Resource(?r), SensorLevel(?s11), SensorLevel(?s12),
hasSensorLevel(?r, ?s11), hasSensorLevel(?r, ?s12),
DifferentFrom(?s11, ?s12) -> SensorFusion(?r),
LevelSF(?s11), LevelSF(?s12)
```

Significado da regra:

?r é um recurso, ?sl1 é um sensor de nível, ?sl2 é um sensor de nível, ?r tem sensor de nível ?sl1, ?r tem sensor de nível ?sl2, ?sl1 é diferente de ?sl2 ENTÃO ?r tem fusão de sensores, o tipo de fusão do sensor ?sl1 é de nível, o tipo de fusão do sensor ?sl2 é de nível

Para que a regra tenha efeito é preciso acionar o motor de inferência, isto é o Pellet. A figura 32 mostra os resultados da aplicação da regra, aqui chamada de Regra 1.

The image displays a software interface for ontology management and querying. On the left, a class hierarchy is shown with 'SensorFusion' as the root class, containing subclasses: 'TemperatureSF', 'PressureSF', 'LevelSF', 'HumiditySF', and 'FlowSF'. Below this, a smaller hierarchy shows 'LevelSF', 'HumiditySF', and 'FlowSF'. On the right, there are two panels. The top panel, titled 'Instances', shows a list of instances for 'SensorFusion': 'tanque1' and 'tanqueA'. The bottom panel, titled 'DL query', shows a query for 'SensorFusion' with 'Execute' and 'Add to' buttons. Below the query, the 'Query results' section shows a list of 7 instances: 'sl1', 'sl2', 'sl3', 'slA', 'slB', 'tanque1', and 'tanqueA'.

Figura 32: Resultado da Regra 1: o lado superior esquerdo mostra os recursos com fusão, do mesmo lado embaixo, mostra os sensores de nível em fusão e no lado direito a consulta *DL Query* com todos os elementos participantes de fusão de sensores. Fonte: Elaborado pelo autor

É possível observar na figura 32 que os resultados apresentados pela aplicação da Regra 1 foram plenamente satisfatórios, visto que na implantação dos dados para a realização dos testes foram relacionados ao *tanque1* três sensores de nível (*sl1*, *sl2* e *sl3*) e ao *tanque2* dois sensores de nível (*slA* e *slB*). Esses são justamente os indivíduos mostrados nas três formas de consulta apresentadas.

A atribuição de quantidade diferentes de sensores para cada um dos tanques teve a intenção de verificar se a Regra 1 atendia aos requisitos de quantidade mínima, no caso dois sensores, ou acima do mínimo, em conjunto com o recurso, para serem considerados em fusão, resultado que também foi positivo como pode ser observado na figura 41. Logo após, foram testadas as condições que não contemplavam quantidade mínima de dois sensores em conjunto com um recurso. Então, primeiramente foi retirado um sensor do

tanque2, outra vez o resultado foi positivo, pois o Pellet não identificou a fusão no referido tanque e nem no único sensor que a ele ficou relacionado. Depois, retirou-se o *tanque1* e novamente o Pellet também não identificou mais fusão para os três sensores relacionados a esse tanque.

Com os resultados obtidos ficou então constatado que a Regra 1 atendeu plenamente à regra da fusão de sensores definida anteriormente.

- **Quais as linhas de produção que contém fusão de sensores?**

Esse questionamento é referente ao nível imediatamente superior, no contexto deste trabalho, aos níveis dos sensores e recursos. Trata-se do nível referente as linhas de produção, as quais têm em sua composição estes dois elementos.

A lógica da regra aplicada para atender a este questionamento é bastante simples, isto porque como já mencionado anteriormente, a Regra 1 serve como fundamentação para a lógica de sua construção, ou seja, essa regra só funciona se a regra anterior funcionar também. Ela ficou assim definida: se um recurso possui sensores em fusão e como ele pertence a uma linha de produção, conseqüentemente essa linha de produção tem elementos em fusão. Escrita em SWRL a Regra 2 ficou assim:

```
Resource(?r), SensorFusion(?r), ProductionLine(?pl),
hasresource(?pl, ?r) -> SensorFusion(?pl)
```

Significado da regra:

?r é um recurso, **?r** tem fusão de sensores, **?pl** é uma linha de produção, **?pl** tem recurso **?r** ENTÃO **?pl** tem fusão de sensores

Acionado o Pellet obteve-se as respostas exibidas na figura 33.

Na figura 33 observa-se que os resultados apresentados pela aplicação da Regra 2 foram plenamente satisfatórios, visto que na implantação dos dados para a realização dos testes o *tanque1* foi atribuído a linha de produção *smar1*, e o *tanque2* foi atribuído a linha de produção *smarA*, como ambos os tanques possuem sensores em fusão, conseqüentemente a linha de produção também contém fusão.

A validação da Regra 2, assim como a construção de sua lógica, também ficou facilitada, pois bastou excluir um dos indivíduos dos níveis mais abaixo da estrutura (recursos ou sensores) para o Pellet não identificar mais fusão na linha de produção a qual o indivíduo pertencia. Dessa forma, com os resultados obtidos ficou então constatado que a Regra 2 atendeu plenamente à condição de identificar as linhas de produção contendo

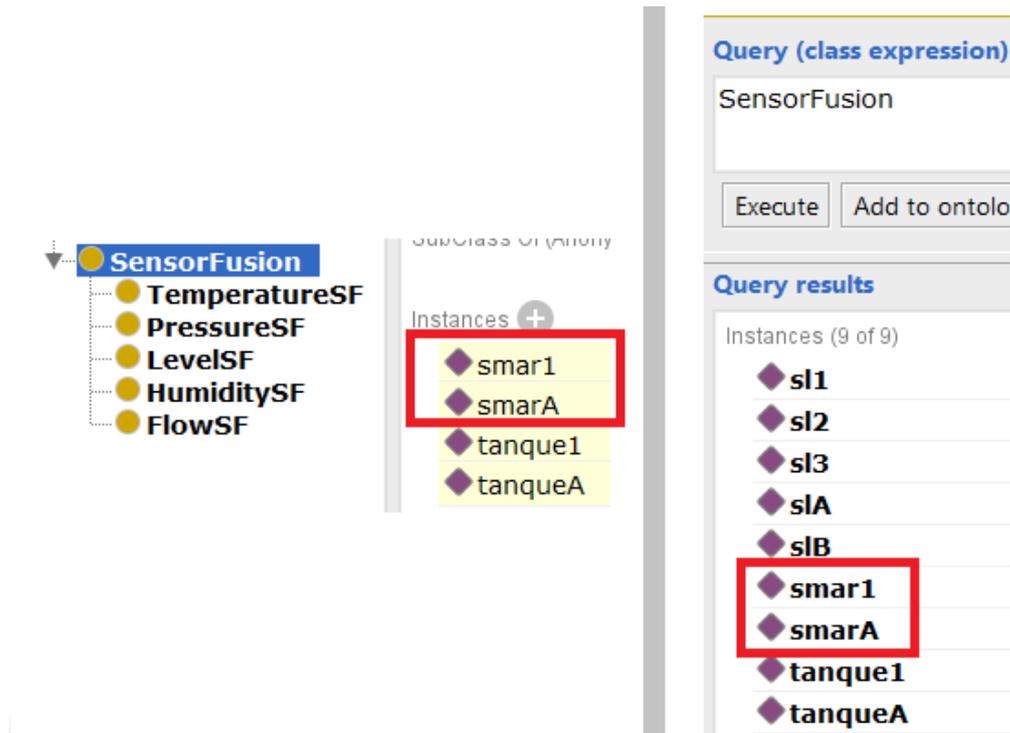


Figura 33: Resultado da Regra 2: o lado esquerdo mostra os recursos e, realçado em vermelho, as linhas de produção com fusão. No lado direito é mostrada a consulta *DL Query* com todos os elementos participantes de fusão de sensores, agora com as linhas de produção. Fonte: Elaborado pelo autor

fusão de sensores.

- **Quais as áreas que contém fusão de sensores?**

Subindo mais um nível na estrutura, agora o questionamento refere-se as áreas que contém fusão de sensores em suas instalações. A lógica da regra aplicada para atender a este questionamento é tão simples quanto a anterior e com a mesma lógica. Porém, ela é dependente das duas regras anteriores, sem elas não irá funcionar. Sua definição é: se uma área tem uma linha de produção e essa linha de produção tem fusão de sensores, consequentemente essa área tem elementos em fusão. Na SWRL a Regra 3 ficou assim:

```
Area(?a), ProductionLine(?pl), SensorFusion(?pl),
hasProductionLine(?a, ?pl) -> SensorFusion(?a)
```

Significado da regra:

?a é uma área, **?pl** é uma linha de produção, **?pl** tem fusão de sensores, **?a** tem linha de produção **?pl** ENTÃO **?a** tem fusão de sensores

Na figura 34 são exibidas as respostas à Regra 3 após o acionamento do Pellet.

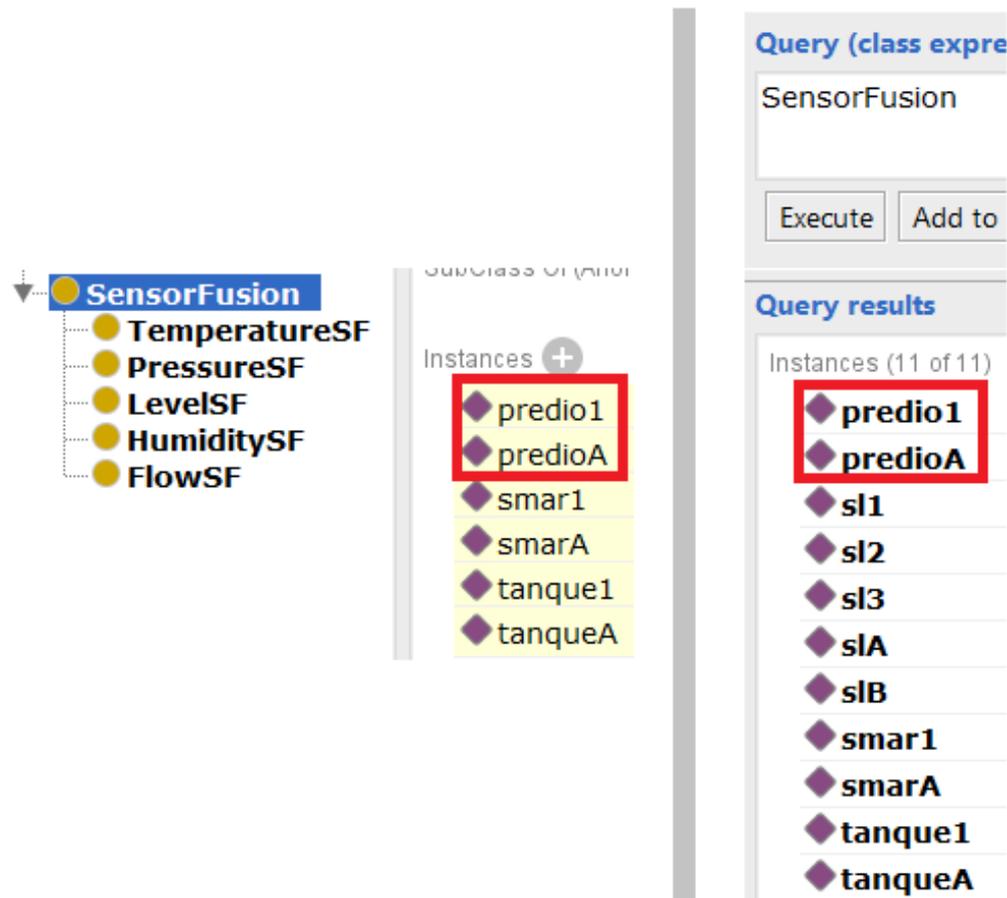


Figura 34: Resultado da Regra 3: o lado esquerdo mostra os recursos, as linhas de produção e, realçado em vermelho, as áreas com fusão. No lado direito é mostrada a consulta *DL Query* com todos os elementos participantes de fusão de sensores, incluindo as áreas. Fonte: Elaborado pelo autor

Com os resultados apresentados na figura 34, observa-se que a aplicação da Regra 3 atendeu as expectativas, pois na implantação dos dados para a realização dos testes a linha de produção *smar1* foi atribuída a área *predio1*, e a a linha de produção *smar2* foi atribuída a área *predioA*, como ambas as linhas de produção possuem elementos envolvidos com fusão de sensores, conseqüentemente a área onde as linhas de produção estão localizadas contém fusão.

Para validar a Regra 3, assim como na Regra 2, também foi feita a exclusão de um dos indivíduos dos níveis mais abaixo da estrutura da ontologia (linhas de produção, recursos e sensores) para o Pellet não identificar mais fusão na área onde estes elementos estavam relacionados. Aqui também os resultados obtidos demonstraram que a Regra 3 atendeu plenamente à condição de identificar as áreas relacionadas a fusão de sensores.

- Quais os sites que contém fusão de sensores?

Finalmente o último questionamento referente aos elementos físicos da ontologia. É o nível mais alto da estrutura representada e refere-se aos sites, mais especificamente as plantas industriais. Essa é outra regra dependente das demais regras apresentadas até aqui. Ela está definida assim: se um site tem uma área e essa área tem fusão de sensores, consequentemente esse site contém elementos em fusão. A regra 4 em SWRL ficou assim:

```
Site(?s), Area(?a), SensorFusion(?a), hasArea(?s, ?a) ->
SensorFusion(?s)
```

Significado da regra:

?s é um site, **?a** é uma área, **?a** tem fusão de sensores, **?s** tem área **?a** ENTÃO **?s** tem fusão de sensores

As respostas do Pellet à Regra 4 são exibidas na figura 35.

A aplicação da Regra 4 também teve sucesso, como pode ser observado na figura 35, pois na implantação dos dados para a realização dos testes a área *predio1* foi atribuída ao site *planta1*, e a área *predioA* foi atribuída ao site *plantaA*, como ambas as áreas possuem elementos envolvidos com fusão de sensores, consequentemente o site onde as áreas estão localizadas contém fusão.

A Regra 4, foi validada também com a exclusão de um dos indivíduos dos níveis mais abaixo da estrutura (áreas, linhas de produção, recursos e sensores) a fim de que o Pellet não identificasse mais fusão no site do qual fazem parte. O sucesso da Regra 4, atesta seu uso na identificação dos sites relacionados a fusão de sensores.

- **Há divergência de medição entre os sensores que estão em fusão em determinado recurso?**

A fim de atender a este questionamento foram elaboradas duas regras, as quais dependem da Regra 1. Uma delas é para identificar se as divergências de dados entre os sensores estão acima do limite de tolerância, e a outra para identificar se as divergências estão abaixo do limite de tolerância. A lógica destas duas regras é mais complexa que as anteriores.

Com relação ao funcionamento destas duas regras ficou estabelecido que para cada recurso que possuir sensores em fusão, deve ser determinada uma faixa de tolerância à variação dos valores nas medições realizadas pelos sensores. Sendo assim, para cada

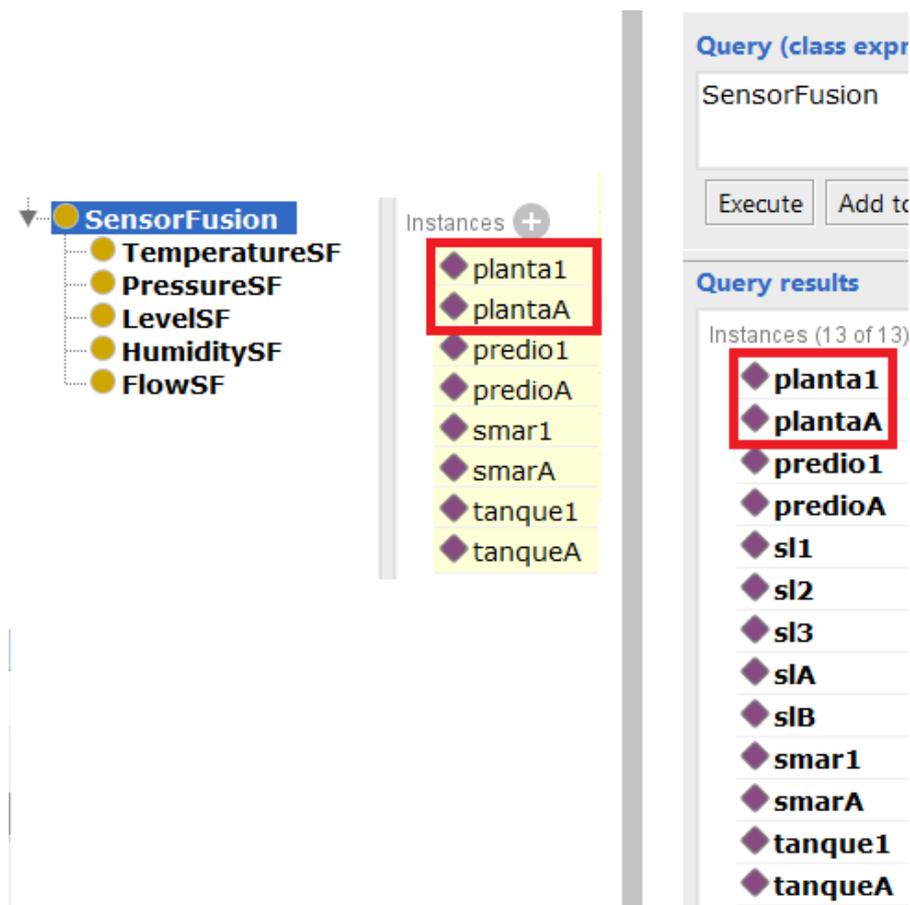


Figura 35: Resultado da Regra 4: o lado esquerdo mostra os recursos, as linhas de produção, as áreas e, realçado em vermelho, os sites com fusão de sensores. No lado direito é mostrada a consulta *DL Query* com todos os elementos participantes de fusão de sensores, incluindo os sites. Fonte: Elaborado pelo autor

tipo de fenômeno a ser medido pelos sensores, deve haver um parâmetro atribuído ao recurso que os contém. Por exemplo, se forem utilizados sensores de fluxo em fusão, o recurso onde os sensores atuam tem que estar relacionado a um parâmetro composto por dois valores, o valor máximo e o valor mínimo para tolerância as divergências nas medições. Se o mesmo recurso possuir outros sensores em fusão de qualquer outro tipo, deve ser estipulado outro parâmetro para o tipo específico utilizado. Para esta lógica ter efeito foi necessário primeiramente criar as propriedades de dados (*Data Properties*) para o armazenamento dos valores relativos aos parâmetros para as medições. Dessa forma foram criadas as propriedades mostradas na figura 26, uma representando o valor mínimo e a outra o valor máximo relativos a cada fenômeno medido.

Todas as propriedades de dados referente aos parâmetros foram definidas como funcionais, isso significa que cada uma delas só pode estar vinculada a um único parâmetro. O *domain* de cada uma delas é a classe de parâmetros do tipo de fenômeno medido. Exemplo, o domínio da propriedade de dados *paramMaxLevel* é a classe *ParameterLevel*. A *range* de todas as propriedades que estão relacionadas aos parâmetros é a mesma. É que

no caso das propriedades de dados a *range* serve para definir o tipo de dados suportado pela propriedade, e para todas elas o tipo definido foi *integer*. O motivo da escolha desse tipo de dado foi apenas ilustrativo, mais precisamente para facilitar a entrada de dados na execução dos testes, porém se a ontologia fosse aplicada em um sistema de controle real essas propriedades poderiam ser do tipo *real*.

Como pode ser observado na figura 27, além das propriedades de dados relacionadas aos parâmetros, mais uma propriedade foi definida. Trata-se da propriedade de dados *measuredValue*, a qual foi definida para armazenar o valor da medição do sensor, sendo que o seu *domain* é a classe **TypeSensor** e sua *range* também foi definida como *integer* pelas mesmas razões descritas no parágrafo anterior em relação as propriedades dos parâmetros.

A razão de definir que o *domain* da propriedade de dados *measuredValue* é a classe **TypeSensor**, é que essa classe é de mais alto nível relativa aos sensores dentro da ontologia. Ao fazer essa atribuição as suas subclasses também recebem essa propriedade de dados como herança. Ou seja, todos os sensores, independente do tipo, possuem um valor de medição.

Para identificar as divergência acima do limite definido a regra foi elaborada desta forma: um recurso tem fusão de sensores e determinado tipo de sensor, o sensor tem um valor medido, existe um valor máximo como parâmetro do recurso para as medições, se o valor medido for maior que o valor do parâmetro então existe uma divergência acima. Em SWRL a Regra 5 é esta:

```
Resource(?r), SensorFusion(?r), SensorLevel(?s),
hasSensorLevel(?r, ?s), measuredValue(?s, ?mv),
ParameterLevel(levelt1), hasParameterLevel(?r, levelt1),
paramMaxLevel(levelt1, ?pmax), swrlb:greaterThan(?mv,
?pmax) -> DivergenceUp(?s)
```

Significado da regra:

?r é um recurso, **?r** tem fusão de sensores, **?s** é um sensor de nível, o recurso **?r** tem um sensor de nível **?s**, o sensor **?s** tem um valor medido **?mv**, tem um parâmetro de nível **levelt1**, o recurso **?r** tem um parâmetro de nível **?levelt1**, o parâmetro de nível **?levelt1** tem um valor máximo **?pmax**, o valor medido é maior que o valor do parâmetro máximo ENTÃO **?s** tem divergência acima do limite máximo

Para verificar se o valor medido é maior que o valor do parâmetro máximo definido, é utilizada a função *swrlb:greaterThan* da biblioteca matemática da SWRL. O significado dessa função é o seguinte, se o valor da esquerda da função é maior que o valor da direita, ela retorna verdadeiro, caso contrário retorna falso.

Para poder testar a validade da Regra 5 foi preciso determinar valores de medição para os sensores e parâmetros. Então, para a realização dos testes foram atribuídos os seguintes valores:

- Sensores *tanque1*: *sl1* = 1, *sl2* = 70 e *sl3* = 40
- Parâmetros de nível *tanque1*: *paramMaxLevel* = 50 e *paraMinLevel* = 20

O resultado retornado após o acionamento do Pellet é mostrado na figura 36.

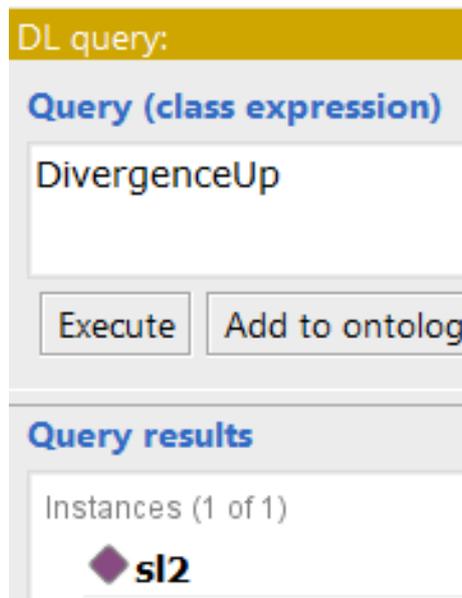


Figura 36: Resultado da Regra 5: a consulta *DL Query* mostra o sensor cujo valor da medição está acima do limite máximo permitido para o recurso onde ocorre a fusão de sensores. Fonte: Elaborado pelo autor

Com o resultado apresentado na figura 36, ficou evidenciado que a Regra 5 mostrou sua eficiência na identificação de divergências de medições de sensores acima do limite permitido, já que no teste o único sensor com valor de medição acima do limite foi identificado pelo Pellet. Para validar essa regra somente foi necessário fazer alteração dos valores relativos aos dados utilizados nos testes.

Com relação a identificação das divergência abaixo do limite definido a regra foi elaborada desta forma: um recurso tem fusão de sensores e determinado tipo de sensor, o sensor tem um valor medido, existe um valor mínimo como parâmetro do recurso para as medições, se o valor medido for menor que o valor do parâmetro então existe uma divergência abaixo. Em SWRL a Regra 6 é esta:

```
Resource(?r), SensorFusion(?r), SensorLevel(?s),
hasSensorLevel(?r, ?s), measuredValue(?s, ?mv),
ParameterLevel(levelt1), hasParameterLevel(?r, levelt1),
paramMaxLevel(levelt1, ?pmax), swrlb:lessThan(?mv, ?pmax)
```

-> DivergenceDown(?s)

Significado da regra:

?r é um recurso, **?r** tem fusão de sensores, **?s** é um sensor de nível, o recurso **?r** tem um sensor de nível **?s**, o sensor **?s** tem um valor medido **?mv**, tem um parâmetro de nível **levelt1**, o recurso **?r** tem um parâmetro de nível **?levelt1**, o parâmetro de nível **?levelt1** tem um valor mínimo **?pmin**, o valor medido é menor que o valor do parâmetro mínimo ENTÃO **?s** tem divergência abaixo do limite mínimo

O teste desta regra foi feito com os mesmos valores definidos no teste anterior, o resultado retornado pelo Pellet é mostrado na figura 37.

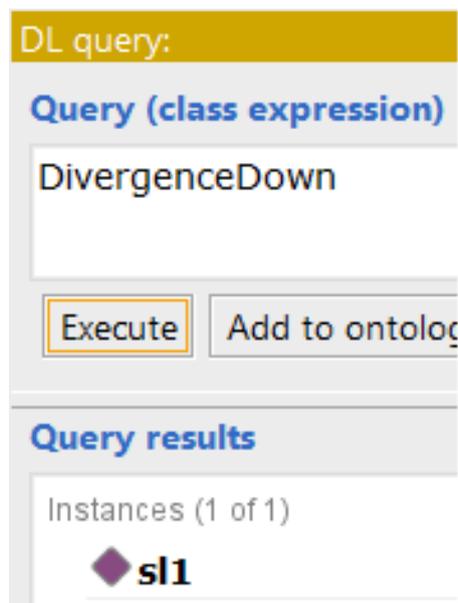


Figura 37: Resultado da Regra 6: a consulta *DL Query* mostra o sensor cujo valor da medição está abaixo do limite mínimo permitido para o recurso onde ocorre a fusão de sensores. Fonte: Elaborado pelo autor

O resultado apresentado na figura 37 atesta também que a Regra 6 identificou o sensor cuja a divergência de medição é abaixo do limite mínimo definido no parâmetro do recurso que contém os sensores. Dessa forma, todas as regras implementadas obtiveram sucesso naquilo a que se propunham, ou seja, possibilitar a identificação da fusão de sensores dentro do ambiente fabril, permitindo não só a identificação das instalações e equipamentos envolvidos em tal processo, como também a ocorrência de divergência de dados nas medições dos sensores utilizados nessas ocasiões.

Consultas com SPARQL

A seguir serão apresentadas algumas consultas aos dados da ontologia que foram construídas através da utilização da linguagem definida para esse fim, a SPARQL. Para as consultas feitas com a SPARQL é desnecessário acionar o Pellet, pois o motor de inferência não interfere em nada, diferentemente das regras SWRL que somente funcionam com ele.

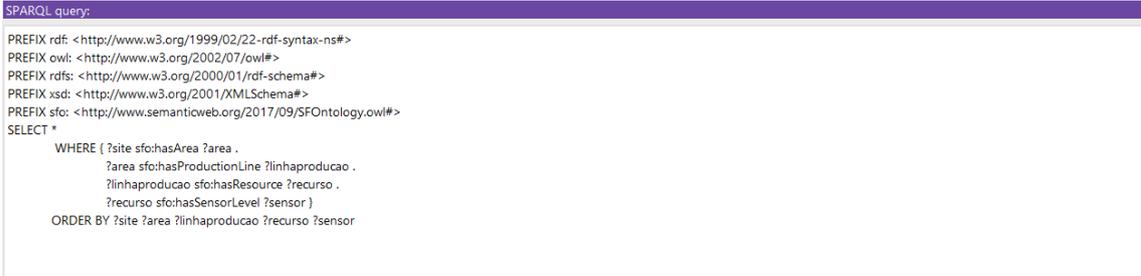
- Retorna todos os sites, suas áreas, suas linhas de produção, recursos e sensores

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX sfo: <http://www.semanticweb.org/2017/09/SFOntology.owl#>
SELECT *
WHERE ?site sfo:hasArea ?area .
      ?area sfo:hasProductionLine ?linhaproducao .
      ?linhaproducao sfo:hasResource ?recurso .
ORDER BY ?site ?area ?linhaproducao ?recurso ?sensor

```

A figura 38 mostra os resultados retornados com a Consulta 1.



SPARQL query:

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX sfo: <http://www.semanticweb.org/2017/09/SFOntology.owl#>
SELECT *
WHERE { ?site sfo:hasArea ?area .
        ?area sfo:hasProductionLine ?linhaproducao .
        ?linhaproducao sfo:hasResource ?recurso .
        ?recurso sfo:hasSensorLevel ?sensor }
ORDER BY ?site ?area ?linhaproducao ?recurso ?sensor

```

site	area	linhaproducao	recurso	sensor
planta1	predio1	smar1	tanque1	sl1
planta1	predio1	smar1	tanque1	sl2
planta1	predio1	smar1	tanque1	sl3
plantaA	predioA	smarA	tanqueA	sIA
plantaA	predioA	smarA	tanqueA	sIB

Figura 38: Resultado da Consulta SPARQL 1. Fonte: Elaborado pelo autor

- Retorna os recursos e seus sensores

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX sfo: <http://www.semanticweb.org/2017/09/SFOntology.owl#>

```

```

SELECT *
WHERE ?area sfo:hasProductionLine ?linhaproducao .
?linhaproducao sfo:hasResource ?recurso .
ORDER BY ?site ?area ?linhaproducao ?recurso ?sensor

```

A figura 39 mostra os resultados retornados com a Consulta 2.

SPARQL query:	
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX owl: <http://www.w3.org/2002/07/owl#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> PREFIX sfo: <http://www.semanticweb.org/2017/09/SFOntology.owl#> SELECT * WHERE { ?recurso sfo:hasSensorLevel ?sensor }	
recurso	sensor
tanque1	sl3
tanque1	sl2
tanqueA	slB
tanque1	sl1
tanqueA	slA

Figura 39: Resultado da Consulta SPARQL 2. Fonte: Elaborado pelo autor

- Retorna todos os sensores com suas respectivas medições

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX sfo: <http://www.semanticweb.org/2017/09/SFOntology.owl#>
SELECT ?sensor ?mv
WHERE ?sensor rdf:type sfo:SensorLevel .
?sensor sfo:measuredValue ?mv

```

A figura 40 mostra os resultados retornados com a Consulta 3.

Importante ressaltar que para as consultas SPARQL retornarem todos os indivíduos de uma classe é necessário que o tipo (classe a qual pertencem os indivíduos) esteja parametrizado na janela *Description* da aba *Individuals by class*.

Nesta seção foram abordados todos os aspectos ligados ao desenvolvimento da ontologia proposta, bem como sobre o tratamento dado as informações visando a extração do conhecimento sobre o domínio abordado, no caso o uso de fusão de sensores na manufatura. Foram descritos também os testes de validação para assegurar que a lógica envolvida

SPARQL query:	
<pre> PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX owl: <http://www.w3.org/2002/07/owl#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> PREFIX sfo: <http://www.semanticweb.org/2017/09/SFOntology.owl#> SELECT ?sensor ?mv WHERE { ?sensor rdfs:type sfo:SensorLevel . ?sensor sfo:measuredValue ?mv } </pre>	
sensor	mv
sl1	"1"^^<http://www.w3.org/2001/XMLSchema#integer>
sl2	"70"^^<http://www.w3.org/2001/XMLSchema#integer>
sl3	"40"^^<http://www.w3.org/2001/XMLSchema#integer>

Figura 40: Resultado da Consulta SPARQL 3. Fonte: Elaborado pelo autor

no processo está construída de forma correta para obtenção de resultados seguros. Pode-se observar que o uso de regras SWRL possibilita uma maneira segura para a aplicação de restrições em uma ontologia.

5 CONCLUSÃO

Neste trabalho de pesquisa foi proposta e desenvolvida uma ontologia para manufatura a fim de servir de suporte à tomada de decisão em sistemas de controle de plantas industriais, mais especificamente no que diz respeito a aplicação de fusão de sensores em linhas de produção. Para isso, foi levada em consideração a revisão bibliográfica dos conceitos envolvidos nos dois eixos principais da proposta, o uso de ontologias e de fusão de sensores.

Dessa forma, a partir dos estudos realizados foi adotada a metodologia Methontology para apoiar a construção da ontologia, a qual garantiu que todos os aspectos relevantes relacionados ao domínio abordado fossem descritos nela e que todas as etapas necessárias ao seu desenvolvimento fossem cumpridas de forma organizada.

Outro aspecto envolvendo as etapas de desenvolvimento da ontologia foi a definição da ferramenta Protégé, a qual mostrou ser bastante robusta, oferecendo diferentes opções de construção da lógica, de visualização da estrutura da ontologia e também suporte a possíveis ocorrências de erros em sua estrutura. Além disso, ele também suporta a linguagem OWL-DL que foi escolhida para a implementação da ontologia, na qual os relacionamentos entre sujeito, predicado e objeto, tornaram o entendimento dos conceitos mais precisos e organizados dentro do domínio da ontologia. No entanto, algumas características observadas na utilização do Protégé devem ser levadas em consideração pois são relevantes se a ontologia for integrada a um sistema real de controle industrial.

A primeira delas é a necessidade de definir alguns parâmetros no Protégé de acordo com a operação realizada. Por exemplo, as inferências realizadas pelo Pellet não necessitam de definição de parâmetros (ex: informar a qual classe determinado indivíduo pertence), apenas com base nas descrições dos indivíduos e de suas relações ele faz sua deduções. Já no caso das consultas com SPARQL é preciso fazer tal parametrização, ou seja, para consultar os indivíduos pertencentes a uma determinada classe é necessário antes declarar que eles pertencem a classe. Essa característica é devido ao fato de que nas consultas com SPARQL o motor de inferência não tem relevância nenhuma. Os dados acessados nessas consultas são dados fixos que estão diretamente gravados no arquivo XML da ontologia, diferentemente dos dados inferidos pelo motor de inferência, os quais

são exibidos na interface da ferramenta, só existindo logicamente. Justamente o fato das instâncias ficarem armazenadas no próprio código da ontologia é outra característica que desenvolvedores de sistemas precisam estar atentos pois o tamanho e a complexidade do código podem compromete-lo.

Com relação a fusão de sensores, as pesquisas permitiram observar que existem inúmeros trabalhos envolvendo a questão das diferenças nos dados dos sensores envolvidos, porém em todos eles as abordagens referem-se a soluções matemáticas complexas, tanto para identificação e classificação destas diferenças, por exemplo como a definida por [47] e exibida na figura 4, como para resolução delas. Esses algoritmos de solução envolvendo fórmulas matemáticas complexas são de difícil implantação com a SWRL. Isso fez com que neste trabalho fosse adotado o termo **Divergência** para a identificação destes casos, já que a proposta da ontologia não é a de resolver estas situações, apenas identifica-las, apontar onde estão ocorrendo e os elementos envolvidos.

É importante ressaltar que a revisão bibliográfica sobre as tecnologias semânticas permitiu observar a importância do uso de ontologias e suas tecnologias para a representação do conhecimento, ainda mais quando utilizadas pelos motores de inferência em conjunto com as restrições construídas em SWRL.

Justamente a combinação da OWL-DL, utilizada na construção da ontologia proposta, com as restrições SWRL e o Pellet, permitiu que o objetivo principal dessa pesquisa fosse alcançado. Visto que na medida em que a fundamentação teórica, a aplicação da metodologia, o desenvolvimento da ontologia e a aplicação das restrições trouxeram os resultados esperados, como mostrado na validação da proposta. Sendo assim, foi possível a identificação e classificação dos componentes envolvidos na aplicação da fusão de sensores, a descrição de cada um deles e a identificação das divergências de dados entre sensores utilizados em fusão em plantas industriais, provando assim que a ontologia possui condições para dar suporte a sistemas de controle industriais.

5.1 Trabalhos Futuros

Durante a elaboração desse trabalho foram constatadas as seguintes possibilidades para trabalhos futuros:

- solução para classificação das ocorrências de diferenças de dados entre sensores em fusão (podendo servir como base a taxonomia apresentada por [47]), seja através de descrição lógica ou de restrições SWRL;

- integração da ontologia aqui apresentada com aquela apresentada pelo CIFASIS, a fim de permitir o compartilhamento do módulo de supervisão que dela faz parte, o qual possui características interessantes com respeito aos tratamentos dos eventos que ocorrem em uma planta industrial, os quais estão muitas vezes relacionados aos sensores. Ela permite a identificação de mal funcionamento de equipamentos, apontando os riscos e

suas consequências, entre outras características.

REFERÊNCIAS

- [1] *Michaelis Dicionário Brasileiro da Língua Portuguesa*. Editora Melhoramentos, São Paulo, 2015. Disponível em: <http://michaelis.uol.com.br/moderno/portugues/index.php>.
- [2] A. R. Abba Lawan and N. Alechina. Semantic web rule language (swrl) expressiveness extensions. *Semantic Web 1*, 2016.
- [3] W. A. Abdulhafiz and A. Khamis. Handling data uncertainty and inconsistency using multisensor data fusion. *Advances in Artificial Intelligence*, 2013:11, 2013.
- [4] L. Abele, C. Legat, S. Grimm, and A. W. Müller. Ontology-based validation of plant models. In *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, pages 236–241, 2013.
- [5] T. Aicher, D. Schütz, and B. Vogel-Heuser. Consistent engineering information model for mechatronic components in production automation engineering. In *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, pages 2532–2537, 2014.
- [6] M. T. T. Andrade, C. V. Ferreira, and H. B. d. B. Pereira. Uma ontologia para a gestão do conhecimento no processo de desenvolvimento de produto. *Revista Gestão & Produção*, 2010.
- [7] T. B. Armando W. Colombo, Stamatis Karnouskos. *Towards the Next Generation of Industrial Cyber-Physical Systems*. Springer International Publishing AG, 2014.
- [8] F. Baader. *The description logic handbook: Theory, implementation and applications*. Cambridge University Press, 2003.
- [9] R. Baheti and H. Gill. Cyber-physical systems. *The Impact of Control Technology*, pages 161–166, 2011.
- [10] F. Bendik and N. Schmidt. Exchange of engineering data for communication systems based on automationml using an ethernet/ip example. In *ODVA Industry Conference and 17th Annual Meeting, Friso, Texas, USA*, 2015.

- [11] S. Biffl, A. Schatten, and A. Zoitl. Integration of heterogeneous engineering environments for the automation systems lifecycle. IEEE Computer Society, 2009. In: IEEE International Conference on Industrial Informatics.
- [12] S. Borgo and P. Leitão. Foundations for a core ontology of manufacturing. In: *Kishore, R., Ramesh, R., Sharman, R. (eds.) Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems*, 2006.
- [13] W. N. Borst. Construction of engineering ontologies for knowledge sharing and reuse, 1997. CTIT series No. 97-14.
- [14] A. Buchsbaum and T. Pequeno. O método dos tableaux generalizado e sua aplicação ao raciocínio automático em lógicas não clássicas. *Revista ?O que nos faz pensar?, Cadernos do Departamento de Filosofia da PUC-Rio*, 3(81-96):18, 1990.
- [15] H. Cheng, P. Zeng, L. Xue, P. W. Zhao Shi, and H. Yu. Manufacturing ontology development based on industry 4.0 demonstration production line. IEEE Computer Society, 2016. In: Third International Conference on Trustworthy Systems and their Applications.
- [16] C.-M. Chituc and F. J. de Oliveira Restivo. Challenges and trends in distributed manufacturing systems: Are wise engineering systems the ultimate answer? In *Second MIT International Symposium on Engineering Systems*, 2009.
- [17] M. Compton, P. Barnaghi, L. Bermudez, R. García-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, K. Janowicz, W. D. Kelsey, D. L. Phuoc, L. Lefort, M. Leggieri, H. Neuhaus, A. Nikolov, K. Page, A. Passant, A. Sheth, and K. Taylor. The ssn ontology of the w3c semantic sensor network incubator group. *Web Semantics: Science, Services and Agents on the World Wide Web*, 17(Supplement C):25–32, 2012.
- [18] A. consortium. Automationml, 2006.
- [19] C. N. da Indústria. Desafios para indústria 4.0 no brasil. Technical report, 2016. Conselho Temático Permanente de Política Industrial e Desenvolvimento Tecnológico - COPIN - Brasília, Brasil.
- [20] D. L. da Silva, R. R. Souza, and M. B. Almeida. Ontologias e vocabulários controlados: comparação de metodologias para construção. *Ciência da Informação*, 37(3), 2008.
- [21] O. F. N. de Araújo. *Proposta de uma rede de compartilhamento de habilidades no ambiente da Manufatura*. PhD thesis, Universidade Estadual de Campinas, 2003. Tese de Doutorado em Engenharia Mecânica.

- [22] J. Dong, D. Zhuang, Y. Huang, and J. Fu. Advances in multi-sensor data fusion: Algorithms and applications. *Sensors*, 9(10):7771–7784, 2009.
- [23] P. R. dos Santos. Indústria 4.0 - sistemas inteligentes para manufatura do futuro. 2016. Acesso em: 20 jul. 2017.
- [24] J. M. dos Santos Pinheiro. Web semântica: Uma rede de conceitos. *Cadernos UniFOA*, 4(9):23–27, 2009.
- [25] B. DuCharme. *Learning SPARQL*. O'Reilly Media, Inc, 2011.
- [26] W. Elmenreich. *Sensor Fusion in Time-Triggered Systems*. PhD thesis, Technischen Universität Wien, Fakultät für Technische Naturwissenschaften und Informatik, 2002. Tese de Doutorado em Ciência Técnicas.
- [27] P. A. H. Enzo Morosini Frazzon, Lucas de Souza Silva and H. F. Borges. Aplicação do conceito de cyber-physical systems em manufacturing execution systems. *XXXV Encontro Nacional de Engenharia de Produção, Fortaleza-CE*, 2015.
- [28] J. C. Fonseca. *Aplicação de técnicas de fusão/integração sensorial de dados no levantamento do relevo de objectos*. PhD thesis, Universidade do Minho - Portugal, 1999. Tese de Doutorado em Engenharia Electrónica Industrial.
- [29] A. Frikha and H. Moalla. Analytic hierarchy process for multi-sensor data fusion based on belief function theory. *European Journal of Operational Research*, 241(1):133–147, 2015.
- [30] M. Garetti, L. Fumagalli, and E. Negri. Role of ontologies for cps implementation in manufacturing. *Management and Production Engineering Review*, 6(4):26, 2015.
- [31] P. Gerbert, M. Lorenz, M. Rübmann, M. Waldner, J. Justus, P. Engel, and M. Harnisch. Industry 4.0: The future of productivity and growth in manufacturing industries. 2015. Acesso em: 10 jul. 2017.
- [32] A. D. Gomez. Organización de la información mediante el uso de lenguajes de modelado: viejos recursos para nuevas necesidades. *Revista Scire: Representación y Organización del Conocimiento*, Vol. 8, No. 1, 2002.
- [33] M. Grüninger and M. S. Fox. Methodology for the design and evaluation of ontologies. in: *Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal*, 1995.
- [34] M. Groover. *Automação Industrial e Sistemas de Manufatura*. Pearson Prentice Hall, 2011.

- [35] W. O. W. Group. *OWL 2 Web Ontology Language Document Overview (Second Edition)*, 2012.
- [36] W. R. W. Group. *Resource Description Framework*, 2014.
- [37] N. Guarino. Formal ontology and information systems. *National Research Council*, 1998. Proceedings of FOIS'98, Trento, Italy, 6-8 June.
- [38] R. M. V. Gutierrez and S. S. K. Pan. Complexo eletrônico: Automação do controle industrial. In *Revista BNDES Setorial*, number 28, pages 189–232. Banco Nacional de Desenvolvimento Econômico e Social, 2008. Acesso em: 05 de jul. 2017.
- [39] M. Hermann, T. Pentek, and B. Otto. Design principles for industrie 4.0 scenarios: A literature review. *Manufacturing Letters*, 2015.
- [40] S. Iarovyi, W. M. Mohammed, A. Lobov, B. R. Ferrer, and J. L. M. Lastra. Cyber - physical systems for open-knowledge-driven manufacturing execution systems. *Proceedings of the IEEE*, 104(5):1142–1154, 2016.
- [41] ISA. *ANSI/ISA-95.00.01-2010, Enterprise-Control System Integration - Part 1: Models and Terminology*, 2010.
- [42] ISA. *ANSI/ISA-S88.00.01-2010, Enterprise-Control System Integration - Batch Control Part 1: Models and Terminology*, 2010.
- [43] S. Isotani and I. I. Bittencourt. *Dados Abertos Conectados*. Editora Novatec, 2015.
- [44] A. P. John G. Breslin, David O'Sullivan and L. Vasiliu. Semantic web computing industry. *Computers in Industry*, 61(8):729–741, 2010.
- [45] E. A. José Barbosa, Paulo Leitão and D. Trentesaux. Self-organized holonic multi-agent manufacturing system: the behavioural perspective. *IEEE International Conference on Systems, Man, and Cybernetics*, 2013.
- [46] R. Joshi and A. C. Sanderson. *Multisensor fusion: A minimal representation framework*, volume 11. World Scientific, Singapore, 1999.
- [47] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi. Multisensor data fusion: A review of the state-of-art. *Journal Information Fusion*, 14(1):28–44, 2011.
- [48] O. Kovalenko, M. Wimmer, M. Sabou, A. Lüder, F. J. Ekaputra, and S. Biffli. Modeling automationml: Semantic web technologies vs. model-driven engineering. In *2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA)*, pages 1–4, 2015.

- [49] M. Kumar, D. P. Garg, and R. A. Zachery. A method for judicious fusion of inconsistent multiple sensor data. *IEEE Sensors Journal*, 7(5):723–733, 2007.
- [50] J. L. M. Lastra and I. M. Delamer. Semantic web services in factory automation: Fundamental insights and research roadmap. *IEEE Transactions on Industrial Informatics*, Vol. 2, No. 1, February 2006, 2006.
- [51] J. L. M. Lastra and I. M. Delamer. Ontologies for production automation. In *Advances in Web Semantic I*. Ed. Springer, 2009. LNCS 4891, pp. 276 a 289, EUA.
- [52] P. J. P. Leitão. *An agile and adaptive holonic architecture for manufacturing control*. PhD thesis, Faculdade de Engenharia da Universidade do Porto, 2004.
- [53] S. Lemaignan, A. Siadat, J. Y. Dantan, and A. Semenenko. Mason: A proposal for an ontology of manufacturing domain. In *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, pages 195–200, 2006.
- [54] M. E. Leusin, M. U. Maldonado, and E. M. Frazzon. Mapeamento da produção científica sobre os sistemas ciber-físicos (cps) com foco na troca de dados. 2017. XX Simpósio de Administração da Produção, Logística e Operações Internacionais. São Paulo.
- [55] R. G. F. Lima. *Modelagem Conceitual de Processos Industriais com Aplicações*. PhD thesis, Universidade Federal do Rio Grande do Norte, 2014. Tese de Doutorado em Engenharia Elétrica e de Computação.
- [56] J. A. G. Luna, M. L. Bonilla, and I. D. Torres. Metodologías y métodos para la construcción de ontologías. *Scientia et technica*, (50):133–140, 2012.
- [57] R. C. Luo and C. C. Chang. Multisensor fusion and integration aspects of mechatronics. *IEEE Industrial Electronics Magazine*, 4(2):20–27, 2010.
- [58] R. C. Luo and C. C. Chang. Multisensor fusion and integration: A review on approaches and its applications in mechatronics. *IEEE Transactions on Industrial Informatics*, 8(1):49–60, 2012.
- [59] R. C. Luo and M. G. Kay. A tutorial on multisensor integration and fusion. In *Industrial Electronics Society, 1990. IECON '90., 16th Annual Conference of IEEE*, pages 707–722 vol.1, 1990.
- [60] R. C. Luo, C.-C. Yih, and K. L. Su. Multisensor fusion and integration: approaches, applications, and future research directions. *IEEE Sensors Journal*, 2(2):107–119, 2002.

- [61] A. G.-P. Mariano Fernández and N. Juristo. Methontology: From ontological art towards ontological engineering. *in: AAAI-97 Spring Symposium Series*, 1997.
- [62] F. S. Markus Krötzsch and I. Horrocks. A description logic primer. 2012.
- [63] C. J. Matheus, D. Tribble, M. M. Kokar, M. G. Ceruti, and S. C. McGirr. Towards a formal pedigree ontology for level-one sensor fusion. 2006. 10th International Command & Control Research and Technology Symposium The Future of C2: Information Operations/Assurance.
- [64] E. Musulin, F. Roda, and M. Basualdo. A knowledge-driven approach for process supervision in chemical plants. *Computers and Chemical Engineering*, (59):164–177, 2013.
- [65] E. F. Nakamura, A. A. F. Loureiro, and A. C. Frery. Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Comput. Surv.*, 39(3), 2007.
- [66] P. Novák. *Design and Integration of Simulation Models for Industrial Systems*. PhD thesis, Czech Technical University in Prague, 2016. Doctoral Thesis in Electrical Engineering and Information Technology.
- [67] N. F. Noy and D. L. McGuinness. Ontology development 101: A guide to creating your first ontology. *Report Ksl-01-05 And Stanford Medical Informatics Technical Report Smi-2001-0880*, 2001.
- [68] M. O’connor, H. Knublauch, S. Tu, B. Grosz, M. Dean, W. Grosso, and M. Musen. Supporting rule system interoperability on the semantic web with swrl. *The Semantic Web–ISWC 2005*, pages 974–986, 2005.
- [69] F. of the future project team in the IEC Market Strategy Board. Factory of the future. Technical report, International Electrotechnical Commission, 2015.
- [70] C. Pillajo and J. E. Sierra. Importancia del estudio del control para los sistemas cyber-físicos. *Universidad Politécnica Salesiana*, 2014.
- [71] D. Plinere and A. Borisov. Swrl: Rule acquisition using ontology. *Scientific Journal of Riga Technical University. Computer Sciences*, 40(1):117–122, 2009.
- [72] J. Posada, C. Toro, I. Barandiaran, D. Oyarzun, D. Stricker, R. de Amicis, E. B. Pinto, P. Eisert, J. Döllner, and I. Vallarino. Visual computing as key enabling technology for industrie 4.0 and industrial internet. *Browse Journals & Magazines*, 2015.

- [73] J. Pérez, M. Arenas, and C. Gutierrez. *Semantics and Complexity of SPARQL*, pages 30–43. Springer Berlin Heidelberg, 2006.
- [74] S. Rawat and S. Rawat. Multi-sensor data fusion by a hybrid methodology ? a comparative study. *Computers in Industry*, 75:27–34, 2016.
- [75] M. B. A. e. R. M. A. B. Renato Rocha Souza. Ciência da informação em transformação: Big data, nuvens, redes sociais e web semântica. *Revista Ciência da Informação*, 44(2), 2015.
- [76] M. A. Ribeiro. *Automação Industrial*. Tek treinamento & Consultoria Ltda, 2011. 4ª edição.
- [77] A. Rivolli and D. A. Moreira. Aprimorando a visualização e composição de regras swrl na web. *Scientific Journal of Riga Technical University. Computer Sciences*, 2014.
- [78] A. Rivolli, J. P. Orlando, C. H. Yamamoto, and D. Moreira. Regras swrl: Análise de similaridade e detecção de erros. In *WebMedia*, 11, 2011. WebMedia’11: Proceedings of the 17 Brazilian Symposium on Multimedia and the Web.
- [79] F. Roda. *Supervisión de Procesos Basada en Modelado Semántico*. PhD thesis, Facultad de Ciencias Exactas, Ingeniería y Agrimensura, 2015.
- [80] J. M. Rosário. *Automação Industrial*. Editora Baraúna, São Paulo, 2009.
- [81] P. L. Rothman and R. V. Denton. Fusion or confusion: knowledge or nonsense?, 1991.
- [82] M. Sabou, F. Ekaputra, O. Kovalenko, and S. Biffl. Supporting the engineering of cyber-physical production systems with the automationml analyzer. In *2016 1st International Workshop on Cyber-Physical Production Systems (CPPS)*, pages 1–8, 2016.
- [83] M. Schleipen, D. Gutting, and F. Sauerwein. Domain dependant matching of mes knowledge and domain independent mapping of automationml models. In *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies Factory Automation (ETFA 2012)*, pages 1–7, 2012.
- [84] C. Schlenoff, T. Hong, C. Liu, R. Eastman, and S. Foufou. A literature review of sensor ontologies for manufacturing applications. In *2013 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*, pages 96–101, 2013.
- [85] J. H. I. H. D. L. M. P. F. P.-S. Sean Bechhofer, Frank van Harmelen and L. A. Stein. *OWL Web Ontology Language Reference*, 2004.

- [86] R. Seleme and R. B. Seleme. *Automação da Produção: uma abordagem gerencial*. Editora Intersaberes, 2013.
- [87] E. N. Silva and C. Bezerra. Uso de ontologias e swrl para ensino personalizado baseado em estilos e teorias de aprendizagem. In *Nuevas Ideas en Informática Educativa - Memorias del XVIII Congreso Internacional de Informática Educativa, TISE 2013*.
- [88] M. S. Sonzini and M. Vegetti. Ontología para la gestión unificada de variantes y versiones de productos. In *I Simposio Argentino de Ontologías y sus Aplicaciones (SAOA)-JAIIO 44 (Rosario, 2015)*, 2015.
- [89] S. Strzelczak. Towards ontology-aided manufacturing and supply chain management - a literature review. In *Advances in Production Management Systems: Innovative Production Management Towards Sustainable Growth*. Shigeki Umeda, and Masaru Nakano, and Hajime Mizuyama, and Hironori Hibino, and Dimitris Kiritsis and Gregor von Cieminski, 2015. IFIP International Conference on Advances in Production Management Systems, pp. 467-474.
- [90] J. H. Tim Berners-Lee and O. Lassila. The semantic web. *Scientific american*, 284(5):28–37, 2001.
- [91] C. Toro, I. Barandiaran, and J. Posada. A perspective on knowledge based and intelligent systems implementation in industrie 4.0. *Procedia Computer Science*, 60:362–370, 2015. Knowledge-Based and Intelligent Information and Engineering Systems 19th Annual Conference, KES-2015, Singapore, September 2015 Proceedings.
- [92] M. Uschold and M. King. Towards a methodology for building ontologies. *IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal, Canada*, 1995.
- [93] C.-H. Yang, V. Vyatkin, and C. Pang. Model-driven development of control software for distributed automation: A survey and an approach. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.
- [94] S. Zander and Y. Hua. Utilizing ontological classification systems and reasoning for cyber-physical systems. In *Karlsruhe Service Summit Research Workshop, February*, 2016.
- [95] J. Zhou, T. Leppänen, M. Liu, E. Harjula, T. Ojala, M. Ylianttila, and C. Yu. Sensor information representation for the internet of things. In *On The Move to Meaningful Internet Systems: OTM 2012 Workshops*, pages 515–524, 2012.

ANEXO A GRÁFICO DA ONTOLOGIA



Figura 41: Gráfico da ontologia. Fonte: Elaborado pelo autor