

Universidade Metodista de Piracicaba

Programa de Pós-Graduação em Engenharia de Produção

***FRAMEWORK PARA DESENVOLVIMENTO DE UM
MASTER PROTOTYPE DO PRODUTO***

Karine Melissa de Queiroz Rigon

Orientador: Prof. Dr.-Ing. Klaus Schützer

Santa Bárbara d'Oeste
Dezembro 2020

Ficha Catalográfica elaborada pelo Sistema de Bibliotecas da UNIMEP
Bibliotecário: Fábio Henrique dos Santos Corrêa – CRB: 8/10150

R572f Rigon, Karine Melissa de Queiroz
Framework para desenvolvimento de um Master Prototype do
produto / Karine Melissa de Queiroz Rigon. – 2020.
115 fls.; il.; 30 cm.

Orientador (a): Prof. Dr. Klaus Schützer.
Dissertação (Mestrado) – Universidade Metodista de Piracicaba,
Programa de Pós-Graduação em Engenharia de Produção, Santa
Bárbara D'Oeste, 2020.

1. Desenvolvimento de produtos. 2. Indústria 4.0. 3. Master
Prototype. 4. Gêmeo Digital. I. Schützer, Klaus. II. Título.

CDD – 670

Universidade Metodista de Piracicaba

Programa de Pós-Graduação em Engenharia de Produção

***FRAMEWORK PARA DESENVOLVIMENTO DE UM
MASTER PROTOTYPE DO PRODUTO***

Karine Melissa de Queiroz Rigon

Orientador: Prof. Dr.-Ing. Klaus Schützer

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção, da Universidade Metodista de Piracicaba – UNIMEP, como requisito para a obtenção do título de Mestre em Engenharia de Produção.

Santa Bárbara d'Oeste
Dezembro 2020

Agradecimentos

Em primeiro lugar, agradeço a Deus por ter me levado a essa conquista e todo o aprendizado que me permitiu vivenciar ao longo desse processo. Sem Ele nada disso seria possível.

Agradeço a minha família, meu pai Joelso, minha mãe Divonete e minha irmã Anne, que sempre me apoiaram na busca pelo conhecimento, isso desde os meus mais tenros anos.

Sou extremamente grata ao professor Klaus pela confiança, dedicação e aprendizado que me ofereceu.

Sinto uma enorme gratidão pelos meus colegas de laboratório, Julia, Carolina, Nathalia, Felipe, Marcelo e Bruno, pelo companheirismo, disponibilidade e grande amizade. Aprendi muito com vocês.

Agradeço aos meus queridos amigos Glaucia, Andreza, Mayara, Bianca e meu namorado Rodrigo, que me deram todo o suporte possível durante o mestrado.

Por fim, agradeço a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES – Brasil pelo apoio e financiamento à minha pesquisa.

Sumário

Resumo	I
Abstract	II
Lista de Figuras	III
Lista de Siglas	VI
1. Introdução	1
1.1. Motivação	3
1.2. Objetivos	4
1.3. Estrutura do Trabalho.....	4
2. Referencial Teórico	6
2.1. Processo de desenvolvimento do produto	6
2.2. Gestão do Ciclo de Vida do Produto (PLM)	12
2.2.1. Fases do ciclo de vida do produto.....	12
2.2.2. Sistemas PLM	14
2.2.3. Funcionalidades dos sistemas PLM	16
2.2.3.1. Estruturação do produto	16
2.2.3.2. Gestão configuração	17
2.2.3.3. Gestão de documentos	18
2.2.3.4. Classificação e recuperação	18
2.2.3.5. <i>Workflow</i>	19
2.2.3.6. Responsabilidades e direitos de acesso	19
2.2.4. Digitalização do processo de desenvolvimento do produto.....	19
2.2.4.1. Modelo de informações do produto	20
2.2.4.2. Formatos de dados neutros	20
2.2.4.3. Estrutura dos produtos e lista de Materiais	21
2.2.4.4. Definição do produto baseada em modelos	23
2.3. Indústria 4.0	24
2.3.1. Modelo de Arquitetura de Referência para a Indústria 4.0	25
2.3.2. Modelo Integrado de Dados do Componente (ICDM)	29
2.4. Gêmeo Digital	37
2.4.1. Ciclo de vida do Gêmeo Digital	39

2.4.2. <i>Master Prototype</i>	43
3. Método	47
3.1. <i>Design Science Research</i>	47
3.2. Etapas da pesquisa.....	51
4. Proposta para o desenvolvimento do um <i>Master Prototype</i>	55
4.1. Informações ao longo do processo de desenvolvimento do produto ...	56
4.2. Modelagem do produto e PMI	60
4.3. Objetos PLM	62
4.4. Funcionalidades PLM.....	64
4.5. Visão geral do <i>framework</i>	70
5. Resultados	74
5.1. Cenário.....	74
5.2. Planejamento e esclarecimento da tarefa	78
5.3. Projeto conceitual.....	80
5.4. Projeto preliminar	85
5.5. Projeto detalhado	90
6. Conclusões	96
7. Referências	100

Resumo

A digitalização das informações do produto e dos processos de produção é um dos princípios da Indústria 4.0. Baseando-se nesses princípios, foi desenvolvido o Modelo Integrado de Dados do Componente (ICDM). Esse modelo amplia a representação do produto, gerada durante o seu processo de desenvolvimento, para que, quando instanciado, possa compreender a representação dos produtos e componentes físicos individuais durante o processo de produção. No modelo ICDM, o *Master Prototype* compreende o modelo parcial de especificação do produto. Esse modelo é usado para dar significado ao modelo das instâncias referentes aos produtos físicos. Portanto, esse *Master Prototype* pode vir a compor o Gêmeo Digital do produto. Esse trabalho visou entender como as informações do *Master Prototype* são criadas e gerenciadas durante o processo de desenvolvimento do produto e propor um *framework* para o desenvolvimento de um *Master Prototype* do produto baseado no modelo ICDM. Para isso foi usada uma abordagem PLM para a criação e gerenciamento dessas informações. O método usado na determinação do *framework* foi o *Design Science Research*. O *framework* foi verificado em um cenário fictício com a utilização das ferramentas Teamcenter® e NX®. O *Master Prototype* criado se mostrou coerente com a proposta inicial de acordo com o modelo parcial de informações do produto do ICDM para os dados de produção. Portanto, foi realizada a representação geométrica e especificações de manufatura por meio da funcionalidade de integração do Teamcenter® com o NX® e a visualização dessas informações através dos módulos de visualização de dados JT. As informações administrativas e organizacionais do produto foram estruturadas em itens e configurados para serem exibidos através de módulo *Multi-Structure Manager* do Teamcenter® e a relação dos elementos da estrutura com os requisitos de especificação iniciais foi realizada por meio de *trace links*.

Palavras-chave: Desenvolvimento de Produtos; Indústria 4.0; *Master Prototype*; Gêmeo Digital.

Abstract

The digitalization of information from products and production processes is one of the principles of Industry 4.0. Based on these principles, the Integrated Component Data Model (ICDM) was developed. This model expands the product representation generated during its development process, so that when instantiated, it may comprehend the representation of products and physical individual components during the production process. In the ICDM model, the Master Prototype comprises the partial product specification model. This model is used to provide meaning to the instances model referent to the physical products. Therefore, the Master Prototype can compose the product Digital Twin. The present work aimed at understanding how the Master Prototype information is created and managed during the product development process and at proposing a framework for the development of a product Master Prototype based on the ICDM model. In doing so, a PLM approach was employed for the creation and management of information. The method utilized for the framework determination was Design Science Research. The framework was verified in a fictional scenario aided by the tools Teamcenter® and NX®. The created Master Prototype presented to be coherent with the initial proposal according to the ICDM partial product information model for production data. Therefore, the geometric representation and manufacturing specifications were realized through Teamcenter® integration functionality with NX® and the visualization of this information through JT data visualization modules. The administrative and organizational product information was structured into items and set to exhibition through Teamcenter® Multi-Structure Manager module, and the relation between the structure elements and initial requirements specification was realized through trace links.

Keywords: Product Development; Industry 4.0; Master; Prototype, Digital Twin.

Lista de Figuras

Figura 1: Abordagem para desenvolvimento do produto (traduzido de Pahl e Beitz [1])	8
Figura 2: Constituição de um princípio de solução (Rozenfeld et. al. [2]).	10
Figura 3: Fases do ciclo de vida do produto (traduzido de Picard [10]).	13
Figura 4: Abordagem do sistema PLM (traduzido de Abramovici e Aidi [20]). .	15
Figura 5: Diferentes estruturas no ciclo de vida do produto (traduzido de Stekolschik [25]).....	22
Figura 6: Representação do RAMI 4.0 (traduzido de VDE/VDE [8], fonte: Soares [34]).....	27
Figura 7: Estrutura do invólucro administrativo (Soares [34]).....	28
Figura 8: Ciclo de vida do componente (traduzido de Strang e Anderl [43]). ...	31
Figura 9: Modelo Integrado de Dados do Componente (traduzido de Picard e Anderl [9]).....	33
Figura 10: Visão geral das etapas de criação e uso das informações do ICDM para manufatura (adaptado de Picard [10]).	34
Figura 11: Estrutura do ICDM para manufatura de peças (traduzido de Picard [10]).....	35
Figura 12: Elementos do Gêmeo Digital (adaptado de Stark e Demerau [11]).	39
Figura 13: Informações trocadas nas fases do ciclo de vida digital e físico do produto inteligente (traduzido de Abramovici et. al. [50])	41
Figura 14: Ciclo de vida do Gêmeo Digital (adaptado de Stark e Demerau [11])	42
Figura 15: Esquema de condução do DSR (adaptado de Lacerda et. al. [60]).	50
Figura 16: Visão geral do <i>framework</i> para DSR (traduzido de Johannesson e Perjons [61]).....	50

Figura 17: Método para o desenvolvimento do trabalho.	52
Figura 18: Representação das principais etapas e tipos de informações gerados no desenvolvimento do <i>Master Prototype</i>	57
Figura 19: <i>Datasets</i> podem armazenar documentos de diferentes formatos (Zhan et. al. [63]).	63
Figura 20: Visão geral do <i>framework</i> para o desenvolvimento de um <i>Master Prototype</i> do produto.	72
Figura 21: Modelagem de casos de uso para a verificação do <i>framework</i>	76
Figura 22: Representação do <i>framework</i> para a etapa de planejamento e esclarecimento da tarefa.	78
Figura 23: Sequência de mensagens para a estruturação da lista de requisitos.	79
Figura 24: Representação do <i>framework</i> na etapa de projeto conceitual.	81
Figura 25: Sequência de mensagens para estruturar em conjuntos.	82
Figura 26: Visualização do esboço da geometria da tampa do cilindro armazenado no Teamcenter®.	84
Figura 27: Sequência de mensagens para aprovar versão.	84
Figura 28: Representação do <i>framework</i> para a etapa projeto preliminar.	85
Figura 29: Sequência de mensagens para criar versão nova.	86
Figura 30: Sequência de mensagens para modelar geometria de uma peça. .	87
Figura 31: Sequência de mensagens para a montagem de conjuntos.	89
Figura 32: Representação de <i>framework</i> para etapa de projeto detalhado.	90
Figura 33: Sequência de mensagens para adicionar PMI.	91
Figura 34: Visualização do desenho detalhado da tampa do cilindro pneumático no Teamcenter®.	93

Figura 35: Sequência de mensagens para a criação de <i>trace links</i>	94
Figura 36: Estrutura do cilindro pneumático e do <i>Item Revision</i> da tampa traseira.	94
Figura 37: Visualização da geometria, PMI e da estrutura da tampa traseira no Teamcenter®.	95

Lista de Siglas

BOL	Início de vida do produto
BOM	Lista de materiais
CAD	Projeto auxiliado por computador
CAE	Engenharia auxiliada por computador
CAM	Manufatura auxiliada por computador
CAx	Sistemas computacionais de auxílio
CPS	Sistemas Físico-Cibernéticos
DMU	<i>Digital Mock Up</i>
EBOM	Lista de materiais de engenharia
DSR	<i>Design Science Research</i>
EOL	Fim de vida do produto
ERP	Planejamento dos recursos da empresa
GD&T	Dimensionamento geométrico e toleranciamento
ICDM	Modelo Integrado de Dados do Componente
IoT	Internet das Coisas
MBOM	Lista de materiais para manufatura
MBD	Definição Baseada em Modelos
MOF	Meio de vida do produto
PLM	Gestão do Ciclo de Vida do Produto

PMI	Informações de Fabricação do Produto
RAMI 4.0	Modelo de Arquitetura de Referência para Industrie 4.0
STEP	Padrão para troca de dados de produtos
TIC	Tecnologia da Informação e Comunicação
UML	Linguagem de modelagem unificada

1. Introdução

Os avanços recentes das tecnologias de informação e comunicação tem possibilitado a estruturação das informações do produto desde as etapas iniciais do processo de desenvolvimento. A partir do estudo de abordagens e métodos de desenvolvimento do produto tradicionais, como a abordagem amplamente conhecida e utilizada de Pahl e Beitz [1], é possível identificar quais informações determinam as características do produto ao longo do processo de desenvolvimento do produto.

O processo de desenvolvimento do produto parte da coleta dos requisitos do produto, segue para a determinação da função principal e subfunções, identificação dos elementos que portarão essas funções em uma estrutura funcional e de funcionamento e então são definidas as características de forma e de arquitetura do produto. No final, o produto é definido e detalhado nos documentos de produção [1,2].

Durante o processo, ferramentas computacionais de modelagem e gestão de informações do produto possibilitam a organização dessas informações em modelos de informação do produto. O padrão de troca de dados do produto, STEP, fornece uma base para a modelagem das informações do produto de maneira que elas sejam computacionalmente processáveis [3].

Os sistemas PLM dão suporte a implementação de modelos informações do produto baseados no STEP [4]. Suas funcionalidades apoiam a gestão de informações e processos relacionados ao produto envolvendo todo o ciclo de vida. As principais funcionalidades são: estruturação, gestão da configuração, gestão de documentos, gestão de colaboração e processos via *workflow*, classificação e gestão de direitos de acesso. Além disso, os sistemas PLM podem ser integrados com outros sistemas como CAD/CAE/CAM e ERP [5,6].

Os sistemas PLM tem sido alvo de estudos para a aplicação dos princípios da Indústria 4.0 e Gêmeo Digital, conceitos que fundamentam também a contribuição desse trabalho. A Indústria 4.0 envolve o uso de tecnologias de

informação e comunicação e sistemas físico-cibernéticos em uma fábrica digitalizada a fim de impulsionar a 4ª Revolução Industrial [7].

O modelo de arquitetura para a implementação da Indústria 4.0 (RAMI) [8] foi desenvolvido para ser base na aplicação desses conceitos no meio industrial. Em uma fábrica inteligente, cada peça, conjunto ou produto completo portam informações o que torna possível de se comunicarem com os outros recursos da fábrica durante o seu processo de fabricação, trazendo autonomia durante o processo [9].

As informações de cada peça e produto completo, usadas durante a fabricação, são criadas durante o processo de desenvolvimento do produto. Para o RAMI, nessa fase, o produto é denominado classe. A partir do momento em que a matéria prima é transformada em peças e depois combinada no conjunto final do produto, cada produto individual passa a ser denominado instância no mundo virtual. Ou seja, o produto é criado seguindo um modelo que é a classe, especificação ou *Master Prototype* [8].

O *Master Prototype* define as características do produto, porém de forma abstrata ou conceitual. Isso significa que, quando instanciado, o produto assume especificações individuais do produto físico que atendem a essa especificação conceitual. Esse conceito deu base para o modelo ICDM desenvolvido por Picard [10].

O *Master Prototype* é um dos três elementos do Gêmeo Digital. Os outros elementos são a sombra digital e os algoritmos que relacionam esses dois. O Gêmeo Digital é uma representação digital de um produto real que compreende suas características, propriedades, condições e comportamentos por meio de modelos, informações e dados podendo envolver uma fase ou mais fases do ciclo de vida [11,12]. Portanto, torna-se necessário determinar as características e os métodos para a criação de um *Master Prototype* [11].

1.1. Motivação

O conceito do Gêmeo Digital tem recebido destaque no meio acadêmico e industrial como uma tecnologia capaz de permitir o acompanhamento e a otimização do produto durante a fase de operação e uso, permitindo também o aprimoramento de novas gerações de produtos [13].

Para a implementação do Gêmeo Digital é necessário que estas informações estejam estruturadas de maneira que compartilhem da mesma semântica, para a descrição de vários dados, ou seja, que permita a interoperabilidade entre diferentes sistemas [14], determinando o *Master Prototype* do produto.

Essas informações são geradas ao longo do processo de desenvolvimento do produto, e compõem a classe ou *Master Prototype* do produto [8,11]. Torna-se necessário identificar em quais etapas do processo de desenvolvimento do produto essas informações são determinadas e investigar como essas informações podem ser criadas de acordo com os requisitos do *Master Prototype*.

O ICDM é um modelo de informações do componente desenvolvido para compreender informações do produto planejado e de um produto físico individual, optou-se pela proposta de um *framework* para o desenvolvimento do *Master Prototype* de um produto, tendo como base esse modelo.

Neste contexto, a abordagem PLM possibilita coletar e mapear informações e processos em torno do produto – desde a coleta dos requisitos e até o descarte – fornecendo semântica para a definição do produto por meio da implementação de modelos de dados do produto, propiciando assim o suporte aos processos de mudanças e rastreabilidade dessas mudanças ao longo do ciclo de vida do produto.

1.2. Objetivos

Esse trabalho tem como objetivo propor um *framework* para o desenvolvimento de um *Master Prototype* considerando as informações necessárias para a sua realização e requisitos para a implementação de um Gêmeo Digital.

Visando alcançar o objetivo geral, definiram-se os seguintes objetivos específicos:

- Compreender quais são as principais características do *Master Prototype* assim como quais informações ele deve conter.
- Entender como essas informações são geradas e gerenciadas ao longo do processo de desenvolvimento do produto.

1.3. Estrutura do Trabalho

Este trabalho está estruturado em sete capítulos. O Capítulo 0 apresenta a introdução com a exposição da motivação e dos objetivos. O Capítulo 2 consiste da revisão de literatura compreendendo os seguintes subcapítulos:

O Subcapítulo 2.1 apresenta as etapas do processo de desenvolvimento de produto sistemático, com a intenção de descrever como essas etapas são realizadas.

O Subcapítulo 2.2 descreve a abordagem da gestão do ciclo de vida do produto. O Subcapítulo 2.2.1 apresenta a abordagem e as fases da gestão do ciclo de vida do produto. Os Subcapítulos 2.2.2 e 2.2.3 descrevem, respectivamente as principais características e funcionalidades dos sistemas PLM. No Subcapítulo 2.2.4, são expostos alguns aspectos referentes a digitalização do processo de desenvolvimento do produto relevantes para esse trabalho.

O Subcapítulo 2.3 apresenta os elementos e princípios da Indústria 4.0. No Subcapítulo 2.3.1 é apresentado o Modelo de Arquitetura de Referência para a

implementação da Indústria 4.0 (RAMI 4.0). No Subcapítulo 2.3.2 é apresentado o Modelo de Dados Integrado do Componente (ICDM).

O Subcapítulo 2.4 explicitado o conceito de Gêmeo Digital, junto a definição, elementos. No Subcapítulo 2.4.1 contém a explicação das etapas do ciclo de vida do Gêmeo Digital e no Subcapítulo 2.4.2 são apresentadas as características do *Master Prototype* de acordo com o conceito do Gêmeo Digital.

O Capítulo 3 apresenta o método de pesquisa e as etapas para a execução do trabalho. O método é descrito no Subcapítulo 3.1 e as etapas usadas no trabalho são apresentadas no Subcapítulo 3.2.

A proposta para o desenvolvimento do *Master Prototype* do produto é descrita no Capítulo 4. O Subcapítulo 4.1 apresenta as principais informações geradas durante o desenvolvimento do produto com foco nas informações do *Master Prototype*. O Subcapítulo 4.2 contém uma breve descrição dos conceitos relacionados a modelagem do produto e inclusão das informações de manufatura do produto (PMI). O Subcapítulo 4.3 apresenta os principais objetos dos sistemas PLM de acordo com os objetos do Teamcenter®. No Subcapítulo 4.4 são apresentadas as funcionalidades PLM do Teamcenter®, que atendem as necessidades do *framework* para desenvolvimento do *Master Prototype*. O Subcapítulo 4.5 apresenta a visão geral do *framework* para desenvolvimento do *Master Prototype* do produto.

No Capítulo 5 são apresentados os resultados que consiste da verificação do *framework* em um cenário. O cenário apresentado no subcapitulo 5.1 e os Subcapítulos 5.2, 5.3, 5.4 e 5.5 apresentam, respectivamente, a verificação do *framework* para as etapas de planejamento e esclarecimento da tarefa, projeto conceitual, projeto preliminar e projeto detalhado.

No Capítulo 6 é apresentada a conclusão descrevendo como os objetivos foram atendidos, quais as limitações e sugestões para pesquisas futuras. No Capítulo 7 são apresentadas as referências bibliográficas utilizadas nesta pesquisa.

2. Referencial Teórico

Este capítulo contém uma revisão de literatura dos principais conceitos deste trabalho.

2.1. Processo de desenvolvimento do produto

O processo de desenvolvimento do produto é um conjunto de atividades nas quais a partir das necessidades do mercado, das estratégias competitivas dos produtos da empresa e das restrições tecnológicas, chegam-se as especificações necessárias para o projeto do produto e para o seu processo de produção. Envolve também o acompanhamento dos produtos após o lançamento no mercado, a fim de realizar as mudanças necessárias nas especificações, planejar a descontinuidade do produto e absorver as lições aprendidas ao longo do ciclo de vida do produto [2].

Esta fase do ciclo de vida do produto compreende tomadas de decisões nas quais as informações do produto são gradualmente criadas, selecionadas, organizadas, disseminadas e então direcionadas a fase seguinte, de produção [15]. Nesta etapa, os projetistas devem aplicar seus conhecimentos científicos e de engenharia para resolver problemas e otimizar as soluções para que atendam aos requisitos e restrições, levando em consideração aspectos como, materiais, tecnologias, economia, legalidade, ambiente e seres humanos [1].

Na abordagem para desenvolvimento do produto proposta por Pahl e Beitz [1], a descrição das atividades é essencialmente baseada nos fundamentos dos sistemas técnicos, na abordagem sistemática e nos processos para resolução de problemas gerais. O processo parte do esclarecimento da tarefa, seguindo com a identificação das funções requeridas, elaboração de princípios de solução, determinação da arquitetura, e concluída com a versão final da documentação do produto [1,16].

O processo é dividido em quatro etapas principais: planejamento e esclarecimento da tarefa, projeto conceitual, projeto preliminar e projeto

detalhado. Ele apresenta iterações para atualização das informações e otimização da solução entre as etapas. Portanto o processo consiste de etapas, nas quais, por meio da transformação de informações, conduzem o produto a sua concretização conforme ele avança [16].

A Figura 1 representa o processo apresentado em Pahl e Beitz [1] com o fluxo de atividades do processo de uma etapa a outra. As tomadas de decisões determinam a continuidade, o retorno as etapas anteriores ou a parada do processo de acordo com as necessidades dos projetistas.

Na etapa de planejamento e esclarecimento de tarefas é configurada a lista de requisitos, ou seja, a especificação inicial do produto que será desenvolvido. Esta lista é utilizada como referência ao longo do processo a fim de julgar o seu sucesso. A lista de requisitos se baseia nas informações geradas nas pesquisas e análise de mercado e são julgadas quanto aos critérios do plano estratégico da empresa. Portanto, a lista de requisitos determina o objetivo de produto e em quais circunstâncias isto será atendido [1].

A construção da lista de requisitos pode ser facilitada classificando-a em subsistemas (baseados em funções, partes e conjuntos) ou por meio de uma lista de verificação baseada em tópicos fundamentais ao produto como geometria, cinemática, força, energia, materiais, sinais, entre outros. É essencial que esta lista contenha informações sobre as funções essenciais e restrições existentes, referentes as transformações de energia, materiais e sinais. Essas informações devem ser agrupadas, ordenadas e intituladas [1].

É importante distinguir entre os requisitos exigidos e os desejados, identificar dos autores de cada requisito e registrar modificações, adições e omissões. Uma vez alinhada aos requisitos de todas as partes interessadas, como projeto, produção, clientes, logística, etc., ela se torna apta a passar para a próxima etapa [1].

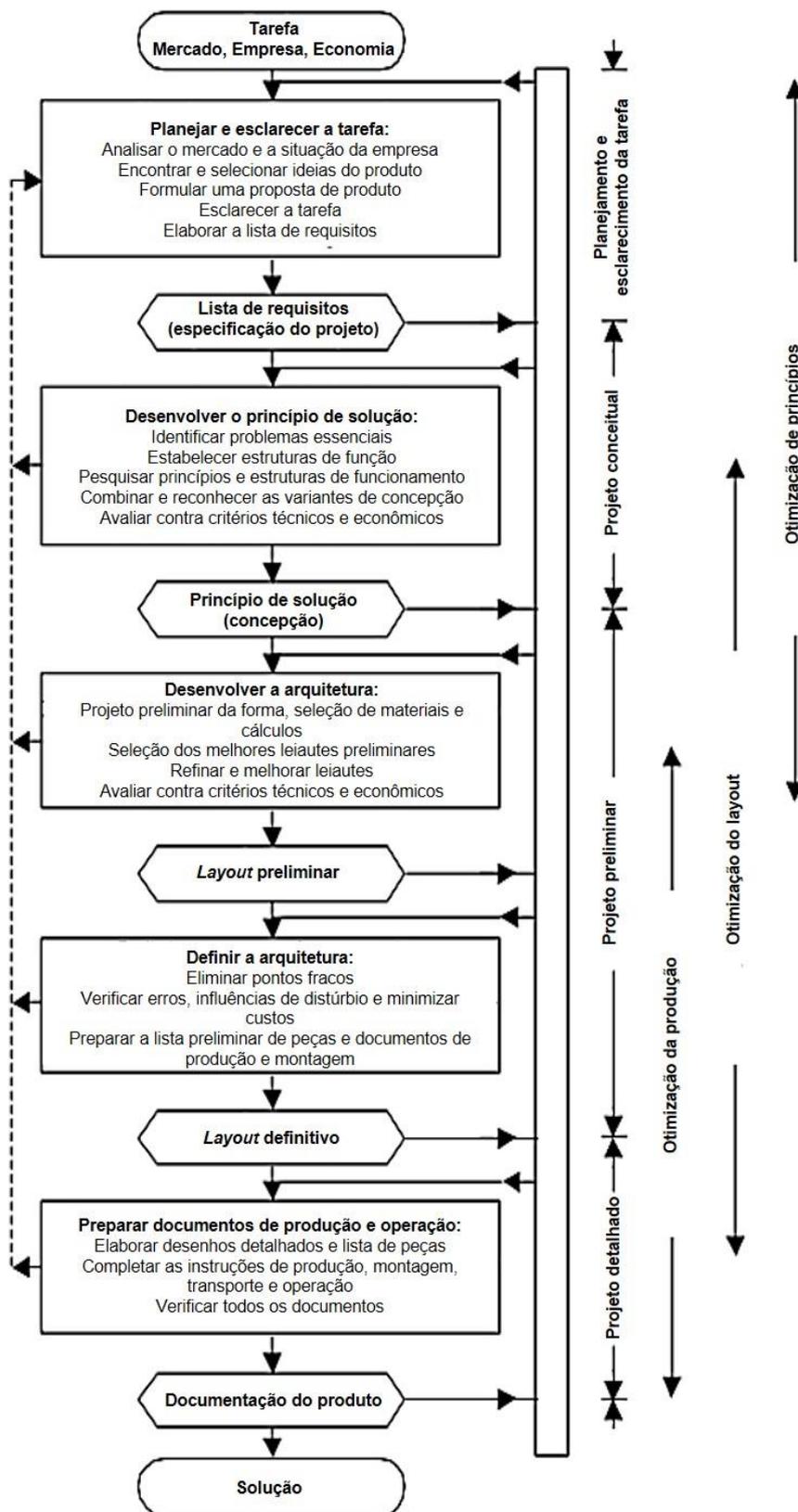


Figura 1: Abordagem para desenvolvimento do produto (traduzido de Pahl e Beitz [1])

O projeto conceitual é a etapa em que se inicia o projeto do produto. Esta etapa visa especificar a solução principal (ou as soluções principais). Ela tem como ponto de partida as decisões tomadas na etapa anterior como demanda a ser atendida, requisitos dos clientes e ideias concebidas preliminarmente, para então determinar dos princípios de solução [5,1].

Para a modelagem funcional do produto são usados alguns métodos para a abstração e ampliação da visão do problema, e para coleta de novas ideias para a solução do problema (produto) [2]. As especificações contidas na lista de requisitos podem ser submetidas a um processo de abstração, contribuindo com novas ideias. Partindo da abstração do problema, a estrutura de função é definida [1]. Essa estrutura descreve as funções e subfunções necessárias ou desejadas, que tornarão o produto apto a desempenhar seus objetivos de acordo com as especificações iniciais [2].

Após definir a estrutura funcional do produto são pesquisados princípios de solução que conduzem a soluções mais concretas. Para cada uma das funções e subfunções da estrutura funcional, são atribuídos um ou mais princípios de solução [2].

Os princípios de solução devem refletir os efeitos físicos necessários para o atendimento da função requerida, assim como características geométricas e de materiais. Para isso, buscam-se soluções tecnológicas mais viáveis partindo das funções principais para as auxiliares, estabelecendo, assim, os portadores de efeito (ou função), conforme a Figura 2. Eles podem ser obtidos por meio de bancos de dados de princípios, catálogos ou por métodos que estimulam a criatividade dos projetistas [1,2].

Informações referentes aos elementos que constituem os princípios de solução incluem: tipo de elemento, quantidade, geometria, posição, movimentos e atributos de material. É importante que essas informações apresentem apenas a forma aproximada dos elementos, como por exemplo os materiais específicos

dos elementos não precisam ser definidos, porém atributos necessários aos materiais, como ductilidade, rigidez, fragilidade, etc. podem ser determinados [2].

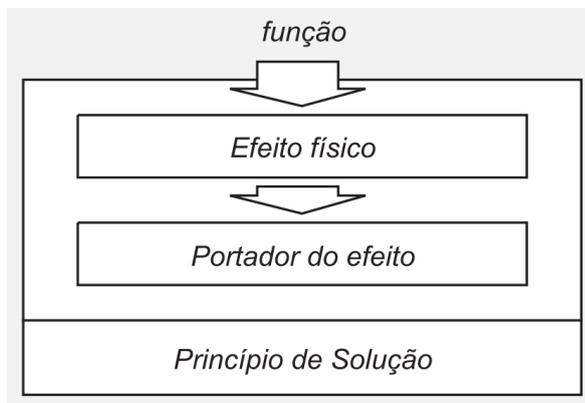


Figura 2: Constituição de um princípio de solução (Rozenfeld et. al. [2]).

Os princípios de solução são depois combinados em uma estrutura de funcionamento, gerando um princípio de solução, por meio da utilização de métodos como matriz morfológica¹ que facilita a combinação dos elementos [2,1]. Princípios de solução podem ser representados por esboços, modelos tridimensionais (*Digital Mock Up*), croquis, diagramas de bloco, descrições textuais, etc., capazes de indicar a maneira pela qual as funções são realizadas.

As alternativas de solução são examinadas quanto ao compartilhamento de funções e as restrições de compatibilidade física e geométrica e passam por avaliações rigorosas quanto a sua viabilidade técnica e econômica, assim como quanto a consistência dos princípios de funcionamento e ao atendimento aos requisitos do cliente [2,1].

O projeto conceitual é finalizado com a seleção da alternativa (ou das alternativas) mais promissora que serão utilizadas, na próxima etapas, para a determinação da arquitetura² e *layout* do produto. As informações do produto,

¹ “A matriz morfológica constitui-se de uma abordagem estruturada para a geração de alternativas de solução para o problema de projeto” [2, p. 249].

² Esquema pelo qual os elementos funcionais do produto (funções) são arranjados em partes físicas e como essas partes interagem. A relação entre os elementos funcionais e as partes físicas pode ser de um-para-um, um-para-vários e vários-para-um. Pode ter uma forma gráfica (*layout* do produto) e uma estruturada (BOM inicial) [2].

resultantes da etapa, precisam ser documentadas com clareza, determinando quais portadores de função precisam ser desenvolvidos e quais existem, ou já foram padronizados.

No projeto preliminar as principais ideias são fixadas a respeito das alternativas de princípio de solução. Esta etapa deve levar a determinação da arquitetura e *layout* geral e processos de produção [2,1]. Para o estabelecimento da arquitetura do produto é necessário que se identifiquem os sistemas, subsistemas e componentes individuais, assim como sua localização e orientação [2].

Para definir a arquitetura do produto, é importante identificar e analisar os aspectos críticos e definir os principais parâmetros do produto. Para isto são determinadas as formas, materiais utilizados, os processos de produção que serão realizados e é realizado o dimensionamento dos sistemas, subsistemas e componentes. Essas informações tomam forma em um modelo geométrico do produto [2].

Nessa etapa são criados, também, modelos de simulação e podem ser realizados testes a fim de verificar o desempenho da solução. Ao longo da etapa ocorrem várias interações de definição, criação e avaliação, até que se chegue a solução mais viável e otimizada, formando o *layout* definitivo.

A definição completa das características do produto é realizada no projeto detalhado, com a especificação final do produto. Todas as instruções a respeito da forma, dimensões, propriedades de superfície de todos os componentes, a seleção definitiva dos materiais e o exame detalhado dos métodos de produção, procedimentos de operação e dos custos são completados e detalhados [1].

Esta etapa consiste da documentação do produto, em que todas as informações relevantes do produto são preparadas e definidas para serem aplicadas nas próximas fases do ciclo de vida do produto, ou seja, para produção, uso e descarte do produto [1].

O procedimento adotado para esta etapa consiste basicamente em finalizar, integrar, completar e verificar. A finalização da arquitetura definitiva compreende o detalhamento dos modelos geométricos e otimização da forma, materiais, superfícies e combinações [1]. Simulações podem ser realizadas nos modelos finais, ou em um protótipo do produto, visando verificar conformidade em relação a lista de requisitos [2].

Os componentes são então integrados nos conjuntos e no produto completo. Os produtos e seus sistemas, subsistemas e componentes são detalhados. Esse detalhamento compreende todos os modelos geométricos do sistema, subsistemas e componentes com as cotas e tolerâncias finais, assim como a configuração final do produto que reflete na estrutura do produto e no BOM [1,2].

Nesse ponto ocorre o congelamento dessas informações em uma configuração do produto como projetada e liberada para o planejamento de produção. A configuração pode ser alterada mediante um pedido de mudança do produto [2].

2.2. Gestão do Ciclo de Vida do Produto (PLM)

O acrônimo PLM se refere ao termo *Product Lifecycle Management* que consiste de uma "atividade de negócios para gerenciar o ciclo de vida de um produto da forma mais eficiente, desde a primeira ideia de produto até seu descarte [17]." É uma abordagem fundamental para que as empresas lidem com os desafios da digitalização e da Internet das Coisas (IoT) [18]. Ela integra processos, pessoas e tecnologias, assegurando o fluxo de informações consistentes, além da rastreabilidade e armazenamento dessas informações a longo prazo [5,19].

2.2.1. Fases do ciclo de vida do produto

As fases do ciclo de vida do produto compreendem todas as etapas pelas quais o produto passa, desde sua concepção, até a sua reciclagem ou descarte [5]. Para Stark [17] o ciclo de vida do produto tem cinco etapas. Cada etapa representa o produto em um estado distinto: ideação, definição, realização,

uso/serviço e descarte, retirada ou reciclagem. Na etapa de ideação a ideia de produto ainda está na cabeça das pessoas. Na fase de definição, as ideias são convertidas em uma descrição detalhada.

O produto é concretizado ao final da etapa de realização em que está pronto para ser utilizado pelo usuário. Durante a etapa de uso, é quando o cliente consome o produto. Ao final da vida útil, o produto é retirado do mercado pela empresa ou descartado pelo usuário, existindo a possibilidade de ser reciclado também.

O ciclo de vida do produto pode ser, também, dividido em três fases. O início da vida (BOL), que compreende o planejamento do produto, o projeto do produto, planejamento de processo de produção e a produção. O meio da vida (MOL), consiste da fase de vendas e uso. A fase final de vida (EOL) representa a etapa de reciclagem ou descarte [16], conforme mostra a Figura 3. A cada fase diferentes dados e informações são gerados, alimentando a fase subsequente [10].

Nesse trabalho, a fase de interesse é a que ocorre o processo de desenvolvimento do produto, apresentada no tópico 2.1. Os sistemas PLM se destacam quanto ao gerenciamento das informações do produto nessa etapa.

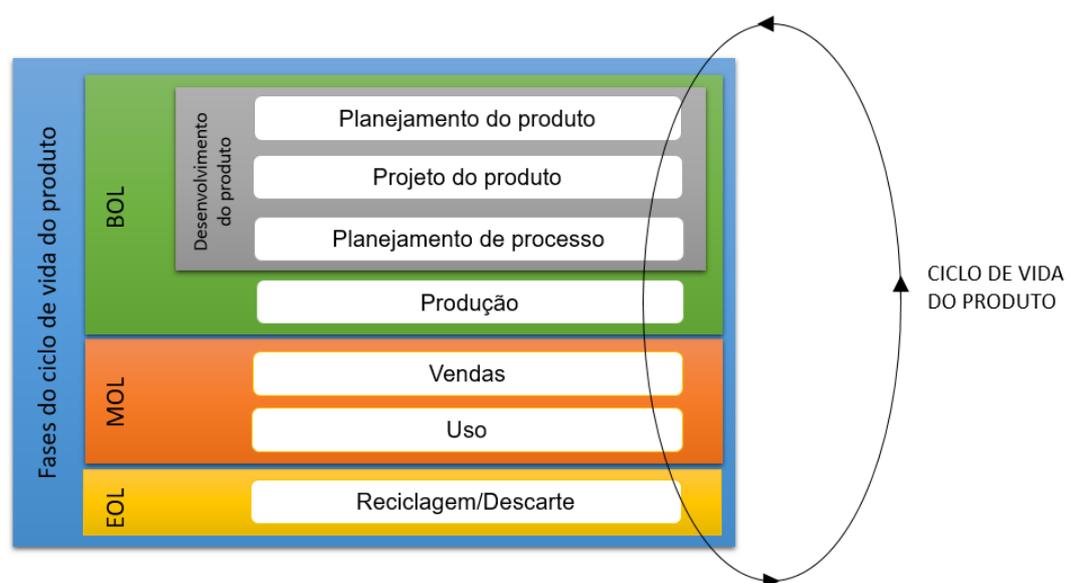


Figura 3: Fases do ciclo de vida do produto (traduzido de Picard [10]).

2.2.2. Sistemas PLM

Os sistemas PLM realizam a gestão de dados e processos que ocorrem em torno do produto ao longo de seu ciclo de vida [20]. Isso é realizado por meio da criação e configuração de estruturas para o produto, gestão de dados e documentos e do controle dos processos de desenvolvimento e mudanças do produto [4].

Os principais métodos e componentes da abordagem PLM são apresentados na Figura 4. Os métodos gerais do PLM incluem a gestão de dados e a gestão de processos. A gestão de dados do PLM abrange os métodos de análise, organização, modelagem, e a gestão dos dados (como metadados, estruturas do produto) e de documentos do produto [20].

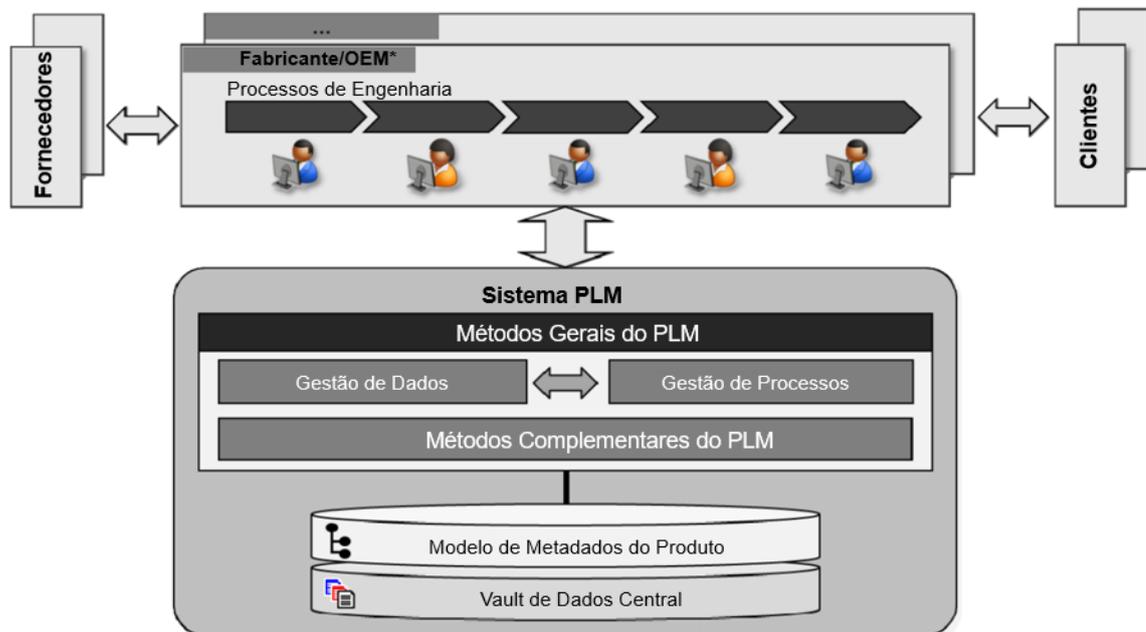
A gestão de processos inclui métodos de análise, modelagem, simulação, controle e documentação de processos administrativos específicos do PLM como a liberação e gestão de mudanças. Essas duas categorias de métodos são complementadas por outros métodos, relacionados à abordagem PLM, como gestão de acesso, para gestão de colaboração de engenharia ou métodos de apoio a tomadas de decisão [20].

Os elementos dos sistemas comerciais apresentam uma arquitetura semelhante. Geralmente, consistem de um volume (*Vault*), uma base de dados, contendo o modelo de metadados do sistema, e um aplicativo que executa as funções do sistema [4,21].

Além disso, os sistemas PLM integram outras ferramentas de criação dos dados de engenharia para utilização e armazenamento dessas informações através do sistema, como os sistemas CAx: CAD/CAE/CAM (*Computer Aided Design*, *Computer Aided Engineering* e *Computer Aided Manufacturing*) [22].

Outras aplicações para a criação de documentos podem ser associadas ao sistema para que a aplicação adequada ao tipo de arquivo usado seja iniciada. Adicionalmente, algumas aplicações podem possuir programas de

conversão de informações, convertendo dados do produto para um outro formato para facilitar a visualizações do conteúdo pelo usuário do sistema [4].



OEM: Original Equipment Manufacturer, ou "Fabricante Original do Equipamento"

Figura 4: Abordagem do sistema PLM (traduzido de Abramovici e Aidi [20]).

O *Vault* serve como uma banco de informações, contendo os documentos e arquivos referentes as diferentes fases do ciclo de vida do produto [21].

Um sistema PLM por si só não consegue interpretar o conteúdo dos documentos (arquivos). É necessário que o usuário adicione as informações necessárias – como nome do arquivo, nome do criador, relações com a estrutura do produto, ou seja metadados [4].

Os sistemas PLM atuais se baseiam na arquitetura e tecnologia orientada a objetos. Cada objeto pertence a sua própria classe de objetos que as aplicações processam por regras [4].

A base de metadados mantém toda a estrutura de informações do sistema, ou seja o modelo de metadados (ou objetos) do produto [20]. Essa base de dados lida com as relações entra as peças individuais, estruturação das informações e as regras e princípios que regem o registro sistemática das informações. Mantém

o registro dos dados do produto produzidos por diferentes sistemas e aplicações contidos no âmbito do PLM [4,21].

O aplicativo possibilita a transferência e conversão de dados e todas as funções que atendem aos princípios do PLM. É o elo entre as diferentes aplicações e sistemas e torna as conexões entre bancos de dados separados possíveis. É a partir do aplicativo que é realizada a interface do sistema com o usuário [4,21].

2.2.3. Funcionalidades dos sistemas PLM

As principais funcionalidades dos sistemas PLM são: estruturação do produto, gestão de documentos, classificação e recuperação, gestão de mudanças e controle de configuração e gestão de fluxos de trabalho [5].

2.2.3.1. Estruturação do produto

A estrutura do produto é a parte central do sistema PLM. Por meio dela os componentes, documentos, conjuntos e outros elementos são ligados ao produto e relacionados uns aos outros. Esta funcionalidade é a base para a realização das funcionalidades básicas do sistema [4,21].

Estruturas são construídas a partir de objetos fundamentais para a organização dos dados do produto nos sistemas PLM denominados itens. Um objeto item é um elemento de dados que descreve certo componente ou elemento do produto, incluindo módulos, subsistemas ou conjuntos [4]. A estrutura de um produto possui a identificação dos itens e a definição do relacionamento entre eles, além de relacionar esses itens aos seus respectivos dados e documentos [5,4].

Na estruturação do produto, as funções definidas que devem atender aos requisitos do produto são combinadas em partes físicas. Essas partes são detalhadas em componentes individuais vindo a integrar à estrutura [5,6].

Os sistemas PLM permitem ao usuário definir várias “visões” (como visão de projeto, visão de manufatura, etc.) a partir de uma única estrutura do produto de forma que a integridade da estrutura é mantida. É possível também determinar

quais peças são alternativas ou substitutas em uma mesma estrutura, sendo reduzido o número de estruturas de produtos que precisam criadas e para melhor gerenciamento dessas estruturas [4,5,6].

2.2.3.2. Gestão de configuração

A gestão de configuração abrange a gestão de versões, *status* e validação, controle de aprovações, gestão de mudanças e o controle de configuração. As versões indicam as alterações ou revisões realizadas nos objetos [4]. Essas revisões, são armazenadas manualmente ou automaticamente no sistema PLM [5].

É possível verificar o *status* que especifica a situação dos objetos (como, preliminar, em avaliação, liberado), definindo assim as ações permitidas ao objeto no sistema. É possível indicar também em quais situações eles serão empregados (como, datas, clientes, regiões geográficas). Essa combinação define como o PLM vai gerenciar a evolução dos objetos ao longo do ciclo de vida do produto [5,4].

O controle de aprovações gerencia processos de análise e de aprovação dos objetos no sistema, causando ações automáticas no sistema como congelamento de objetos, para evitar alterações, o envio de dados para outros sistemas e a notificação das pessoas envolvidas [5,4].

É possível definir no sistema PLM eventos para registrar diferentes configurações do produto que podem ser congeladas, para permitir que as informações sejam rastreadas. O sistema possibilita o rastreamento das configurações a qualquer instante [5].

A gestão de mudanças permite que a última informação válida sobre as mudanças em objetos e componentes (como mudanças de versões) seja registrada em documentos ou itens, tornando-se disponível para as partes apropriadas [4].

A funcionalidade de controle de configuração possibilita o acompanhamento de toda a evolução da configuração do produto ao longo do ciclo de vida. São determinados os objetos que devem ser gerenciados e as configurações que devem ser congeladas para que sejam rastreadas a qualquer momento [5].

2.2.3.3. Gestão de documentos

Os documentos são armazenados no *vault* e descritos por metadados. Ou seja, são representados por objetos no sistema e, portanto, apresentam propriedades, comportamento e relações [15]. Cada documento pode ser relacionado a outros objetos incluindo aos itens [5].

Os documentos armazenados no *vault* contêm um conjunto de informações sobre o produto. Esses documentos são estruturados para a compreensão humana e, portanto, tratados como uma unidade. Eles contêm informações sobre o produto, sendo essenciais para a PLM [15].

Documentos podem ser de vários formatos, podendo incluir dados produzidos por sistemas CAx, ou outros relacionados a engenharia de manufatura ou dados produzidos por outros aplicativos como processadores de texto, planilhas eletrônicas, imagens e etc. [4].

O mecanismo *check in* e *check out* é utilizado durante a criação, atualização e aprovação e na solicitação dos documentos contidos no *vault*. O acesso aos documentos podem ser solicitados de acordo com a permissão de acesso estabelecida no sistema [4,21].

2.2.3.4. Classificação e recuperação

O agrupamento claro e lógico dos itens em classes, subclasses e grupos facilita o gerenciamento e a recuperação de itens individuais [4]. As classes podem ser organizadas em uma hierarquia. Para organizar as informações são criados atributos que descrevem cada item no sistema [5].

Os objetos podem ser classificados quanto as suas características específicas (como diâmetro) e classes (e.g. eixo dianteiro). Desta forma, eles podem ser

pesquisados quanto a esses atributos específicos, de forma que, objetos de uma mesma classe, possam ser comparados durante a recuperação de informações [5,4].

2.2.3.5. Workflow

O *workflow*, ou fluxo de trabalho, monitora a ocorrência de eventos. Quando um evento ocorre, segue uma série de tarefas predefinidas no sistema. Informações sobre a ocorrência ou não ocorrência dos eventos, são enviadas as partes apropriadas. O estado do processo é atualizado a todo momento, sendo possível rastrear e gerar relatórios sobre as tarefas em andamento [17].

Essa funcionalidade permite que haja colaboração entre membros do times de projeto por meio das interfaces de comunicação, que geram cooperação, além de compartilhamento de aplicativos e bancos de dados de conhecimento [5]. A comunicação e divisão de tarefas é realizada por e-mail ou lista de tarefas e a visualização do fluxo por meio de modelos gráficos [4].

2.2.3.6. Responsabilidades e direitos de acesso

Por meio a definição de responsabilidades e direitos de acesso aos usuários é possível determinar regras quanto ao acesso aos dados e informações relevantes, responsabilidade em endereçar *workflows*, e especificar o caminho no qual as informações relevantes são fornecidas de maneira mais eficiente. Essas responsabilidades e direitos de acesso podem ser relacionados aos cargos (projetista, analista de materiais, etc.) assim como hierarquias (gerente, líder de grupos, etc.) nas organizações [6].

2.2.4. Digitalização do processo de desenvolvimento do produto

Os aspectos do processo de desenvolvimento digitalizado fundamentais para esse trabalho são: o modelo de informações do produto, formatos de dados neutros, estruturas do produto ao longo do ciclo de vida, definição baseada em modelos.

2.2.4.1. Modelo de informações do produto

O modelo de informações do produto é um modelo conceitual que analisa as informações sobre o produto e a sua relação com outros modelos de informação, descrevendo esta relação em um nível conceitual. O objetivo é definir cuidadosamente o conceito do produto e analisar o produto em um nível geral, examinando suas propriedades comuns e formas comuns de informação, gerando um modelo de informações genérico para o produto [4].

O modelo integrado de dados do produto, STEP (*Standard for the exchange of product data model*), também norma ISO 10303 foi desenvolvido no intuito de integrar todas as informações do produto, permitindo troca de dados entre os sistemas e comunicação e colaboração em todas as etapas do ciclo de vida do produto [10,23].

Sendo utilizado amplamente em sistemas CAD, essas informações do produto são divididas em três tipos: informações de: definição do produto, representação do produto e apresentação do produto [10,23].

A definição do produto fornece a identificação do produto assim como dados administrativos e organizacionais relacionados ao projeto do produto. A representação do produto consiste em dados descritivos do produto relativos a forma e propriedades do material de produto, visualizada por meios computacionais. A apresentação inclui os dados necessários para a apresentação dos dados do modelo de representação do produto, de forma gráfica e textual [10].

2.2.4.2. Formatos de dados neutros

Um dos principais desafios na utilização de sistemas PLM consiste da heterogeneidade dos dados de onde são retiradas as informações. Estas informações variam quanto ao tipo (fotos, vídeos, arquivos de texto, arquivos CAD, entre outros), os arquivos CAD podem variar quanto ao seu formato como formatos herdados e formatos neutros (STEP, JT, IGES, STL, entre outros) [24].

Formatos proprietários são específicos das ferramentas proprietárias por exemplo CATIA, CREO, NX, e, portanto, protegidas, dificultando a colaboração. A falta de abertura desses formatos levou ao desenvolvimento de formatos de dados para transferência de dados entre diferentes softwares CAx [24].

Apesar de ser o formato neutro que envolveu os maiores esforços já empreendidos para a criação de um formato de dados padrão, o STEP possui algumas limitações, como por exemplo, é um formato que gera um arquivo “pesado” e difícil de ser comprimido [23,24].

O STEP envolve dados de produto referentes a projetos mecânicos, elétricos, dimensões geométricas e tolerâncias, análise e manufatura. Abrange também informações específicas de várias indústrias como automotiva, aeroespacial, construção, naval, plantas de processo, etc. [23].

O formato de dados JT, em oposição ao formato STEP, é “leve” e consiste de uma representação tessellada do produto. Esse formato é utilizado principalmente para aplicações industriais como um meio de capturar e reaproveitar dados de definição de produto 3D leves [23,24].

É usado tanto como um formato de troca de dados entre parceiros de projeto e fabricantes, como também para aplicações de visualização, como pré-montagem digital (também chamada de *mock-up* digital ou DMU) e visualização, mais comumente referida como visualização/medição/marcação (VMM) [23].

2.2.4.3. Estrutura dos produtos e lista de Materiais

As estruturas do produto podem refletir diferentes perspectivas do produto (projeto e desenvolvimento, planejamento da produção, etc.) e graus de maturidade do produto, como representado na figura [25].

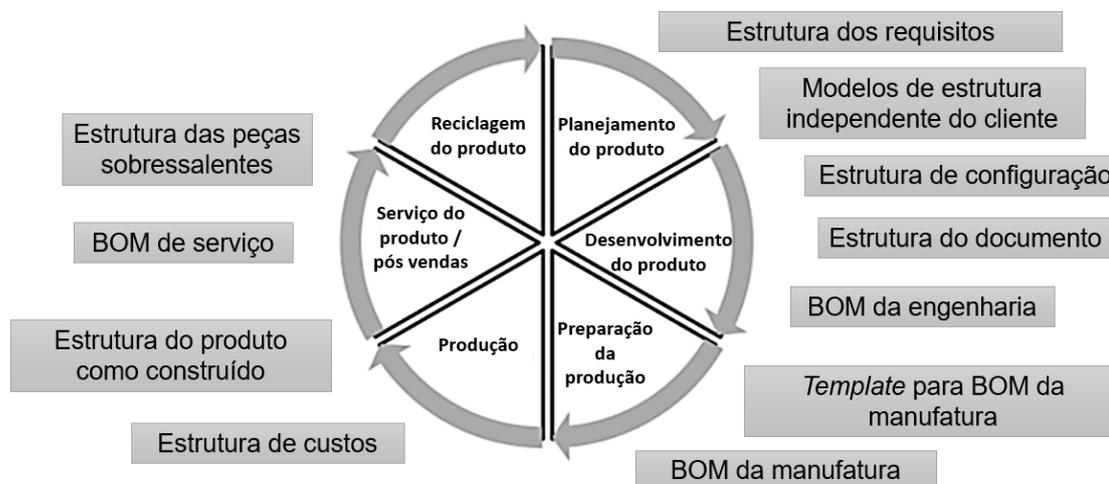


Figura 5: Diferentes estruturas no ciclo de vida do produto (traduzido de Stekolschik [25]).

Essas estruturas são usadas de diferentes maneiras que caminham junto as percepções correspondentes, resultando em requisitos relativos as estruturas. Diferentes formas de aparência - listas de requisitos, lista de materiais, instruções de serviço, estruturas de custo, etc. - podem ser encontradas nos requisitos de produto e estruturas funcionais, estruturas de projeto e manufatura e eventualmente estruturas de serviço [23,25].

O processo de criação dessas estruturas é um processo iterativo e sua progressão estabelece interdependências com outras estruturas. O grau de maturidade e completude de uma estrutura aumenta a cada passo do ciclo de vida [25].

A estrutura de engenharia, criada ao longo do processo de desenvolvimento do produto, é denominada EBOM (*Engineering Bill of Materials*) e representa uma estrutura ou lista de materiais “como projetada” do produto [2,25]. Essa estrutura reflete a visão multidisciplinar do produto baseada na definição funcional do produto, se assemelhando a estrutura de conjuntos no CAD. Geralmente, a EBOM combina documentos de especificação do produto, modelos CAD, desenhos, anotações que dão uma definição clara do produto [25].

A estrutura de fabricação MBOM (*Manufacturing Bill of Materials*) apresenta a configuração de como o produto será fabricado, contendo os conjuntos e subconjuntos dos itens que descrevem o processo de fabricação. Geralmente a MBOM se baseia na EBOM, ou seja, utiliza seus itens, organizando-os de maneira apropriada as especificidades do seu processo de fabricação [2,25].

2.2.4.4. Definição do produto baseada em modelos

A definição do produto baseada em modelos (*Model Based Definition -MBD*) é um conjunto de dados compreendendo geometria 3D e anotações semânticas. Esse pacote consiste de modelos, informações de manufatura (PMI) requisitos de desempenho, documentação, metadados e outros detalhes [26].

Esse conceito tem como objetivo que os processos posteriores a definição de dados do produto, utilizem modelos 3D como fonte de informações para a realização de suas atividades estimulando um fluxo de informações contínuo e eficiente ao longo das fases do ciclo de vida do produto. Essa otimização no acesso aos dados possibilita automação e economia de custos e tempo [26,27,28].

Portanto o MBD não diz respeito a um formato ou ferramenta, mas sim, consiste de uma definição de um “3D Master” com suas descrições associadas e arquivos técnicos para assim apoiar a interoperabilidade. Esse conceito pode ser implementado com o uso de vários formatos como STEP, JT ou PDF [29].

O MBD geralmente envolve informações PMI (*Product Manufacturing Information*). Essas informações PMI, inseridas nos modelos CAD, utilizadas para suportar digitalmente os processos de manufatura. São atribuídos ao modelo, dimensões geométricas e de tolerâncias (GD&T) e não geométricas como especificações de materiais, processos e requisitos para inspeção [26,30].

A simbologia presente nos GD&T dividem-se em: simbologias de apresentação, baseadas nas normas ASME Y.14.41-2012 e ISO 16792-2015 e sintaxe e semântica, referentes aos padrões industriais que utilizam as normas ASME

Y.14.5-2009 e ISO 1101-2017 [30]. As anotações semânticas se referem as anotações digitais que fornecem atributos aos produtos com semântica, para posterior processamento computacional [26].

2.3. Indústria 4.0

A Indústria 4.0 representa uma mudança no paradigma da produção industrial, levando a melhorias significativas nas cadeias de valor em todo o ciclo de vida do produto [31,17,32]. Com o intuito de enfatizar a importância da tecnologia de produção e fortalecer a competitividade para indústria manufatureira alemã, o governo alemão criou a *Plattform I4.0*, uma plataforma que desenvolve e publica periodicamente recomendações para implementação da Indústria 4.0 [33,17,32,34,35,7,36].

A Indústria 4.0 é caracterizada pela disposição de tecnologias de informação e comunicação (TICs) e de sistemas físico- cibernéticos (CPS) a fim de obter vantagens para atender as tendências de redução dos ciclos de vida dos produtos, produção mais rápida, alta personalização de produtos, custos reduzidos no desenvolvimento e produção de produtos e maior disponibilidade de dados e informações. Proporciona também interoperabilidade, flexibilidade, eficiência e tomadas de decisões autônomas [31,37].

A fábrica inteligente é um recurso essencial da Indústria 4.0 [33]. Neste ambiente, os CPS se comunicam por meio da IoT auxiliando pessoas, máquinas e produtos na execução de suas tarefas, por meio da percepção do contexto, proporcionada pelo compartilhamento de informações entre o mundo digital e o mundo físico [37,33,32,38].

Nesse contexto, os processos de produção anteriormente centralizados passam a ser descentralizados e os produtos são capazes de auto percepção em que conhecem o seu histórico de produção, seu estado atual e estado alvo. Em uma ambiente de manufatura inteligente os produtos são capazes de orientar seu processo de manufatura de forma ativa, instruindo máquinas a executarem as atividades de fabricação necessárias [33,17].

No ambiente de fábrica inteligente, os CPS gerenciam, monitoram, controlam e comunicam as informações de forma bidirecional entre a parte física e a digital. Para que ocorra a interação entre o mundo físico e o digital é necessário que o “componente cibernético” crie uma representação digital do “componente físico”. Essa representação consiste em dados e informações digitalizadas [33,39,32].

Os princípios para a realização da Indústria 4.0, são: digitalização, interoperabilidade, descentralização, disponibilidade em tempo real, flexibilidade, orientação a serviço. As empresas são capazes de monitorar seus processos físicos por meio da digitalização. A interoperabilidade consiste na conexão dos sistemas dentro e fora da empresa, isso pode ser facilitado pela padronização da comunicação [37,32].

A descentralização permite que os CPS tomem decisões de forma autônoma em que, somente nos casos de falhas, as tarefas são delegadas a um nível superior. Contudo, é preciso garantir a qualidade e rastreabilidade permitindo o acompanhamento do sistema a qualquer momento [37,33,32]. Os dados precisam ser coletados e analisados em tempo real, para que a planta possa reagir a falha de uma máquina e redirecionar produtos a outra máquina, por exemplo [37,33,32].

Os processos de desenvolvimento, produção e procedimentos de tomadas de decisão em torno dos produtos precisam ser mais flexíveis, permitindo alto grau de personalização. Isto é proporcionado pela modularização. Por fim, por meio da orientação a serviços, os serviços das empresas através da internet podem ser usados por outros participantes de dentro ou fora de uma empresa [37,32].

2.3.1. Modelo de Arquitetura de Referência para a Indústria 4.0

O Modelo de Arquitetura de Referência para a Indústria 4.0 (RAMI 4.0) (Figura 6) é resultado dos esforços da Plataforma I4.0 em padronizar os aspectos da interconexão entre os mundos físico e cibernético nos sistemas de manufatura inteligente [34].

São três os aspectos principais da Indústria 4.0, que guiam o desenvolvimento deste modelo:

- A integração vertical, que descreve a interconexão dos meios de produção entre vários níveis corporativos da empresa [34,8].
- A engenharia ponta-a-ponta, em que os dados técnicos, administrativos e comerciais, fornecidos pelo meio de produção e pelo produto ou componente, são mantidos consistentes e acessíveis no fluxo de valor como um todo [34,8].
- A integração horizontal, que amplia a rede de agregação de valor da empresa transpondo as fronteiras de uma planta, passando a compreender e interligar, de forma dinâmica, todos os participantes diretos e indiretos do processo de manufatura [34,8].

O modelo desenvolvido consiste de três dimensões. Na dimensão vertical, as várias camadas inter-relacionadas representam as várias perspectivas da empresa [34,16]. A camada mais abaixo indica os recursos reais da empresa, compreendendo recursos abstratos como documentos, modelos, ideias, assim como recursos concretos, como máquinas, peças, sensores e atuadores e pessoas [8].

A camada de integração consiste dos sistemas de informação que fornecem as informações sobre os recursos, sobre elementos para comunicação passiva, interface humano-máquina, e que podem ser computacionalmente processadas. A camada de comunicação se baseia em protocolos de comunicação e formatos de dados uniformes e fornece serviços para controle da camada de integração. A camada de informação realiza o processamento dos dados da camada de integração para fornecer dados estruturados à camada funcional. [8].

A camada funcional compreende a descrição formal das funções e se refere a integração horizontal das diferentes funções e da criação de regras e lógicas de aplicação. A última camada, de negócio, assegura a integridade das funções no

fluxo de valor e ligação de diferentes de negócios. Nela também ocorre o mapeamento dos modelos de negócio e do processo resultante, assim como são modeladas as regras que o sistema deve seguir. Nesta camada, consideram-se aspectos legais e regulatórios [16,8].

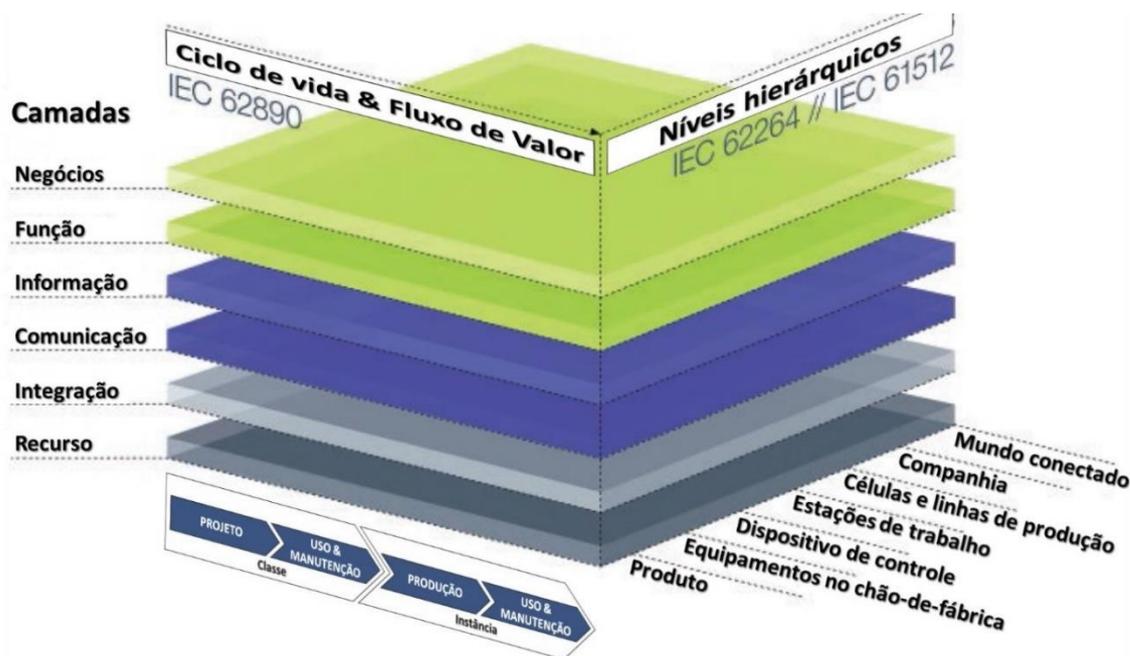


Figura 6: Representação do RAMI 4.0 (traduzido de VDE/VDE [8], fonte: Soares [34]).

Na dimensão de níveis hierárquicos, estão localizadas as funcionalidades e responsabilidades dentro da empresa [16]. A hierarquia se inicia com o produto no nível mais baixo. O produto é um elemento importante no contexto da manufatura inteligente sendo considerado um componente 4.0 por sua habilidade de comunicação no sistema de produção [8,16,40].

O nível aumenta conforme a abrangência do sistema, partindo de dispositivos de campo, com sensores e atuadores, nível de estação, compreendendo máquinas de produção, centros de trabalho, contendo plantas de produção e departamentos, nível de companhia – referindo-se a empresa como um todo - e por fim o mundo conectado, representando a ligação com outras fábrica, colaborações externas, fornecedores e clientes [16,8].

A dimensão Ciclo de vida e Fluxo de Valor, divide o ciclo de vida do produto e componentes em classes e instâncias. A classe consiste da definição dos componentes, concebida na etapa de desenvolvimento, formando a classe geral, da qual são derivadas as instâncias (produtos físicos) durante a produção [8].

Essas instâncias recebem um número de identificação e são envolvidas por um invólucro administrativo [34]. Esse invólucro administrativo contém todas as informações relevantes dos recursos, e as mantém em submodelos (Figura 7) [41].

Um objeto, ou recurso (incluindo produto e peças), somente é considerado “componente I4.0”, quando envolvido por um invólucro administrativo [8]. Portanto, componentes são objetos, que fornecem dados e informações em um sistema de informação, ao mesmo tempo que tem comunicação compatível ou seja são capazes de armazenar o invólucro administrativo em si [34].

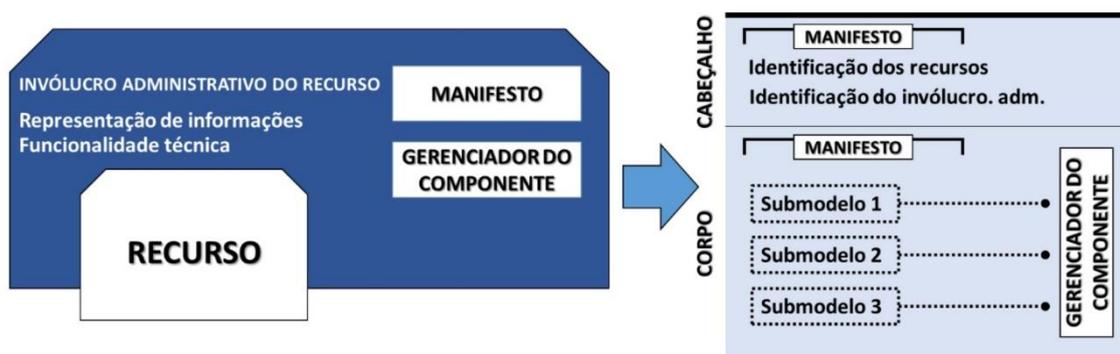


Figura 7: Estrutura do invólucro administrativo (Soares [34]).

Uma classe contém um conjunto de objetos similares contendo a descrição das suas propriedades comuns³ [42]. Ela compreende a fase de desenvolvimento do recurso, iniciando-se com a ideação – ou conceituação - passando pelos testes e prototipagem, até ser validada e liberada para a produção [34,8,42]. A partir de

³ Neste ambiente, propriedades também possuem relacionamento classe e instância. Por exemplo: uma classe de materiais deriva uma instância específica de material.

um modelo criado e fornecido pela classe, que ocorrem as instanciações do recurso [42].

O relacionamento entre classes e instâncias deve ser mantido ao longo do ciclo de vida do recurso. Por meio desse relacionamento, é possível atualizar o recurso a partir das atualizações da classe de forma automática, assim como também transferir informações referentes ao uso de um recurso para a sua respectiva classe a fim de otimizar próximas gerações referentes aquele produto [41].

O invólucro administrativo consiste de um “manifesto” e um “gerenciador do componente”. O manifesto é um diretório com dados individuais baseados em metadados para a descrição digital do recurso. A descrição digital corresponde a todas as informações digitais do produto, por exemplo: modelo geométrico, estrutura de um produto com seus vínculos com outros componentes, imagens, catálogos, modelos de simulação, entre outros [34].

O invólucro administrativo possui um cabeçalho e um corpo. O cabeçalho contém detalhes sobre o invólucro administrativo e sobre os recursos representados. O corpo compreende os submodelos que formam o invólucro administrativo de acordo com os recursos específicos [41].

Informações são estruturadas nos submodelos de acordo com as diferentes disciplinas de engenharia relacionadas ao recurso [41]. O gerenciador tem a função de realizar uma ligação para a comunicação do recurso CPS com as TICs. Essa ligação permite a troca de dados e o acesso autorizado aos dados do invólucro administrativo, ou seja, informações de representação digital e funcionalidade técnica [34,41,42].

2.3.2. Modelo Integrado de Dados do Componente (ICDM)

O Modelo Integrado de Dados do Componente (ICDM) foi desenvolvido com o objetivo de ampliar a representação do modelo do produto a fim de contemplar as informações do produto e seus componentes físicos individuais durante a

produção. O modelo consiste de uma representação estruturada de informações para cada um dos componentes, individualmente, e que integra produto, processos, recursos de produção e pessoas, envolvendo todas as etapas do ciclo de vida do produto [9,43].

Na fábrica inteligente, ele facilita a adaptação descentralizada do produto e seus componentes aos processos de manufatura permitindo o fluxo de informações bidirecional entre os dados digitais e dados dos componentes físicos individuais. Ou seja, o produto e seus componentes são portadores de informação com função ativa nos ambientes de produção. Portanto, refere-se aos sistemas físicos cibernéticos, e suas capacidades de comunicação e colaboração com outros sistemas físico cibernéticos [9,44,34].

Na abordagem de ciclo de vida do componente introduzida por Strang e Anderl [43], são distinguidos dois tipos fluxos de valor, representados na Figura 8. Um deles diz respeito ao ciclo de vida do componente físico e outro da sua representação virtual.

A cada etapa os componentes passam por fluxos de agregação de valor. As instâncias são representadas de acordo com o modelo ICDM. Dessa forma ocorre a combinação de dados referentes ao estado atual do componente físico com os dados de definição do produto criados durante processo de desenvolvimento do produto, em que foram planejadas as condições que o componente deve assumir durante a produção e/ou em outras etapas do ciclo de vida [9].

A primeira fase do ciclo de vida virtual representa as especificações gerais do produto como requisitos, funções, a estrutura e materiais. Na fase seguinte essas especificações são combinadas ao pedido específico de acordo com a configuração do produto, solicitada pelo cliente [44].

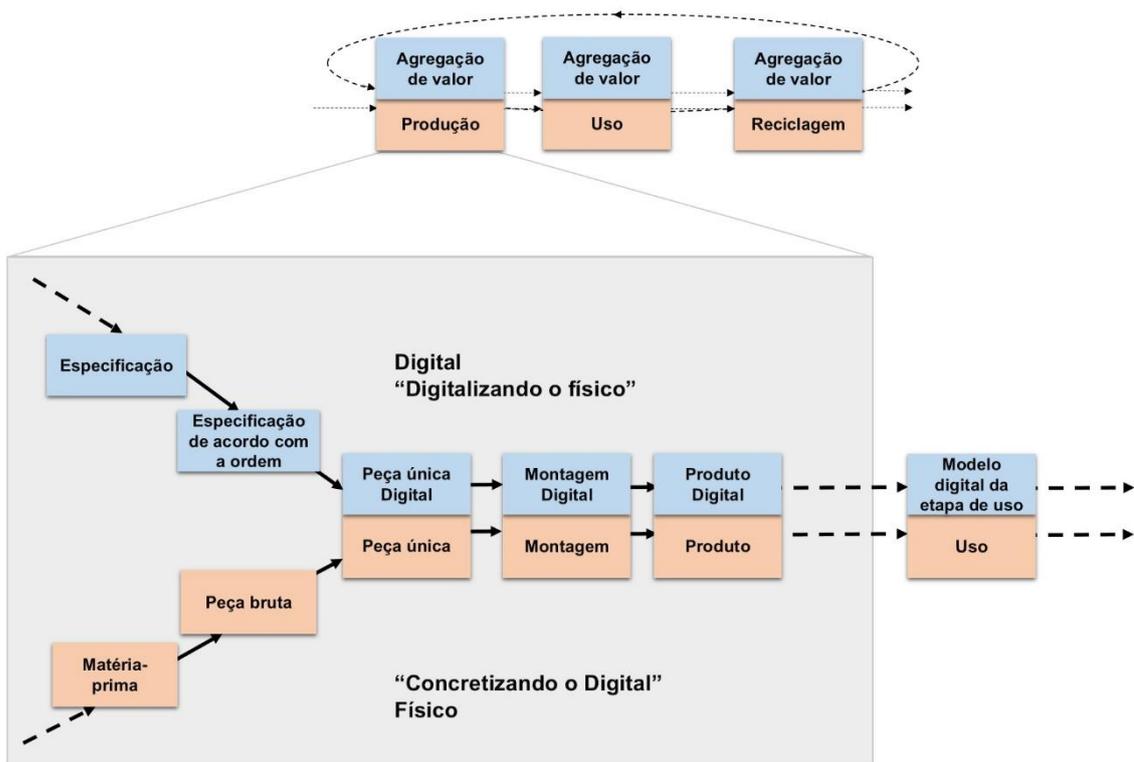


Figura 8: Ciclo de vida do componente (traduzido de Strang e Anderl [43]).

Na fase inicial do ciclo de vida no mundo físico, o produto ainda não foi fisicamente instanciado, portanto, existem apenas informações digitais sobre o produto e a matéria-prima. As informações digitais do produto e componentes são anexadas, no início da manufatura, a peça bruta, que será transformada e comporá o produto ao longo do processo de produção. Portanto, ambos os ciclos, virtual e físico, convergem nessa fase na qual o componente recebe um número de identificação e capacidade de endereçamento e localização de outros componentes.

Conforme as partes reais, como peças brutas, produtos semiacabados e componentes são fabricados eles portam informação sobre si mesmos e dos processos, estabelecendo referência a outros objetos, para formarem conjuntos. Na representação do produto são apresentadas as interações entre componentes mecânicos, elétricos e de *software* assim como sua lógica funcional [44].

O modelo ICDM combina dois conceitos: o modelo de dados integrado do produto, STEP, e a memória semântica do produto (SemProM). O STEP é adotado a fim de reduzir a heterogeneidade dos dados do modelo do produto e definir uma estrutura central padronizada [9].

A memória digital do produto permite que os dados do produto, e do seu ciclo de vida, sejam armazenados diretamente no produto físico. A memória semântica do produto é uma subclasse da memória digital do produto [9]. A memória digital do produto fornece uma descrição das propriedades do componente legível por máquinas e a memória semântica fornece uma descrição compreensível por máquinas baseando-se em tecnologias de *web* semântica, e possibilitar acesso aberto aos dados [9,45].

Essas informações são anexadas fisicamente ao componente e são acessadas por meio de sistemas de identificação como RFID (*Radio Frequency Identification Transponder*) e códigos bidimensionais, possibilitando o acesso às informações via internet [9]. O emprego dessas tecnologias possibilita uma comunicação compatível, de acordo com a Indústria 4.0.

O modelo ICDM consiste de um modelo central e modelos parciais (Figura 1). O modelo central especifica a identificação, endereçamento, localização, informações administrativas e organizacionais e representações geométricas do componente [9].

Os modelos parciais são, inicialmente, divididos em quatro. O modelo parcial de identificação e especificação contém informações sobre requisitos, modelos do produto e especificação do material. O modelo de dados de gerenciamentos de direitos envolve as informações sobre os usuários, grupos, papéis e pontos de vista. O modelo de dados de ordem de produção contém os dados de ordens de produção contém os dados do cliente e as condições de entrega [34,9].

O modelo parcial de produção contém os dados de produção e pode ser dividido em dois modelos específicos: manufatura de peças individuais e montagem de peças individuais para um produto final [43]. O ICDM integra dados do

desenvolvimento do produto e planejamento do processo de produção, a fim de fornecer suporte para o controle e monitoramento dos processos [9].

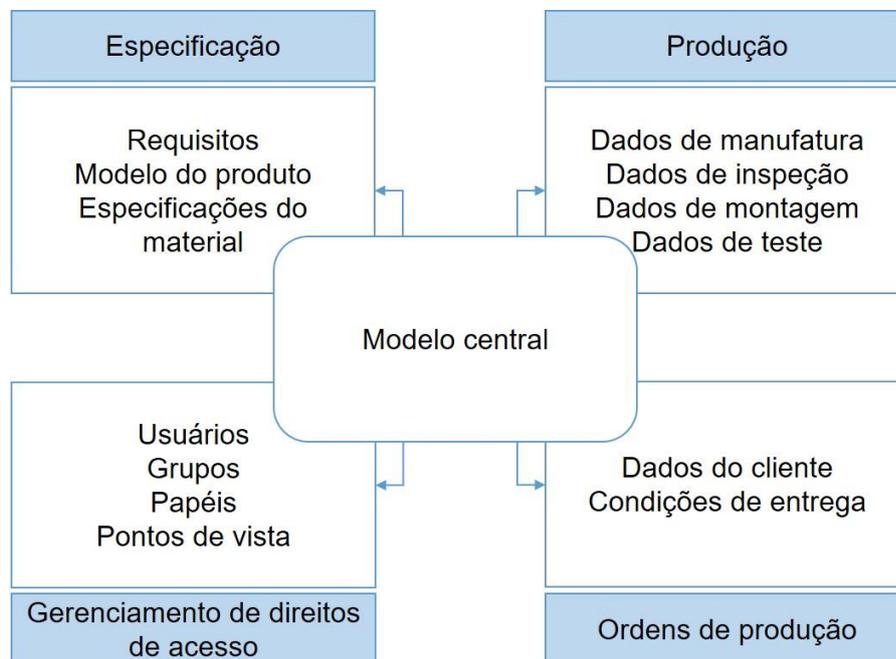


Figura 9: Modelo Integrado de Dados do Componente (traduzido de Picard e Anderl [9]).

Além disso, existe a possibilidade do ICDM pode ser estendido para coletar informações ao longo de todo o ciclo de vida do produto. Ele pode ser adaptado para receber informações relevantes de outros processos referentes as fases de uso e de reciclagem [34].

Os modelos parciais são uma extensão do modelo central. Portanto devem atender aos pré-requisitos integridade, coerência, acumulação e associação. A integridade diz respeito a exatidão/veracidade da semântica da informação. A coerência se refere a capacidade de correlação da informação sem transformação. A acumulação permite a representação explícita de todas as informações relevantes. E a associação é o resultado das informações implícitas [9,34,44].

Picard [10] apresenta em seu trabalho o desenvolvimento e aplicação do modelo ICDM no processo de manufatura de peças. A realização do ICDM compreendeu três etapas conforme apresentado na Figura 10.

Na primeira etapa o produto é desenvolvido, durante o processo de desenvolvimento do produto. A definição do produto é realizada por meio de sistemas CAD. O modelo de informações de especificação do produto é criado a partir dos objetos instanciados no CAD. São criados os dados que descrevem as formas geométricas, dados não geométricos como especificações de materiais e superfície e de comportamento da peça. Nessa etapa é criado o modelo de especificação do produto, ou seja, *Master Prototype*.

Na segunda fase, os responsáveis pelo planejamento da produção determinam o fluxo de trabalho. Como resultado, o plano de fluxo de trabalho é mapeado em um modelo de informações do processo de produção. Este modelo especifica as operações discretas e sua sequência de processamento.

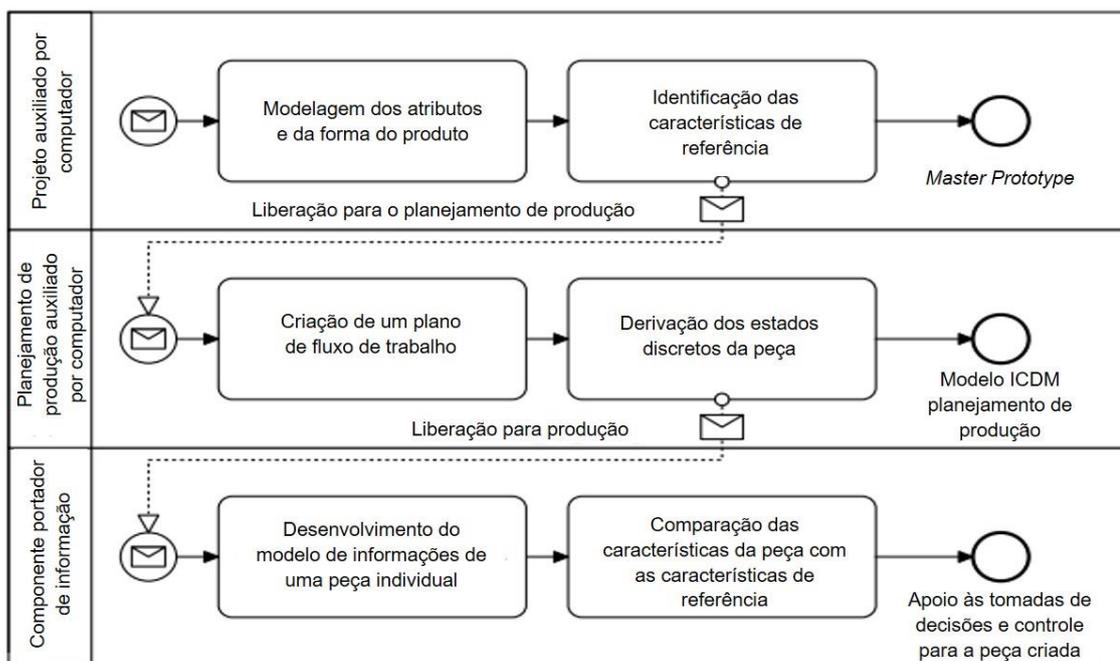


Figura 10: Visão geral das etapas de criação e uso das informações do ICDM para manufatura (adaptado de Picard [10]).

Os dados das características do produto, contidos no *Master Prototype* são usados como referência para a definição da sequência dos processos de produção e das informações de planejamento do processo de produção que serão usadas, criadas e gerenciadas ao longo da produção.

A atribuição dessas características de referência às operações do plano de fluxo de trabalho, leva a modelagem dos estados de manufatura do produto. A combinação dos estados de produção planejados com definições do produto, específicas a cada operação de trabalho, determina ICDM de referência, do qual serão criadas as instâncias que representam a peça no mundo digital.

Na terceira fase, as peças começam a ser fabricadas. Para isso é atribuída a peça semiacabada um número de identificação individual. Dessa forma cada peça é representada por uma imagem digital individual, que consiste de uma instanciação do ICDM.

As informações reais individuais referentes as características de cada peça, e outras informações relacionadas aos outros modelos parciais do ICDM são adquiridas durante o processo de produção, atualizadas no ICDM e gerenciadas pelo invólucro administrativo. O invólucro administrativo realiza as comparações entre o modelo de referência e as características reais da peça durante a produção, por meio de algoritmos.

A Figura 11 apresenta os modelos parciais que formam a estrutura do ICDM para o planejamento dos processos de manufatura de peças de acordo com Picard [10].

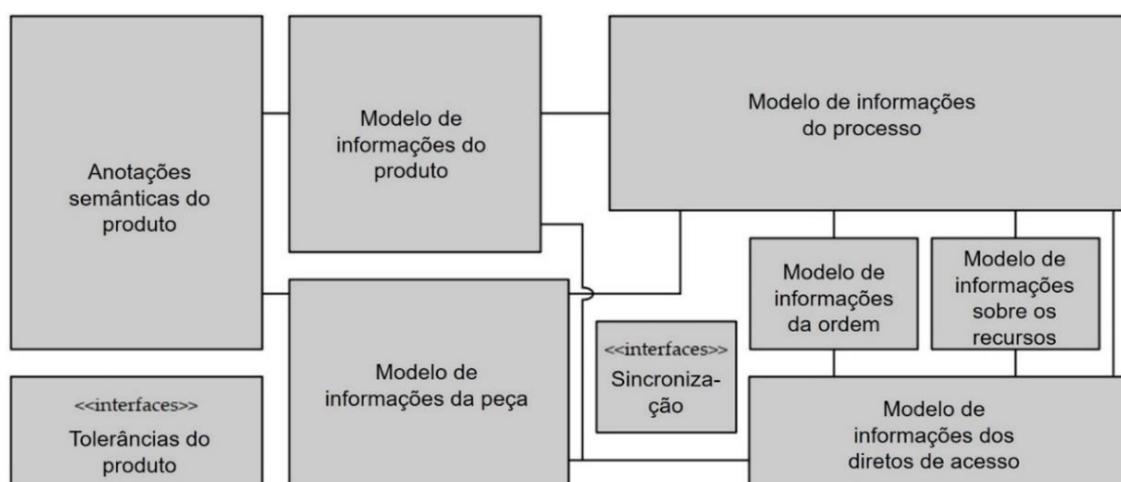


Figura 11: Estrutura do ICDM para manufatura de peças (traduzido de Picard [10]).

O modelo de informações da peça descreve as estruturas de dados que representarão das características da peça individual. As atribuições das características de uma peça incluem dados de especificação do material, descrição da geometria, características de superfície da peça e definição do comportamento da peça ao longo do processo de produção. Outras definições, específicas ao estado da peça durante a produção são incluídas na estrutura de dados como localização da peça, características e *status* do ciclo de vida.

O modelo de informações do produto, especifica a estrutura de dados do modelo 3D do produto e as anotações semânticas sendo utilizado para determinar os estados de manufatura específicos da peça e as características de referência do modelo de referência da peça. Este modelo parcial estende a representação do produto para compreender o *status* do modelo de produto durante a produção e as interfaces de sincronização com os outros modelos parciais. Durante a modelagem, é apresentada a definição do produto.

Ao finalizar o planejamento de produção, o modelo final da peça é congelado e as características da peça a cada etapa do processo são combinadas a sua respectiva etapa do plano de trabalho contido no modelo de processo. A definição das características do produto é específica para cada estado da peça durante o processo, compreendendo a definição do histórico de modelagem, de geometria, tolerâncias e propriedades de superfície. Essas informações dizem respeito ao *Master Prototype* desse trabalho e determinam as características de referência para comparação com as informações reais.

O modelo de informação do processo vincula as características de referência aos processos de trabalho. Ele especifica a representação necessária ao plano de fluxo de trabalho. Este modelo apresenta a sequência de operações discretas do processo de fabricação. Nele são especificados o material inicial a ser usado, as instruções de trabalho necessárias e os recursos de fabricação para converter o início no estado final. Dependendo do progresso do trabalho, os processos de trabalho são divididos em processos de trabalho realizado e planejados. Por meio da combinação do modelo de informações do processo com modelos de

informações do produto, o modelo de referência da peça fornece as definições de dados do produto para a peça individual a cada etapa, como informações de recursos no momento da sua utilização durante a produção.

O modelo de informação representa as interfaces de sincronização relaciona a identificação única que gera as características de referência. As características de referência consistem de uma ligação semântica entre os modelos de informação do produto e da peça em um de modelo referência da peça. As interfaces de sincronização permitem que essas informações se relacionem por meio de TICs.

O modelo de informação de ordens e recursos, possui uma interface para ligar as informações ao planejamento de ordens e recursos. Esse modelo define as interfaces para a conexão com os *softwares* de aplicação relacionados. O ICDM contém estruturas de dados para o mapeamento do cliente, de pedidos e dados de custo, assim como mapeamento de máquinas-ferramenta, equipamentos de produção, materiais e recursos humanos.

O modelo de informação de direitos de acesso, representa os direitos de acesso dos usuários. Esse modelo contém informações dos usuários para especificar direitos de criar, editar e excluir estruturas de dados. Esses direitos podem ser ordenados hierarquicamente, na qual os usuários são atribuídos a grupos de usuários. Os usuários herdam os direitos de acesso dos grupos de usuários e esses direitos podem ser complementados ou substituídos pelas funções do usuário.

2.4. Gêmeo Digital

Em uma apresentação sobre PLM na *University of Michigan*, em 2002, Michael Grieves propôs o conceito de Gêmeo Digital [11,46,47]. Desde então, o conceito esteve inserido no contexto da indústria aeroespacial [11,48]. Recentemente, o conceito se difundiu para outras áreas, sendo objeto de estudo de pesquisadores acadêmicos e corporativos como uma solução para a otimização do ciclos de

vida do produtos agregando valor aos produtos e aos processos das empresas, e permitindo a criação de novos modelos e oportunidades de negócio [11].

Não existe ainda uma definição homogênea para o Gêmeo Digital na comunidade acadêmica. Logo, são encontradas diferentes definições na literatura [47]. O Gêmeo Digital foi definido pela NASA como uma simulação multifísica, multiescalar e integrada, de um veículo ou sistema, que utiliza os melhores modelos físicos disponíveis, atualizações de sensores, entre outros recursos, para espelhar a vida do seu “gêmeo físico” correspondente [47,49,50,51]. Conforme o conceito foi traduzido para outras áreas, o termo veículo foi substituído por produto [48].

Na definição apresentada por Tharma et al [49], o Gêmeo Digital é uma reflexão digital de um produto real em que são descritas as propriedades físicas e funcionais desse produto visando entregar e receber informações ao longo do seu ciclo de vida. Essa definição se assemelha a de Stark et. al. [11, p. 1] que define o Gêmeo Digital como “uma representação digital de um produto individual (dispositivo real, objeto, máquina, serviço ou recurso intangível) ou um sistema produto-serviço individual (sistema que consiste de uma produto e serviço relacionado) que compreende características, propriedades, condições e comportamentos relacionados por meio de modelos, informação e dados, em uma única fase, ou ao longo de várias fases do ciclo de vida”.

Grievess et al [52] explica que Gêmeo Digital está inserido em um ambiente virtual e um ambiente físico, e que existe uma interface de relação entre eles, na qual dados e informações são trocados. Essa definição de estrutura do Gêmeo Digital é tida como base para a maioria das pesquisas sobre o assunto [46,38].

Para Stark et al [11] o Gêmeo Digital consiste de uma combinação da sombra digital individual de um produto, com os modelos de descrição do produto (como modelos geométricos e de simulação). O conceito é representado na Figura 12. Durante processo de planejamento e desenvolvimento do produto, informações e modelos de descrição do produto são gerados e armazenados em um *Digital*

Master, protótipo digital derivado do processo (1). No contexto desse trabalho, denominado *Master Prototype*.

O Gêmeo Digital consiste da combinação de uma instância única de um *Master Prototype*, adaptado a um propósito específico, de uma sombra digital de um produto, que descreve o estado real de um produto físico individual (2), e de uma ligação significativa entre esses elementos por meio de algoritmos, modelos de simulação, correlações, etc. (3) [11,6].

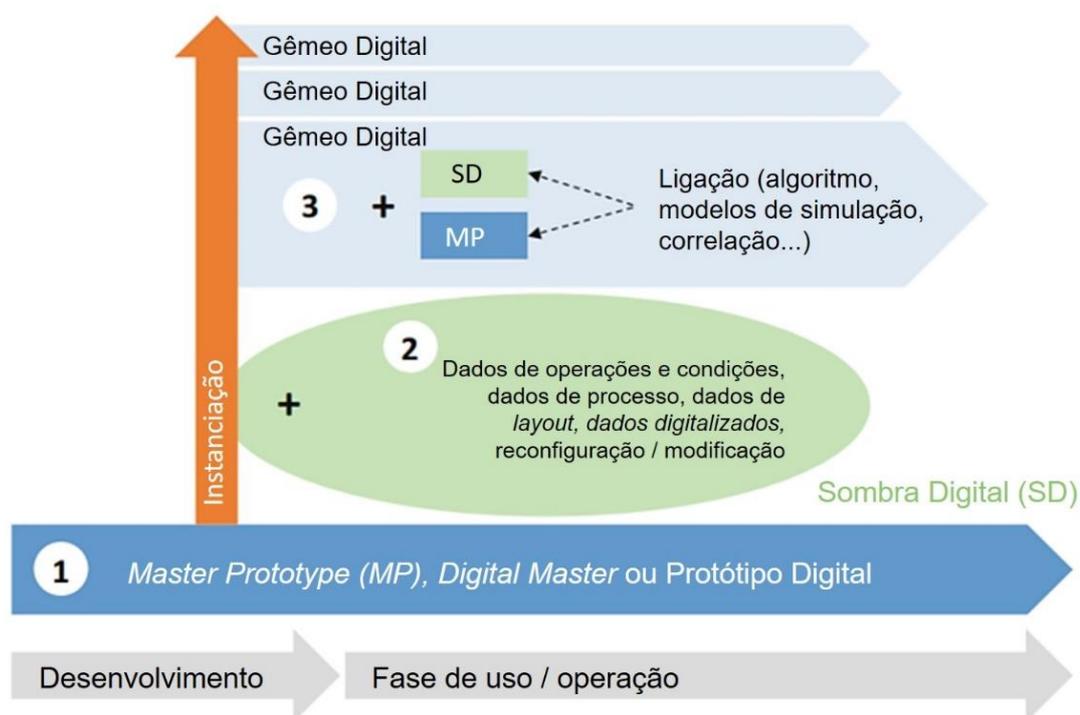


Figura 12: Elementos do Gêmeo Digital (adaptado de Stark e Demerau [11]).

2.4.1. Ciclo de vida do Gêmeo Digital

Como mencionado no Capítulo 2.2 deste trabalho, as fases do ciclo de vida de um produto são divididas, em início da vida (BOL), meio da vida (MOL) e fim da vida (EOF).

Contudo, visto que todo Gêmeo Digital de um produto tem um gêmeo físico durante o ciclo de vida, tornou-se necessário realizar uma distinção entre os ciclo

de vida do produto digital e físico [50]. Tharma et al. [49], Stark et al [11] e Abramovici et al. [50] descrevem as fases do ciclo de vida digital e real dos produtos de forma separada em que os fluxos de informações ao longo das fases dos ciclos de vida são contínuos.

De acordo com Abramovici et al [50] as fases de desenvolvimento do produto, planejamento de produção, gestão do uso do produto e planejamento final correspondem as fases do ciclo de vida digital dos produtos inteligentes (Figura 13). O fluxo de informação é combinado, respectivamente as informações geradas nas fases de prototipagem, produção, uso do produto e fim da vida, correspondendo as fases do ciclo de vida físico do produto inteligente.

Para Tharma et al. [49], os ciclos de vida do produto são divididos em três camadas. O mundo real contém o produto físico e o mundo digital é composto pelo modelo digital do produto, submetido a otimizações ao longo do ciclo de vida, e pelo Gêmeo Digital, que age como intermediário na representação das mudanças que são realizadas nas duas camadas, ao longo das fases do ciclo de vida do produto.

Neste cenário, ciclo de vida do produto compreende as fases de “como-projetado”, “como-fabricado” e “como-mantido”. Na primeira fase, somente os modelos digitais e os protótipos físicos existem, cooperando para encontrar uma solução ótima. A partir do momento em que é realizada a melhor versão do protótipo, o modelo digital é atualizado para esta versão mais otimizada, e pode ser liberado para a produção. Depois do processo de produção, o Gêmeo Digital é criado.

Na abordagem de Stark et al. [11], representada na Figura 14, o ciclo de vida do Gêmeo Digital é realizado em quatro fases. Na fase de Iniciação, busca-se compreender a função que o Gêmeo Digital tem para os negócios. As necessidades e oportunidades em relação a operação do Gêmeo Digital são identificadas e traduzidas em requisitos. Estes são então analisados e

detalhados em estreita colaboração com o produto ou serviço operacional pretendido.

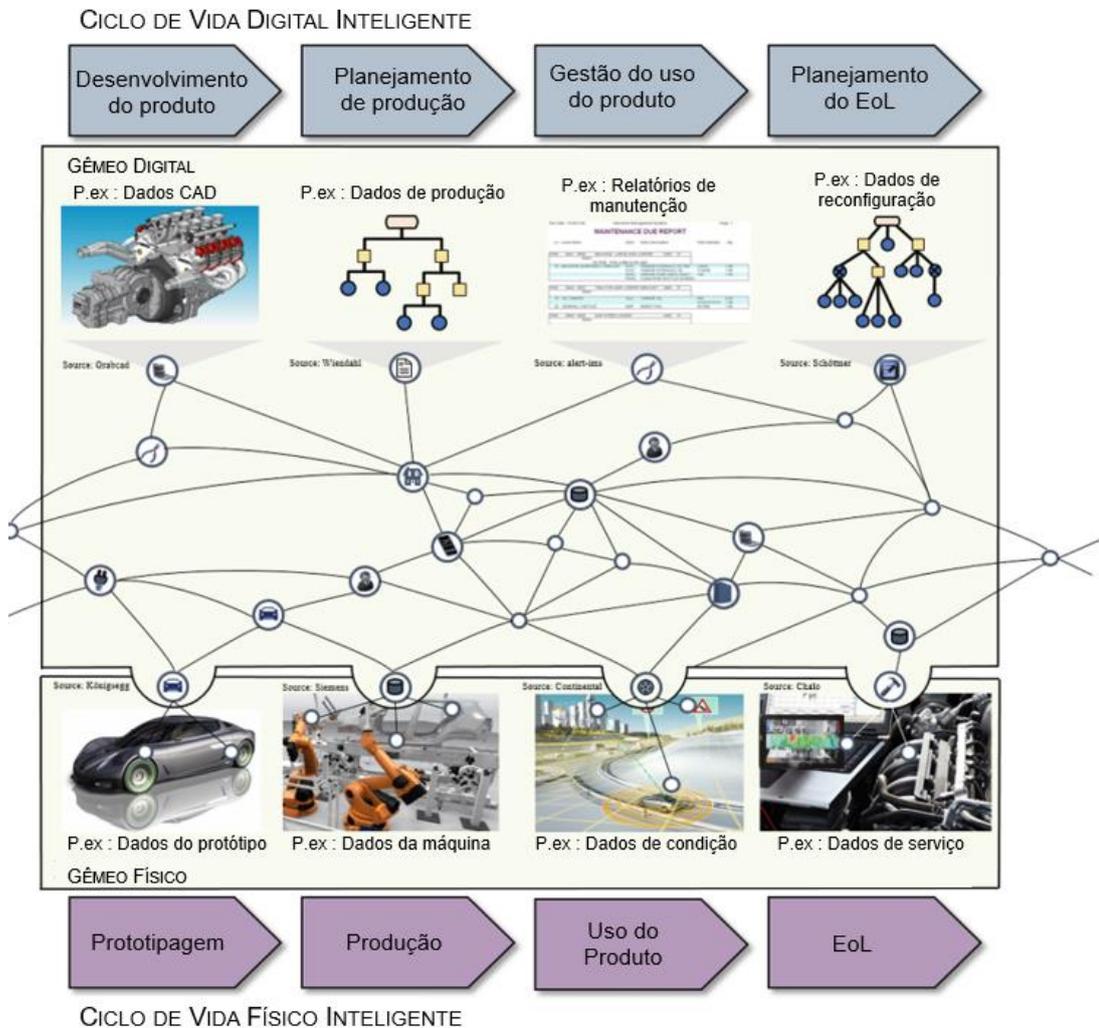


Figura 13: Informações trocadas nas fases do ciclo de vida digital e físico do produto inteligente (traduzido de Abramovici et. al. [50])

O ciclo de vida tem continuidade com a identificação e o desenvolvimento das estruturas e dos elementos de projeto apropriados ao Gêmeo Digital. Para que o Gêmeo Digital atenda aos objetivos para o qual foi planejado, deverá compreender os modelos digitais e simulações do produto realizados durante o desenvolvimento de produtos. Modelos adicionais deverão ser desenvolvidos a fim de determinar as técnicas de aquisição de dados e informações que o Gêmeo Digital deverá utilizar. Para esta abordagem, o Gêmeo Digital pode ser

concretizado na etapa de desenvolvimento do produto, durante a prototipagem física, ou posteriormente com a geração de instâncias individuais do produto, durante a fase de uso do produto [11].

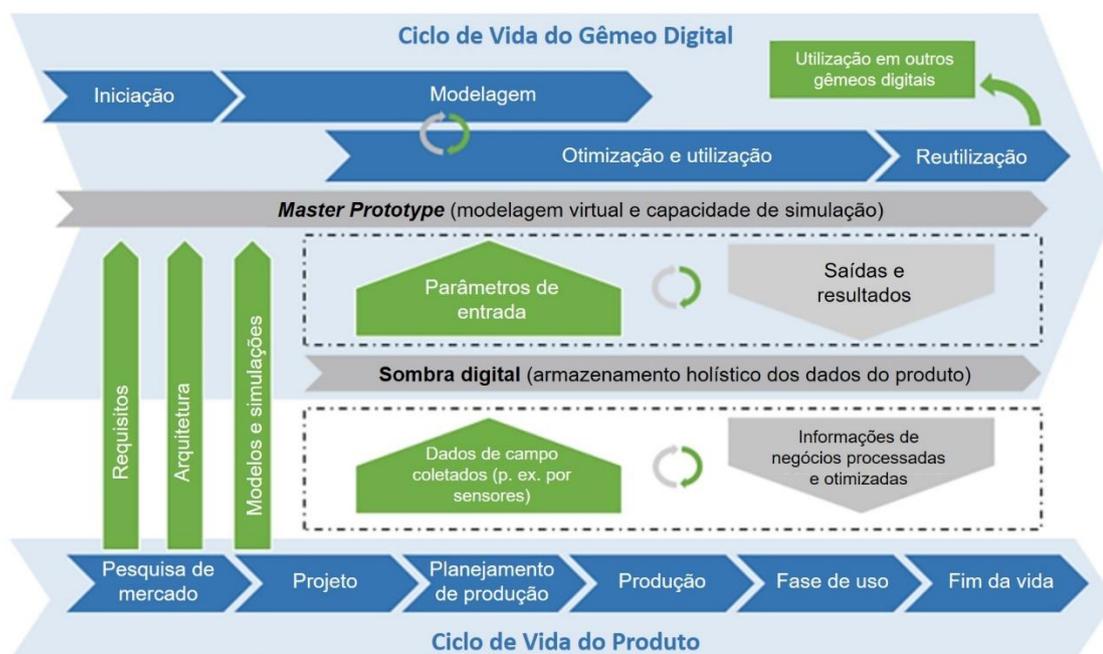


Figura 14: Ciclo de vida do Gêmeo Digital (adaptado de Stark e Demerau [11])

Boschert et. al. [53] afirma que o Gêmeo Digital inclui informações dados relevantes para o processamento de tarefas específicas de simulação. Ele contém somente as informações relevantes que são requeridas para ter se êxito nas fases subsequentes. Por isso é importante que o propósito do Gêmeo Digital seja descrito nas etapas iniciais do ciclo de vida. A partir desses objetivos as tarefas são definidas para que posteriormente respondam as questões e atendam aos objetivos estabelecidos. Isto leva a definição da estrutura de dados necessários e dos modelos de simulação, sendo construída a arquitetura do Gêmeo Digital para uma aplicação concreta e para um conjunto de propósitos.

De acordo com Abramovici et. al. [50], a etapa de produção está relacionada a fase do ciclo de vida digital de planejamento da produção. Logo, é importante que os resultados do processo de desenvolvimento do produto estejam relacionados aos resultados do planejamento de produção.

Não é exigido que para ser considerado um Gêmeo Digital o produto seja completamente inteligente. Mesmo para produtos simples, como componentes individuais, o Gêmeo digital pode ser implementado. Eles necessitam, porém, de capacidades de conexão IoT e sensores [11].

Os dados de produtos ou componentes simples podem ser parte integrante de um sistema maior. Por exemplo, o Gêmeo Digital pode contribuir com o monitoramento do desempenho de produção, garantindo que os recursos sejam alocados para as máquinas adequadas, possibilitando uma produção flexível e auxiliando no controle de qualidade, como otimizar a verificação da conformidade das folgas projetadas com as folgas do produto fabricado. Para isso, os dados do modelo do produto são colocados em contexto com os dados da máquina, ou seja, com os dados da linha de produção, possibilitando a otimização da produção e a rastreabilidade de um componente na produção muito além do número de lote [11,50]

2.4.2. Master Prototype

Apesar dos principais benefícios de um Gêmeo Digital estarem relacionados as etapas de operação e uso dos produtos, ele diz respeito a um produto que precisa ser planejado nas fases iniciais do seu ciclo de vida, etapa na qual as características e requisitos do Gêmeo Digital devem ser definidos [53]. Portanto, é necessário determinar as principais características do *Master Prototype* assim como do seu processo de desenvolvimento.

O *Master Prototype* consiste de um conjunto de informações digitais em que está contida a definição do produto [13]. Este conjunto de informações é tido como referência, para os processos e operações que serão realizados pelo Gêmeo Digital nas fases subsequentes do ciclo de vida. Com a realização do produto no mundo físico, uma instância de *Master Prototype*, específica do produto fabricado é gerada e combinada aos dados reais fornecidos pela sua sombra digital [11].

A *Master Prototype* é a classe abstrata que cobre todas as características que as instâncias têm em comum e, portanto, não representam um Gêmeo Digital completamente funcional. Enquanto que as instâncias fazem parte de um Gêmeo Digital completamente funcional e definem características que não são compartilhadas com outras instâncias [13,54].

As instâncias geradas a partir do *Master Prototype* representam completamente as entidades físicas no mundo real, possibilitando ao Gêmeo Digital simular os comportamentos e o desempenho do produto com fidelidade e em tempo real [55].

As informações que geralmente compõem do *Master Prototype*, de acordo com a literatura, são: modelos CAD, modelos de simulação, PMI, BOM [11,52,13,56]. Porém, os dados e modelos digitais que compõem o *Master Prototype* podem ser de variados tipos [57], como informações sobre requisitos [52], descrição do processo de criação [50], versões de *software* [13], metadados [58], etc.

Um bom funcionamento de sistemas PLM é requerido para a realização do Gêmeo Digital, visto que esses sistemas gerenciam dados de definição do produto, processos, recursos e apoiam tomadas de decisão do produto fornecendo suporte as mudanças e ao rastreamento das mudanças ao longo do ciclo de vida [14,6].

Ou seja, os sistemas PLM gerenciam as informações do *Master Prototype* e são associados aos sistemas CAx, contribuindo com o processo de desenvolvimento do produto, proporcionando um trabalho colaborativo nas etapas de criação, gerenciamento, disseminação e utilização das informações do produto ao longo do ciclo de vida [13].

No trabalho de Abramovici et al. [13] foi desenvolvida e implementada uma abordagem conceitual para a reconfiguração de produtos inteligentes durante a fase de uso. Nesta abordagem, todos os dados gerados durante os ciclos de vida do produto digital e do produto físico são armazenados em um banco de dados do Gêmeo Digital. Esse banco de dados armazena modelos

interdisciplinares (dos domínios da mecânica, eletrônica e informática) que são sincronizados com seus respectivos produtos físicos podendo alimentar o banco de dados com informação útil ou reconfigurar o produto físico.

Na abordagem foi criada uma plataforma que permite a integração de todos os dados relevantes das instâncias do produto. Foram consideradas também as informações sobre os componentes desenvolvidos e fabricados por outras organizações. Para isso do banco de dados do Gêmeo Digital possui uma interface com os sistemas PLM e ERP – sendo considerados a base a manufatura digital – da organização, assim como interfaces com os sistemas de parceiros envolvidos no desenvolvimento e produção do produto [13,59].

O *Master Prototype* contém todas as informações necessárias as partes interessadas resultando um grande volume de informações armazenadas. Contudo, o *Master Prototype* não consiste somente no armazenamento de todos os artefatos digitais (modelos e dados) relacionados ao produto, estes artefatos devem ser estruturados e os elementos devem poder ser conectados, requerendo metadados e contendo semântica [57,58].

Schleich et al. [12] apresenta um modelo de referência para representação do Gêmeo Digital e explica algumas propriedades que devem ser consideradas em sua representação. São elas: escalabilidade, interoperabilidade, expansibilidade e fidelidade. Por escalabilidade, entende-se a capacidade de fornecer informações em diferentes escalas. A interoperabilidade é a aptidão em converter combinar e estabelecer equivalência entre diferentes modelos. A expansibilidade se refere capacidade de integrar adicionar e substituir modelos. A fidelidade, refere-se à descrição mais próxima possível do produto físico [14,12].

Boschert et al. [57] apresenta uma abordagem para o desenvolvimento do Gêmeo Digital, denominada *Next Generation Digital Twin*. A abordagem compreende as fases de desenvolvimento e uso do Gêmeo Digital. Na abordagem, o Gêmeo Digital é desenvolvido com o objetivo de utilizar modelos

de simulação de comportamento do produto, visando combinar com dados reais do produto coletados durante uso para identificar para assim causa de falhas.

No início da fase de desenvolvimento, é criado um *template*. Este *template* deve conter uma descrição inicial da estrutura do produto contendo os relacionamentos e interações entre diferentes componentes. Ao longo do processo de desenvolvimento do produto, esse *template* é preenchido com submodelos. Esses submodelos são conectados e sincronizados ao longo dos processos do ciclo de vida. Além disso diferentes Gêmeos Digitais são conectados (Gêmeos Digitais dos componentes, do produto, do sistema de produção, do produto em uso, etc.) no intuito de explorar os diferentes dados relevantes para cada fase. Essas conexões são realizadas a partir de tecnologias semânticas.

3. Método

Nesse Capítulo será apresentada uma descrição do método usado e a descrição das etapas realizadas para o desenvolvimento deste trabalho.

3.1. *Design Science Research*

O *Design Science Research*, ou DSR, é um método que amadureceu principalmente na área de tecnologia e gestão da informação. Busca conciliar a teoria e prática ao mesmo tempo em que busca contribuir de forma prática criando soluções para problemas reais e gerar conhecimento para o avanço nas pesquisas com base em conhecimento multidisciplinar [60].

Esse tipo de pesquisa, busca resolver problemas complexos relevantes de forma satisfatória, levando em consideração em que contexto seus resultados serão aplicados. Tem como característica o rigor no processo ao planejar os artefatos, uma vez que eles são o resultado prático do processo [60].

Para a realização do método é necessário que, inicialmente, seja realizado um levantamento inicial do problema. Essa etapa é denominada conscientização [60] (Figura 15) ou explicação do problema (Figura 16) [61]. Para isso, realiza-se uma revisão sistemática de literatura, visando identificar resultados empíricos sobre o assunto.

Essa etapa é importante para consolidar a classe de problemas que, uma vez definida, permite caracterizar os artefatos associados. Dessa forma, “artefatos são uma organização dos componentes do ambiente interno a fim atingir objetivos de um determinado ambiente externo” [60, p. 748].

Uma classe de problemas consiste em um conjunto de problemas organizados, podendo ser práticos ou teóricos e que são úteis para a pesquisa [60]. É importante que o problema seja de interesse geral [61]. Portanto, é necessário definir a classe de problemas porque, apesar do problema real e dos artefatos

serem singulares, a definição da classe de problemas permite, ao final do processo, a generalização do conhecimento [60].

Depois de ser realizada a conscientização, por meio da definição, formalização, e delimitação do problema e das soluções satisfatórias necessárias, a próxima etapa é a sugestão. Na sugestão são desenvolvidas uma ou mais soluções para os problemas. Ela consiste de um processo criativo, portanto considerado relativamente subjetivo e difícil de ser padronizado. Na busca por soluções, para o caso que a implementação de uma solução ótima seja inacessível ou inviável, busca-se uma solução ou soluções suficientemente satisfatórias [60].

Johanneson et al [61] apresenta esta etapa como definição de requisitos e assim como Lacerda et al [60] recomenda que a solução seja delimitada de maneira precisa por meio de critérios de aceitação bem justificados. Os requisitos devem definir não somente a funcionalidade, mas também a estrutura e o contexto (ambiente externo) dos artefatos. O resultado da sugestão é a escolha de um, ou mais artefatos, a serem desenvolvidos. Esta etapa tem como saída uma proposta de artefato [60] junto aos requisitos [61].

Os artefatos podem ser classificados em: constructos, modelos, métodos e instanciações. Os constructos são os conceitos que descrevem os problemas em determinado domínio para especificar interrelações. Modelos são uma descrição da realidade visando a otimização, portanto, é um conjunto de declarações que estabelecem as relações entre os constructos [60].

Um método consiste de um conjunto de passos utilizados para a realização de uma tarefa. Ele se baseia em um conjunto de constructos e uma representação de um modelo em um espaço de solução. É uma criação típica do DSR. Por fim, as instanciações são a concretização do artefato em seu ambiente [60].

Na etapa de planejamento e desenvolvimento, o artefato é constituído em si. O ambiente interno do artefato é construído pelo pesquisador. O objetivo desta etapa é gerar conhecimento aplicável e útil para solucionar problemas “O

principal resultado do desenvolvimento é o artefato em estado funcional” [60, p. 750].

O termo “desenvolvimento”, no contexto da DSR, não se refere somente ao desenvolvimento de um produto, mas pode se referir também a criação de conhecimentos aplicáveis e úteis por meio da criação de soluções. Esse conhecimento, denominado conhecimento útil, isto porque apesar de ser gerado para uma solução de problemas pontuais ou desenvolvimento de artefatos novos, é generalizável a definição da classe de problemas [60,61].

A etapa de demonstração foi proposta por Johanneson et al [61], como uma atividade na qual o artefato é colocado à prova em um caso ilustrativo ou real. A intenção é testar a viabilidade do artefato. A demonstração determinará se o artefato é capaz de solucionar o problema proposto.

Para Lacerda et al [60], a etapa de avaliação compreende a verificação do comportamento do artefato, no ambiente para o qual ele foi projetado, verificando a conformidade com as soluções que pretendeu alcançar. Esse procedimento compreende várias etapas rigorosas para verificação do desempenho do artefato. De acordo com Johannesson et al [61], na avaliação é verificado o êxito no atendimento aos requisitos e até que ponto o artefato foi capaz de resolver ou mitigar o problema proposto.

Se durante as etapas de desenvolvimento e avaliação, o artefato não atendeu aos requisitos de pesquisa, o pesquisador pode retornar a etapa de conscientização para entender o problema e continuar a pesquisa. Esse são denominados ciclo de projeto e são essenciais para a melhor compreensão da pesquisa e conhecimento a partir do esclarecimento do processo de construção do artefato (setas na Figura 15). Com esse retorno, é possível que sejam identificadas lacunas na teoria, ou até mesmo o artefato construído pode ter se mostrado inadequado para resolver o problema esclarecido no momento da conscientização [60].

Na etapa de conclusão o pesquisador apresenta os resultados e verifica se a conscientização do problema está completa ou não. Caso não esteja, as lacunas na teoria são identificadas e apresentadas. O procedimento de Johannesson et al [61] é finalizado nesta etapa, resultando no artefato avaliado. Como passo final deve ocorrer a formalização geral do processo e disseminação do conhecimento as partes acadêmicas e profissionais [60].

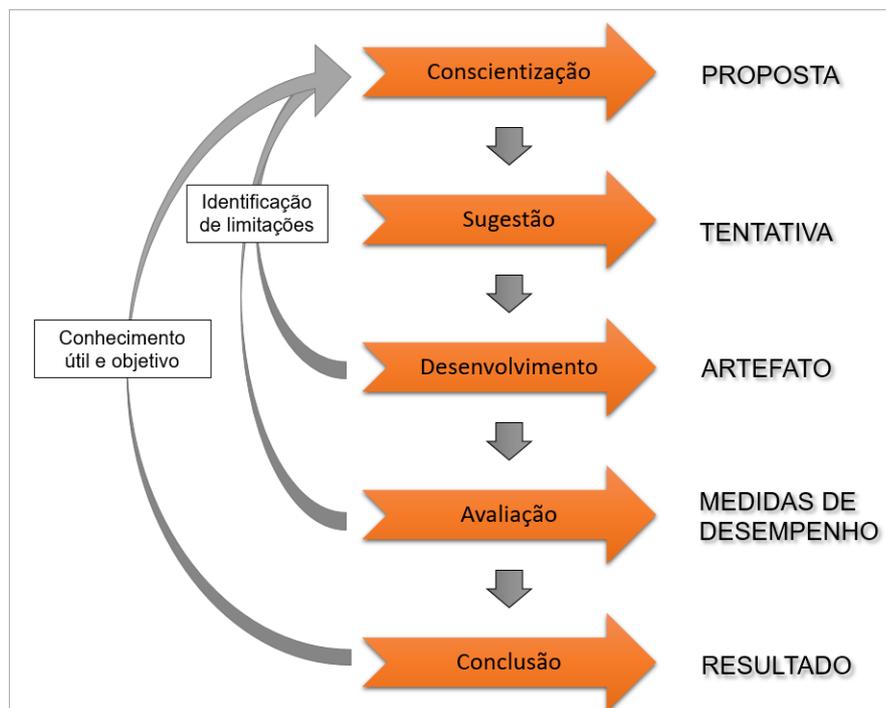


Figura 15: Esquema de condução do DSR (adaptado de Lacerda et. al. [60]).

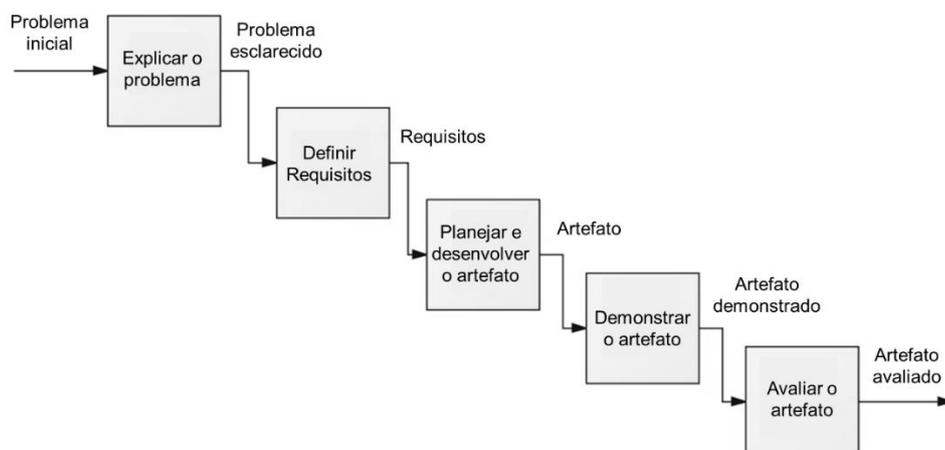


Figura 16: Visão geral do *framework* para DSR (traduzido de Johannesson e Perjons [61]).

3.2. Etapas da pesquisa

Esta pesquisa se enquadra na classe de pesquisa qualitativa e de natureza exploratória. As etapas do método deste trabalho têm como base os procedimentos de Lacerda et al [60] e Johanneson et al [61], conforme a Figura 17 apresenta.

Primeira etapa: Conscientização

A primeira etapa consistiu de uma revisão de literatura, visando compreender melhor o problema de pesquisa. A pesquisa foi realizada nas bases de dados *SCOPUS*, *Science Direct*, *Web of Science* e *Google Scholar*. Tendo o DSR como método de pesquisa e devido sua característica sistemática, o processo consistiu de várias iterações de busca e coleta de informações.

Foram pesquisadas nas bases de dados informações a respeito do Gêmeo Digital, Indústria 4.0, Protótipo Digital e *Digital Master*, combinadas a expressão Indústria 4.0.

Como fonte de informações principal sobre a Indústria 4.0, os documentos da Plataforma I4.0 também foram explorados. Isto permitiu a definição do escopo do trabalho em um nível mais concreto, além de impulsionar a busca por soluções, referentes a próxima etapa.

Segunda etapa: Sugestão

Esta etapa consistiu na busca por soluções, determinação do artefato e delimitação de requisitos. Foi determinado o desenvolvimento de um *framework*. O *framework* se encaixa na categoria de artefato método, significando um conjunto de métodos estruturados que direcionam o desenvolvimento de um *Master Prototype* em um sistema PLM, de acordo com o modelo ICDM, considerado adequado ao objetivo.

O *framework* deveria atender aos requisitos de compreender todas as informações do *Master Prototype*, apresentadas por Picard [10] e Schleich et. al. [12] além de possuir as características apresentadas no tópico 2.4.2.

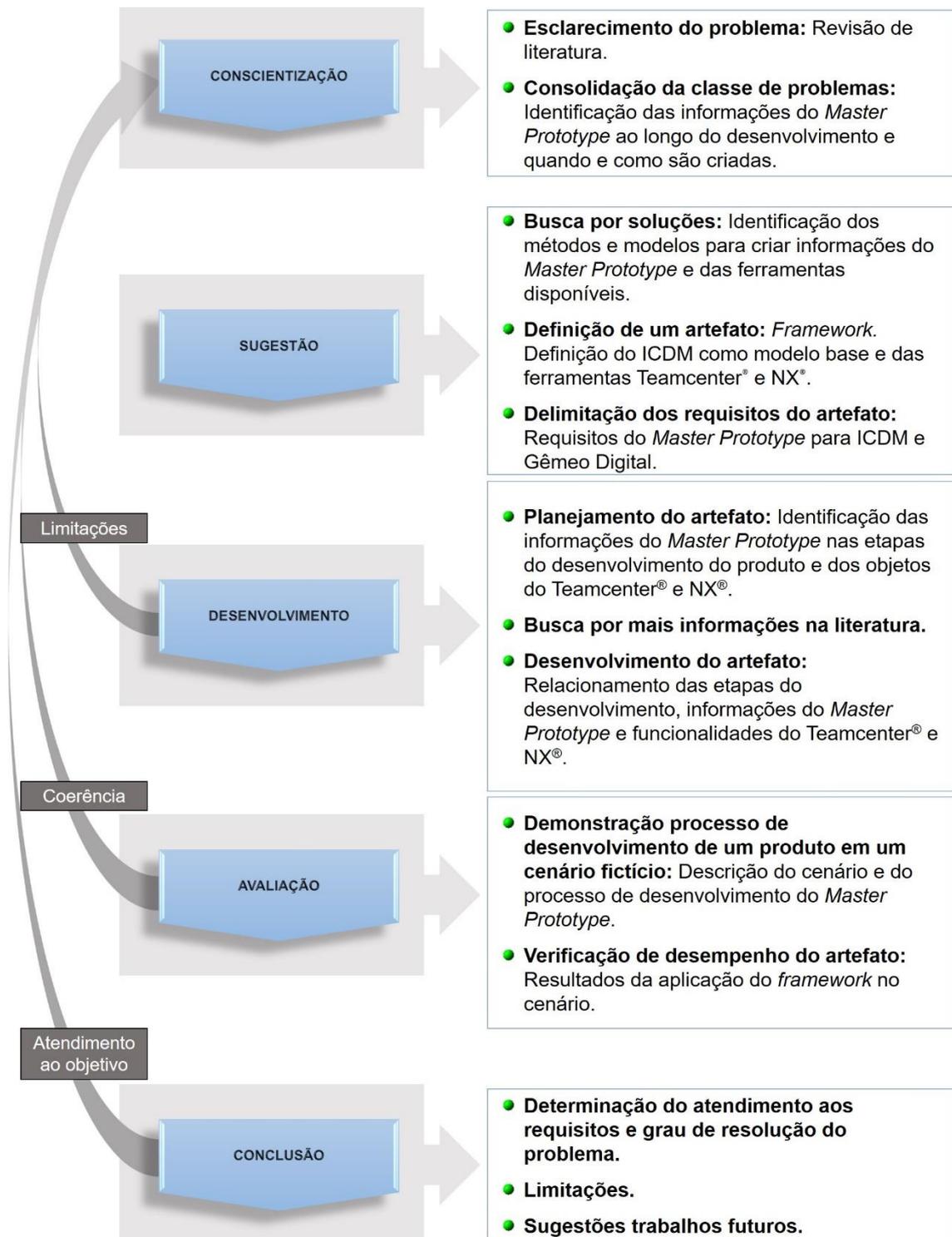


Figura 17: Método para o desenvolvimento do trabalho.

Terceira etapa: Desenvolvimento

Para a realização do framework, foi necessário identificar quais as ferramentas disponíveis para a construção do ambiente interno e posterior verificação. Foi definido que seriam usados os *softwares* PLM Teamcenter® e NX®, para a verificação do *framework* visto que são as ferramentas disponíveis no Laboratório de Sistemas Computacionais para Projeto e Manufatura da UNIMEP.

No início do desenvolvimento do artefato, foram identificadas as etapas e atividades do processo de desenvolvimento do produto em que são geradas as informações do *Master Prototype*.

Buscou-se na conscientização, realizada na primeira etapa, quais seriam as funcionalidades e objetos que atenderiam as necessidades dessas informações. Como resultado, mostrou-se necessário a busca por fontes externas. Além disso, demonstrou-se essencial explorar as funcionalidades dos *softwares* Teamcenter® e NX® na prática contando com o apoio de tutoriais da Siemens®.

Completada essa iteração com a conscientização, conciliando a prática a fim de fechar a lacuna encontrada na teoria, foi possível identificar objetos e funcionalidades essenciais para a construção do *framework* completando assim o planejamento do artefato.

O desenvolvimento do *framework* consistiu em identificar quais dos objetos e funcionalidades explorados seriam essenciais para atender aos requisitos determinados na sugestão. A síntese dessas informações resultou na estrutura de um *framework* para desenvolvimento do *Master Prototype*.

Quarta etapa: Avaliação

A avaliação foi realizada por meio de uma verificação do *framework* aplicado em um cenário fictício. O cenário foi representado por meio de um diagrama UML de casos de uso, em que são apresentadas as funcionalidades, atores e limites do sistema Teamcenter® necessários para o desenvolvimento do *Master Prototype*. A longo do processo buscou-se verificar a coerência com a teoria.

Os métodos usados para a criação das informações foram explicitados conforme foram realizadas as etapas de desenvolvimento do produto. A identificação da ordem de utilização das funcionalidades foi apresentada numericamente de acordo com a visão geral do *framework*, enquanto que, para o detalhamento das principais funcionalidades principais e operações no sistema, foram usados diagramas de sequência.

Quinta etapa: Conclusão

Essa etapa consistiu da avaliação quanto ao grau de atendimento aos requisitos determinados a partir da conceituação. Foram explicitadas também as limitações que foram encontradas e apresentadas sugestões para trabalhos futuros.

4. Proposta para o desenvolvimento de um *Master Prototype*

No Capítulo 2 foi apresentada uma revisão de literatura abordando os constructos dessa dissertação. São eles:

- Processo de desenvolvimento do produto: No tópico 2.1 foram apresentadas as principais etapas do processo, tendo como base a abordagem sistemática para desenvolvimento do produto proposta por Pahl e Beitz [1].
- Gestão do ciclo de vida do produto: No tópico 2.2 foi apresentada a abordagem PLM para gestão de dados do produto, por meio das funcionalidades principais dos sistemas PLM e tópicos relacionados a digitalização do processo de desenvolvimento do produto. As principais referências usadas foram Zancul [5] e Abramovici et. al. [20].
- Indústria 4.0: No tópico 2.3 apresentou os principais elementos, princípios e abordagens da Indústria 4.0. Foi apresentado também o modelo ICDM, focando na definição do *Master Prototype* do produto, baseando-se, principalmente, no trabalho do Picard [10].
- Gêmeo digital: No tópico 2.4 foram descritas as principais características do conceito e a abordagem do *Master Prototype* de acordo com Stark et. al. [11] e Schleich et. al. [12].

Buscou-se explorar o problema de pesquisa que consiste em identificar características das informações do *Master Prototype* e, partir disso, compreender como essas informações podem ser geradas e gerenciadas ao longo do processo de desenvolvimento do produto.

Em vista disso, é apresentada a seguinte proposta: a partir da abordagem sistemática de processo de desenvolvimento do produto, das atividades relacionadas a cada etapa – planejamento e esclarecimento da tarefa, projeto

conceitual, projeto preliminar e projeto detalhado – e das informações necessárias para o modelo de *Master Prototype*, chega-se a um *framework* que apresenta uma estrutura para o desenvolvimento de um *Master Prototype* do produto. Para isso será utilizada a abordagem PLM, uma vez que se mostra adequada à realização do processo.

O *framework* deve buscar atender aos seguintes requisitos:

- Permitir gerar todas as informações essenciais ao *Master Prototype* de acordo com o modelo de dados ICDM apresentado no tópico 2.3.2.
- Atender aos requisitos do Gêmeo Digital ressaltados no tópico 2.4.2.

4.1. Informações ao longo do processo de desenvolvimento do produto

Como foi apresentado nos tópicos 2.3.2 e 2.4.2 o *Master Prototype* corresponde ao modelo parcial de especificação do produto, podendo abranger também o modelo de gerenciamento de acesso. A estrutura de informações de especificação do produto do ICDM se baseiam no modelo STEP. Essas informações incluem os requisitos do produto, modelos do produto, para representações geométricas e simulações de comportamento, informações para manufatura do produto e representação da estrutura do produto, compreendendo as informações administrativas e organizacionais.

Durante o processo de desenvolvimento do produto essas informações são criadas e definidas gradualmente, conforme apresentado no tópico 2.1. A Figura 18 apresenta um esquema das principais atividades realizadas e das informações geradas durante o processo.

No início do processo de desenvolvimento são determinadas as especificações iniciais do produto, estruturadas em uma lista de requisitos. Os requisitos são avaliados quanto ao esclarecimento de todos os aspectos que envolvem o ciclo de vida do produto para que possam ser transmitidos para a etapa de projeto conceitual.

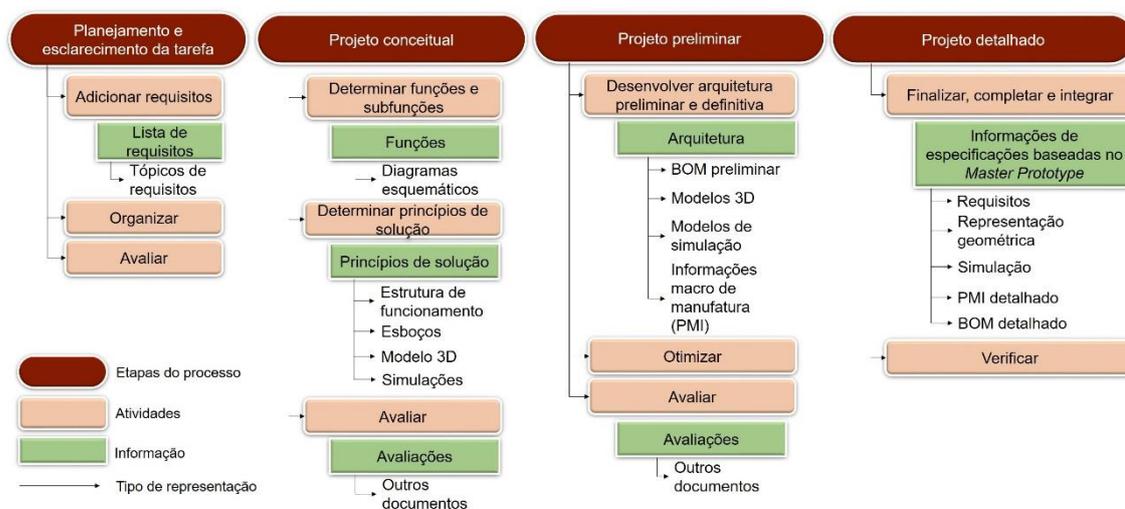


Figura 18: Representação das principais etapas e tipos de informações gerados no desenvolvimento do *Master Prototype*.

Durante a etapa de projeto conceitual, a partir da abstração do problema geral e da lista de requisitos, o modelo do produto começa a tomar forma por meio da definição da função global e identificação de subfunções para essa função, em uma estrutura funcional. Essa estrutura pode ser representada por meio de diagramas esquemáticos. Ela é a base para a busca dos princípios de funcionamento e para a determinação da estrutura de funcionamento, que consiste da combinação desses princípios. Essa combinação determina os princípios de solução que serão avaliados nos critérios técnicos e econômicos no final da etapa.

Os princípios de solução são definidos, de forma simplificada, quanto aos efeitos físicos, características geométricas, de movimento e materiais. Essas informações passam por um procedimento de desenvolvimento visando firmar o conceito. Os resultados podem ser apresentados em esboços, cálculos ou modelos tridimensionais e simulações básicas de comportamento. Por fim são realizadas as avaliações para que os princípios de solução possam seguir para a etapa de projeto preliminar mais esclarecidos quanto aos seus pontos fortes e fracos.

No projeto preliminar são identificados os requisitos determinantes de geometria, arquitetura e de restrições espaciais do produto. Essas restrições podem ser identificadas por meio de métodos como os desenhos em escala, modelos tridimensionais, análises de colisão entre peças, entre outros. Os princípios de solução são então desenvolvidos a partir das partes do produto que representam as funções principais seguindo para partes portadoras de funções auxiliares.

A partir da modelagem e simulação da geometria e comportamento do produto são definidos novos parâmetros que caracterizam o produto em um nível mais alto de concretização. Essas atividades definem geometria, materiais e comportamento do produto. É definida uma arquitetura preliminar para o produto. Essa arquitetura passa por um processo de otimização em que é determinada uma arquitetura definitiva. Por fim são definidos parâmetros de manufatura e montagem (PMI) e a lista de materiais (BOM) preliminares.

No projeto detalhado, a arquitetura definitiva e as informações adicionais são finalizadas, completadas, integradas e verificadas. Essas informações do produto e metadados gerados durante o processo resultam nas informações que compõem o *Master Prototype* do produto proposto neste trabalho.

A partir da descrição das etapas em que são criadas as principais informações do produto, podem ser destacadas as principais características das informações que compõem o *Master Prototype* proposto neste trabalho, assim como quando são geralmente criadas.

- Os requisitos do produto são determinados na etapa de planejamento e esclarecimento da tarefa e consumidos ao longo do processo.
- A representação da estrutura do produto começa a ser definida no projeto conceitual, com a determinação da estrutura dos princípios de solução. A partir dessa estrutura é definida a arquitetura do produto, compreendendo informações adicionais de geometria e comportamento. Da arquitetura é gerada uma lista de materiais (BOM), também denominada EBOM. Ela reúne as informações de definição do

produto que consistem nos documentos de especificação como dimensões, tolerâncias, especificações de materiais e modelos tridimensionais de cada uma das partes e do produto final.

- Tendo como ponto de partida a representação geométrica, uma definição inicial das características geométricas e tipos de materiais surgem na etapa de projeto conceitual. Essas características determinam os parâmetros usados na construção de modelos de representação geométrica do produto. No início do projeto preliminar as representações geométricas em uma estrutura podem ser usadas para a identificação de restrições espaciais. A partir da análise das restrições espaciais, os modelos de geometria são otimizados levando a parâmetros mais adequados.
- Por meio de uma análise funcional da arquitetura do produto são realizadas simulações de modelos de comportamento do produto, por exemplo de elementos finitos para representar força, tensão, deformação, e cinemática para representar movimento e dinâmica, entre outras.
- Depois que o modelo do produto é definido, a representação geométrica é acrescida de informações para manufatura no projeto detalhado. Essas informações podem ser apresentadas em desenhos técnicos ou modelos tridimensionais gerados em sistemas CAD. Elas são parâmetros para a realização das atividades de planejamento da produção do produto.
- As informações administrativas e organizacionais dizem respeito a estrutura do produto e permitem identificar e classificar individualmente um produto e suas partes. A identificação de um produto pode ser realizada através de um sistema de numeração e através da determinação das propriedades específicas de cada parte. Elas compreendem também a classificação do produto nas fases do ciclo de

vida (por exemplo, via *status* de liberação, *status* de revisão e versão), e na estrutura do produto (por exemplo, alternativa, variante) [3]. Essas informações podem ser criadas e manipuladas durante todo o ciclo de vida do produto com o apoio de sistemas de gestão de dados do produto e/ ou sistemas PLM.

- O modelo de direitos de acesso representa os usuários do sistema que contém as informações do produto e seus respectivos direitos de acesso para criar e alterar informações. Esse modelo especifica uma hierarquia de usuários, grupos e responsabilidades, que herdam direitos de acesso dos grupos que pertencem ao longo dos processos do ciclo de vida.

A partir dos aspectos das informações salientados, serão identificados os objetos e funcionalidades do PLM que permitem o desenvolvimento do *Master Prototype* do produto.

4.2. Modelagem do produto e PMI

A modelagem da geometria do produto é realizada por meio dos sistemas de criação CAD, e seus modeladores geométricos. O modelador geométrico do NX® é denominado Parasolid®.

O modelador geométrico é o núcleo dos sistemas CAD. Por meio deles, o sistema efetua cálculos sofisticados para realizar as operações de criação e visualização do modelo [2]. Esse núcleo do sistema é uma biblioteca de funções matemáticas centrais que define e armazena os objetos geométricos em resposta aos comandos dos usuários. Essa biblioteca processa os comandos por meio da interface com o usuário da aplicação, e exibe o resultado graficamente para o usuário [62].

A modelagem é realizada por meio de *features* geométricas (ou de forma). *Features* se referem a uma região de uma parte do produto que contém certas propriedades geométricas e topológicas. Portanto, consistem de informações de

geometria e de parâmetros da região de interesse, sendo representadas pelos objetos geométricos.

A partir de uma primeira *feature* projetada, a superfície paramétrica que representa a borda (ou *boundary*) da *feature* sólida é criada no CAD. A partir dela são adicionadas outras *features*. Ou seja, *features* são um conjunto de objetos geométricos que combinados compõe uma parte.

Ao longo desse processo é criada uma árvore que representa a sequência da criação das *features*. Operações booleanas são empregadas para a adição ou remoção de uma *feature* de outra criada anteriormente. Ao mesmo tempo, a intersecção das curvas é calculada para avaliar as operações booleanas e exibí-las. Esse método é denominado CGS (*Constructive Solid Geometry*).

As intersecções de curvas são aproximadas gerando curvas específicas da parte, que o sistema então processa para armazenar como faces, vértices e arestas. Essas são armazenadas no sistema para serem exibidas. Uma vez que todas as *features* geométricas foram criadas e as relações foram estabelecidas, o modelo sólido da peça se apresenta completamente definido. Esse método é denominado B-Rep (*Boundary Representation*) [62].

As *features* são basicamente categorizadas em cinco grupos: *features* de construção (ou *datum*), de protuberância (extrusão, revolução, etc.), *pick and place* (chanfros, filetes, arredondamentos), de espelhamento, de padrão e de espessuramento.

Restrições, ou *constraints*, podem ser combinadas para relacionar ou restringir entidades. Essas são combinadas principalmente nas operações de criação de esboços de perfil (*sketch profiles*) e para a montagem de conjuntos (*assembly*).

Um outro tipo de *feature* necessário para o desenvolvimento do *Master Prototype* do produto são as *features* de manufatura. Essas *features* especificam as informações de manufatura. Elas consistem das informações da geometria da peça e as informações para a fabricação dessa geometria.

As *features* de tolerância especificam os desvios de dimensão, geometria, posição, orientação, planicidade, circularidade e concentricidade. Para a determinação dessas informações são utilizadas normas que direcionam sua definição apropriada, a mais usada é a ISO 2768. Informações não geométricas como tipo de material e características da superfície (ou rugosidade) completam as informações PMI.

Essas *features* são geralmente criadas nos sistemas CAD, nesse trabalho é utilizado o *software* NX[®]. Além disso, essas informações podem ser visualizadas juntamente à representação geométrica dos modelos JT no visualizador do Teamcenter[®].

Tendo em vista as características dos objetos do Teamcenter[®] e do NX[®] mencionados, serão relacionadas as informações do produto que são geradas no processo de desenvolvimento, com os objetos do Teamcenter[®], capazes de representar essas informações no sistema.

4.3. Objetos PLM

No tópico 2.2.2, os itens foram identificados como objetos fundamentais do sistema, que possibilitam a criação e organização das estruturas do produto. Documentos podem ser relacionados aos itens e conter diferentes tipos de informações – estruturadas e desestruturadas – de acordo com as necessidades do projeto e para a definição do produto. Portanto, são capazes de armazenar diferentes informações das características do produto.

Itens são objetos utilizados para organizar informações do produto em estruturas no sistema PLM, e integram também as informações relacionadas as partes dessa estrutura. Na definição dos dados do produto, um item pode representar um elemento, um conjunto ou um produto completo [4]. Ao instanciar um item, o sistema PLM instancia automaticamente uma primeira versão desse item. Essa versão pode ser revisada. Quando revisada, o sistema cria uma nova versão que representa alguma alteração da versão anterior [63].

Uma estrutura consiste de uma hierarquia. O item localizado no topo da estrutura é considerado o item raiz da estrutura. Os itens no nível imediato abaixo são conhecidos como seus filhos. Uma estrutura pode consistir de vários níveis. Partindo de cima para baixo, todos os itens têm um filho, exceto os itens pertencentes ao nível mais baixo. Partindo de baixo para cima, todos os itens têm pais, exceto o contido no nível mais alto [17,4].

No Teamcenter®, outro objeto essencial para o gerenciamento das informações do produto é o *dataset* (Figura 19). Esse objeto é usado para representar os documentos armazenados no *Vault* [63]. Os *Datasets*, assim como todos os objetos instanciados no sistema, são definidos por meio de metadados que possibilitam seu gerenciamento no sistema.

O Teamcenter® integra as aplicações do Microsoft Office®. Essa integração possibilita a edição dos documentos criados – com as devidas permissões de acesso – nas aplicações do Teamcenter®. A Figura 19 apresenta a estrutura de um item individual com os *datasets*, de diferentes tipos, relacionados a ele.

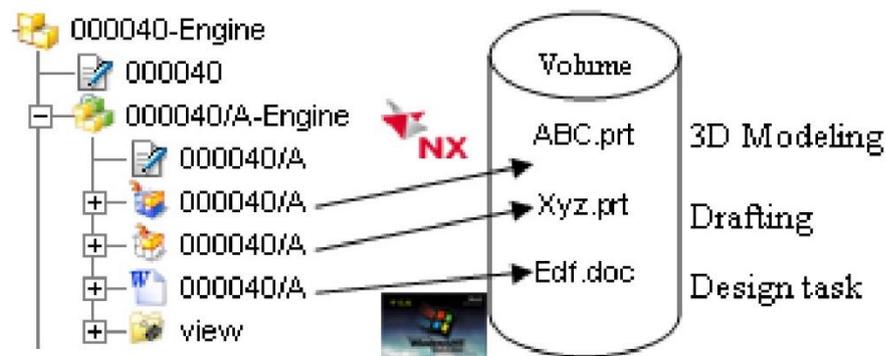


Figura 19: *Datasets* podem armazenar documentos de diferentes formatos (Zhan et. al. [63]).

Os arquivos CAD podem ser armazenados no *Vault* e manipulados de forma integrada por meio de um módulo integração do Teamcenter® com o NX®. Essa integração permite acessar o NX® através do Teamcenter®, assim como, criar *datasets* de tipo específico para arquivos CAD, durante a execução do Teamcenter® [63,22].

As informações contidas no arquivo CAD podem ser convertidas para um formato de visualização JT. Dessa forma elas podem ser visualizadas pela interface do Teamcenter® por meio das aplicações voltadas para a visualização e colaboração. Na conversão, um *dataset* específico para o arquivo JT, denominado *Direct Model*, é criado no Teamcenter®.

4.4. Funcionalidades PLM

Nos tópicos 4.2 e 4.3 foram identificados os tipos de objetos do Teamcenter® e NX® capazes de representar as informações do *Master Prototype* geradas a cada etapa do processo de desenvolvimento do produto. A partir desse reconhecimento determinar determinam-se os objetos adequados para cada informação do *Master Prototype* e as funcionalidades necessárias para a instanciação desses objetos, permitindo, assim, conceber a estrutura do *framework*.

No tópico 4.1 foram apresentadas as informações geradas ao longo do processo de desenvolvimento do produto com foco nas informações do *Master Prototype*. Alguns aspectos dessas informações que foram ressaltados podem levar a uma identificação mais apropriada dos objetos que as representam. Será realizada a descrição desses aspectos, seguida pela identificação dos objetos adequados e das funcionalidades que permitem instanciar e gerenciar esses objetos.

Os requisitos do produto são especificações que descrevem diferentes aspectos do produto que será desenvolvido. Diferentes requisitos podem ser consumidos em diferentes etapas do processo de desenvolvimento do produto para a realização das atividades de definição, criação e avaliação, de acordo com as informações necessárias para a etapa.

Requisitos determinantes de função do produto, por exemplo, são usados na etapa de projeto conceitual, enquanto que os requisitos determinantes de forma e arquitetura são usados principalmente nas primeiras atividades do projeto preliminar. Busca-se atender aos requisitos exigidos antes dos desejados. Alguns requisitos podem ser atendidos por diferentes partes da estrutura do

produto. Além disso, os requisitos podem mudar ao longo do processo, requerendo modificações e atualizações [1].

Portanto, de certa forma, essas informações precisam ser trabalhadas da maneira mais flexível possível. Os itens podem atender a esse tipo de flexibilidade, uma vez que permitem uma distinção individual dos requisitos no sistema. Além disso, esses objetos podem ser relacionados por meio de *trace links* que permitem representar a relação entre requisitos que definem e partes do produto que atendem a essas definições⁴ [64]. Portanto, as funcionalidades necessárias para o gerenciamento dos requisitos são:

- Criar itens para os requisitos.
- Relacionar os requisitos as partes da estrutura do produto por meio de *trace links*.

Foi apresentado no tópico 4.1 que as informações que definem os princípios de solução se referem a função, características geométricas, propriedades de materiais e movimento do produto e das partes. Essas características definem os princípios de solução e são geralmente determinadas de forma simplificada. Pelo seu teor abstrato elas são registradas em arquivos de formato desestruturados como arquivos de texto, planilhas, imagens, entre outros. Porém podem ser criados também modelos tridimensionais para representar as características dos princípios de solução.

Cada elemento portador de função representa uma parte do produto. Essas partes são combinadas em uma estrutura. Para a representação de uma estrutura do produto, os itens, que representam as partes do produto, são geralmente combinados em conjuntos e no item raiz, representando o produto completo.

⁴ *Trace links* são usados para inter-relacionar elementos do produto. É uma forma de informação utilizada para descrever e seguir certos aspectos da vida de um elemento do produto, como por exemplo, os requisitos. O tema é abordado por Königs et. al. [62]. O Teamcenter® permite criar *traces links* para estabelecer rastreabilidade entre requisitos e outros objetos da base de dados.

Os documentos gerados nessa etapa são representados por *datasets*. Eles são relacionados ao item da parte que descrevem, mantendo as informações organizadas.

Portanto, as funcionalidades necessárias para essa etapa são:

- Armazenar documentos com informações geradas no projeto conceitual, como diagramas que representam relações entre subfunções, e documentos, estruturados ou não, que representam características básicas de forma, materiais e comportamento do produto.
- Criar itens para representar estrutura do produto de acordo com os princípios de solução.

Para determinar a arquitetura do produto, parte-se da estrutura dos princípios de solução. Os documentos aprovados da versão de princípios de solução podem ser acessados e quando copiados para a nova versão podem ser também alterados para a versão da arquitetura do produto. Para alterar um documento, o usuário usa o mecanismo *check in* e *check out*⁵, que resulta no congelamento contra mudanças durante o uso do documento.

A arquitetura deve representar as características do produto de forma mais concreta. Durante a determinação da arquitetura pode ser necessário adicionar novas partes a estrutura ou dividir as partes, anteriormente determinadas, em partes menores formando subconjuntos.

Para a determinação da arquitetura são criados modelos geométricos e de simulação de comportamento do produto. Os parâmetros que definem as características de geometria do produto são inseridos em um modelo gerado a partir da instanciação das *features* geométricas.

⁵ Com o *check out*, o usuário que tem permissão retira o documento do *Vault* para a realização das alterações e o *check in* retorna o documento, permitindo alterações por outros usuários também autorizados.

Tendo modelado a geometria das peças, torna-se necessário combinar as partes para relacionar as peças no conjunto final, sendo assim representado graficamente o produto completo e as relações existentes entre as peças. O processo de modelagem geométrica das peças e montagem dos conjuntos e do conjunto final pode ser realizado no NX[®] por meio da integração com o Teamcenter[®].

As informações do modelo geométrico podem ser convertidas para o formato JT para serem visualizadas diretamente no Teamcenter[®]. Outros documentos podem ser criados, referentes a avaliação do produto.

Assim sendo, as funcionalidades necessárias para a determinação da arquitetura do produto na etapa de projeto preliminar são:

- Visualizar e/ou alterar documentos do projeto conceitual.
- Criar itens e estruturar as partes novas da estrutura.
- Modelar a geometria do produto.
- Montar conjuntos.
- Simular os modelos de comportamento do produto.
- Armazenar documentos específicos do projeto preliminar.

O *Master Prototype* compreende as especificações finais do produto. Para sua determinação, novos parâmetros são adicionados aos modelos e objetos de representação da arquitetura do produto. Nos modelos geométricos são adicionadas *features* de manufatura, ou seja, as informações de manufatura do produto (PMI).

Para apresentar as informações PMI no Teamcenter[®], podem ser criados desenhos detalhados das peças e conjuntos, assim como representações de geometria no formato JT que podem incluir essas *features*. A adição dessas

informações no modelo geométrico é realizada através do NX[®] integrado com o Teamcenter[®].

Dessa forma, as funcionalidades que contribuem com a finalização das informações do Master Prototype, na etapa de projeto detalhado são:

- Alterar documentos do projeto preliminar.
- Adicionar informações de manufatura ao modelo geométrico e gerar desenho detalhado.
- Armazenar novos documentos gerados no projeto detalhado.

Alguns aspectos e funcionalidades referentes a definição e controle do processo de desenvolvimento do produto, assim como, aquelas necessárias à configuração do modelo do produto no Teamcenter[®] completam a estrutura do *framework*.

As informações administrativas e organizacionais são definidas ao longo do processo. Essas informações se referem às partes do produto e estrutura. Para visualizá-las deve-se recorrer aos módulos do Teamcenter[®] voltados a gestão de estruturas como *Structure Manager* ou *Multi Structure Manager*. O uso dessas aplicações durante o processo de desenvolvimento do produto facilita a visualização dos dados de representação estrutural do produto.

O produto completo e as partes são definidas pelo item raiz e por componentes, respectivamente. Os componentes podem ser uma parte que é uma peça (*piece part*) ou uma parte que é um conjunto (*assembly part*). Um item raiz e seus componentes são representados no Teamcenter[®] por uma linha no painel da estrutura (*structure panel*). Essas linhas são também denominadas ocorrências.

Ao criar ou adicionar um item na estrutura – adicionado a partir do item raiz ou de outro componente na estrutura – o Teamcenter[®] instancia automaticamente uma ocorrência desse item. Essas ocorrências são armazenadas em um objeto

denominado BVR (*BOM View Revision*) que representa a relação imediata entre uma revisão de item pai e seus respectivos itens, ou revisões de itens, filhos.

No Teamcenter®, para que uma parte receba o status de aprovação deve-se associar a ela um *template* de *workflow*. Essa funcionalidade permite o controle do processo e a atribuição de atividades aos usuários. Ao ser aprovado, o item é congelado no Teamcenter® e não pode mais ser alterado. O que pode ocorrer é a criação de uma nova versão desse item que permite que essas alterações sejam realizadas. Ao criar uma nova versão de um item, é criado um novo objeto *Item Revision*. Os objetos desse item revisado podem ser copiados, ou não, para a nova versão.

Como exposto anteriormente, o módulo *Multi-Structure Manager* pode ser usado para visualizar e gerenciar estruturas do produto. Esse módulo permite manipular diferentes representações de estruturas em um mesmo ambiente, fornecendo ferramentas para a criação e definição de informações ao longo do processo de desenvolvimento do produto. Para que as estruturas sejam gerenciadas nesse ambiente é necessário que sejam criadas as visões das estruturas que serão usadas.

Visões são representações de uma estrutura em um contexto de configuração determinado. No Teamcenter®, para criar uma visão é necessário instanciar três tipos de objetos: o *CCObject* que deverá armazenar as representações das estruturas em um mesmo contexto, o *StructureContext* cria uma representação individual de uma estrutura e o *ConfigurationContext* que estabelece as regras para a representação das revisões do produto.

Na criação de um contexto de configuração para uma estrutura do produto é selecionada uma regra de revisão (*revision rules*) para a estrutura. A regra selecionada determina as revisões dos itens estrutura do produto que deverão ser exibidas no painel de estrutura do *Multi-Structure Manager*. As revisões podem ser exibidas de acordo como *status* do ciclo de vida como por exemplo liberado (*release*), sendo trabalhado (*working*), pela data de criação e etc.

O *Master Prototype* compreende também o modelo de direitos de acesso. Esse modelo representa as relações entre os usuários do sistema, seus grupos e suas responsabilidades. O Teamcenter® possui um modelo padrão para gestão dos direitos de acesso aos objetos do banco de dados. Além disso, permite que os proprietários de um objeto determinem regras de acesso adicionais referentes a esse objeto. As regras de acesso podem ser determinadas para diferentes usuários do sistema, de acordo com o grupo pertencente e as responsabilidades dentro de um projeto ou da organização.

4.5. Visão geral do *framework*

Esse trabalho visa contribuir com uma proposta de um *framework* para o desenvolvimento de um *Master Prototype* do produto. Por *framework* deve-se entender uma estrutura de métodos e operações para a criação das informações do *Master Prototype* ao longo do processo de desenvolvimento de produtos tendo como princípio a abordagem PLM.

Sendo as informações do *Master Prototype* ponto de partida para a determinação das funcionalidades relevantes do Teamcenter®, mostrou-se essencial:

- A criação de uma estrutura de lista de requisitos e uso de *trace links* para determinar uma relação entre os requisitos que definem uma característica do produto e parte da estrutura do produto em conformidade com o requisito.
- A estruturação e configuração do produto para a criação da representação estrutural e determinação das informações administrativas e organizacionais do produto e das partes.
- A modelagem do produto, com a criação das *features* geométricas que representam a geometria do produto.
- A simulação do modelo do produto, a partir da criação de modelos de comportamento que representam as funcionalidades do produto.

- Inclusão de PMI no modelo geométrico, a fim de determinar as características de teste e inspeção do material, completando assim a definição das características do produto e do seu processo de produção.

A Figura 20 apresenta uma visão geral para a realização da proposta do *framework*. A visão geral consiste de três partes:

- Etapas do processo de desenvolvimento do Master Prototype (à esquerda).
- Informações do *Master Prototype* (no centro).
- Funcionalidades do PLM (à direita).

O *framework* será verificado em um cenário fictício de desenvolvimento de produto. Durante o processo serão exibidas para cada etapa do processo, as informações do *Master Prototype* e a ordem – numérica – em que as funcionalidades foram usadas.

As etapas do processo de desenvolvimento do *Master Prototype* são as mesmas usadas nas abordagens de desenvolvimento de produtos tradicionais, ou seja, de acordo com as etapas que foram apresentadas no tópico 4.1. Nesse mesmo tópico foram levantadas também as principais informações do *Master Prototype*.

As funcionalidades foram divididas em macro funcionalidades. As funcionalidades que envolvem a determinação e controle do processo e a configuração da estrutura do produto no *Multi-Structure Manager* foram reunidas na macro-funcionalidade “definição do contexto e do processo”. Funcionalidades que envolvem o acesso ao sistema CAD através do Teamcenter foram reunidas na macro-funcionalidade de “integração com CAD”.

A macro-funcionalidade de gestão de requisitos envolve as funcionalidades de criação de uma estrutura em que cada item representa um tópico da lista de

requisito. Essa estrutura é criada da mesma maneira que a estrutura do produto, porém compreendendo significados diferentes.

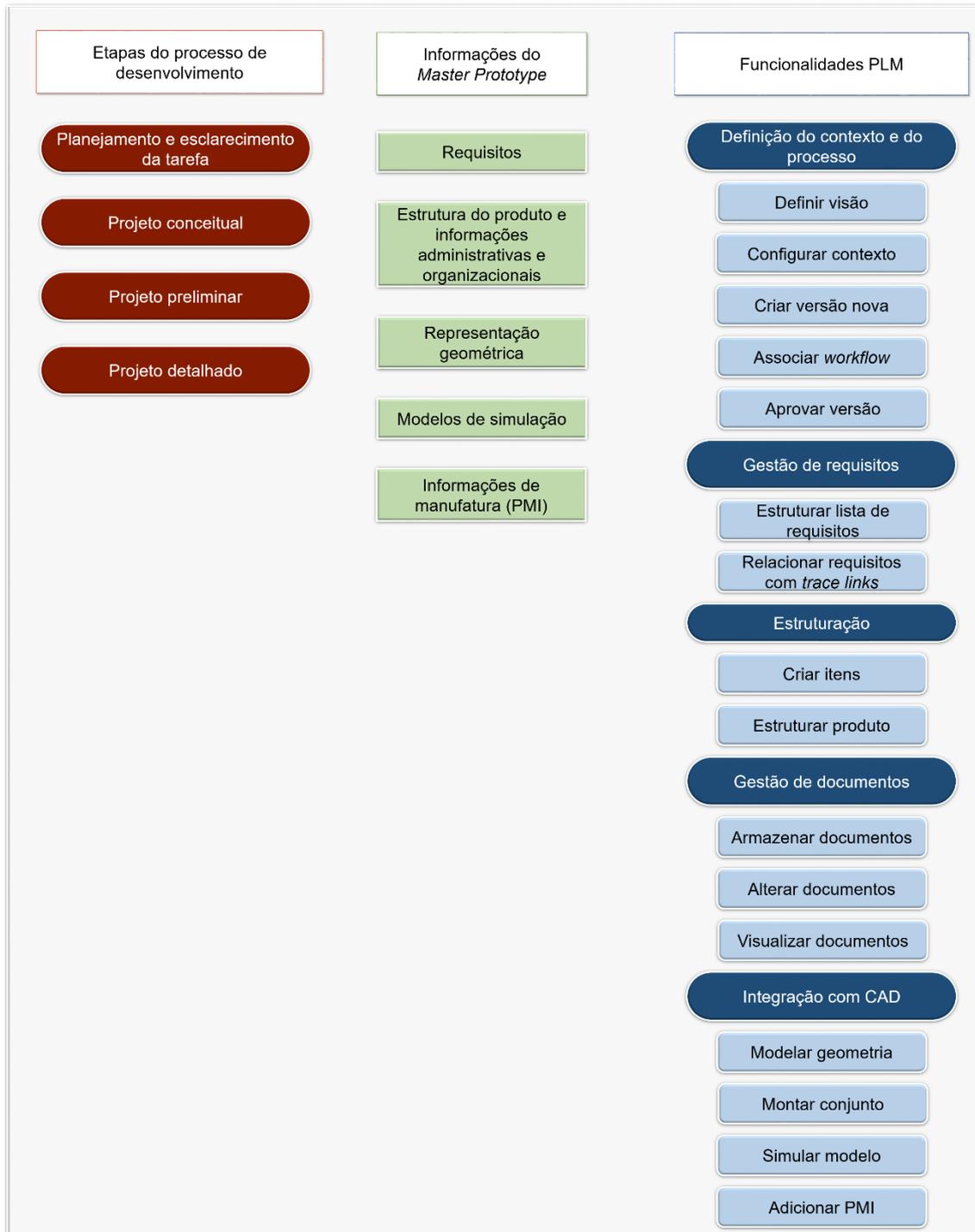


Figura 20: Visão geral do *framework* para o desenvolvimento de um *Master Prototype* do produto.

As outras macro-funcionalidades, de “estruturação” e de “gestão de documentos” representam, respectivamente as funcionalidades de criação e gerenciamento da estrutura de itens que forma o produto e da instanciação e gerenciamento de *datasets* que representam os documentos contendo as informações do produto.

5. Resultados

Esse capítulo apresenta a verificação da proposta do *framework* em um cenário fictício de desenvolvimento de um produto. As atividades realizadas e funcionalidades usadas para a criação do *Master Prototype* de produto, usando as ferramentas Teamcenter® e NX®, serão apresentadas.

Primeiramente o cenário foi modelado usando diagramas de casos de uso. A partir desse cenário, foram determinadas as atividades que seriam realizadas de acordo com o método de desenvolvimento de produtos explicado no tópico 4.1.

Ao longo das etapas, as funcionalidades consideradas mais relevantes foram detalhadas em diagramas de sequência. Esses diagramas apresentam a sequência de mensagens que são trocadas durante a interação de objetos, atores e sistemas.

Os diagramas deste trabalho estão de acordo com Pereira [65] e Seidl et. al. [66], e foram construídos com a ferramenta Visio®.

Na representação da sequência das operações o objetivo não foi cobrir todo os comportamentos dos sistemas Teamcenter® e NX®, como por exemplo todas as respostas ou mensagens que ocorrem na interação entre os objetos no sistema. Serão demonstrados somente os aspectos mais relevantes para a proposta do trabalho e que foram visíveis durante o desenvolvimento e verificação do *framework*.

5.1. Cenário

O cenário fictício considera uma empresa de nome fictício de MP. A MP é uma empresa de desenvolvimento de produtos que busca aplicar a abordagem PLM e os conceitos da Indústria 4.0 para a realização dos seus processos. Para isso adota como princípio fornecer as informações dos produtos de forma integrada, consistente e digitalizada, usando os softwares de gestão do ciclo de vida do produto e de modelagem Teamcenter® e NX®.

A empresa recebe do cliente a tarefa de desenvolver um cilindro pneumático que atenda as especificações apresentadas pelo cliente final. O cliente direto é uma empresa responsável pela produção, aquisição de recursos e montagem do cilindro pneumático que será desenvolvido. Esse cliente usará as informações fornecidas pela MP para otimizar e adequar seus processos aos princípios da Indústria 4.0. Para isso, o cliente estará adotando um *Master Prototype* do produto como ponto de partida para a realização do controle dos seus processos e acompanhamento dos produtos fabricados ao longo do ciclo de vida.

O processo de desenvolvimento do produto adotado pela empresa MP se baseia nas metodologias de processo de desenvolvimento do produto tradicionais, com etapas definidas de acordo com as etapas apresentadas no tópico 4.1.

A Figura 21 apresenta a modelagem dos casos de uso. Cada elipse é um caso de uso que representa determinada funcionalidade do Teamcenter®. O termo ator se refere ao papel que alguém ou alguma coisa interpreta enquanto interage com o sistema usado. Por isso eles foram colocados do lado de fora do quadro – que representa o limite do sistema.

A MP adota grupos para a realização de projetos. As informações dos grupos, usuários e responsabilidades no projeto estão inseridas no Teamcenter®. Os usuários têm dois tipos de responsabilidades e direitos de acesso no Teamcenter®, gerente e projetista. Nesse contexto:

- O gerente é responsável pelas atividades de gerenciamento do processo e das estruturas como criar itens, associar *workflow* aos itens, avaliar e aprovar versões dos itens, criar novas versões, estruturar, criar visões para as estruturas e configurar contexto das estruturas. Ele pode atuar de forma conjunta ou alternada com o projetista nas atividades de armazenar, visualizar e alterar documentos.
- O projetista é responsável pelas atividades de definição e criação do produto. Por isso, todas as funcionalidades que incluem o caso de uso “acessar NX®” foram atribuídas a ele. Ao acessar o NX® o projetista

pode usar as funcionalidades: modelar produto, simular modelo, adicionar PMI e montar conjuntos. Além disso, o projetista tem a responsabilidade de relacionar os requisitos às partes do produto.

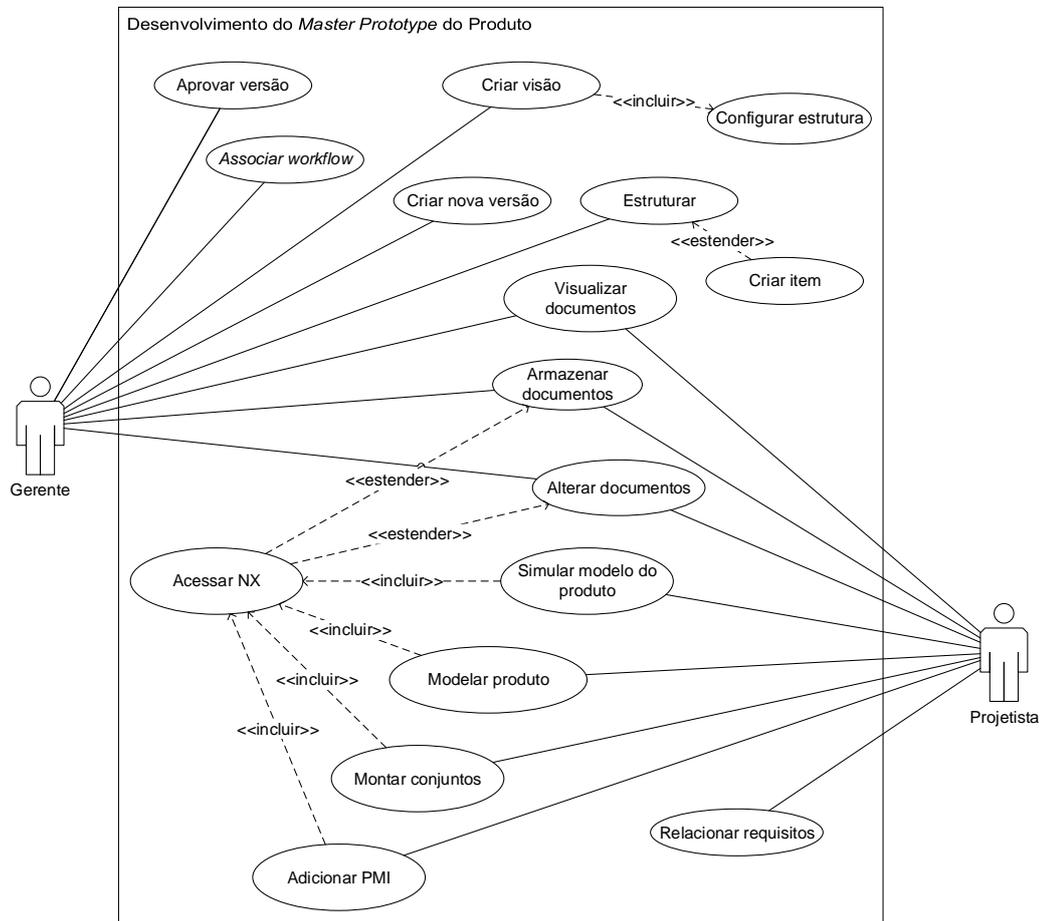


Figura 21: Modelagem de casos de uso para a verificação do *framework*.

O diagrama apresenta algumas relações entre os casos de uso com os rótulos <<estender>> e <<incluir>>. Nas relações com o rótulo <<estender>> o caso de uso que aponta a seta pode ser invocado, ou não, junto ao caso de uso apontado, ou seja, representa uma variação do comportamento normal. Por outro lado, naquelas com o rótulo <<incluir>> o caso de uso que aponta a seta deve usar a funcionalidade do caso de uso apontado durante a realização da atividade.

O caso de uso “estruturar” é estendido pelo caso de uso “criar item”, uma vez que uma estrutura pode ser construída por meio da criação de itens a partir de

outro item, em que este vem a ser o pai de um conjunto ou da estrutura completa. O caso de uso “criar visão” inclui o caso de uso “configurar a estrutura”, uma vez que uma configuração do contexto precisa ser definida para que visão criada.

O caso de uso “acessar NX[®]” estende o caso de uso “alterar documentos”, e também o caso de uso “armazenar documentos”. Isso porque essas funcionalidades podem ser usadas quando o usuário acessa o NX[®]. Quando ele acessa e salva o arquivo pela primeira vez, é usada a funcionalidade “armazenar documentos”. Porém, na outras vezes que acessar o arquivo e salvar, usará a funcionalidade “alterar documentos”.

No NX[®] o projetista pode criar e montar modelos geométricos e de simulação do produto, além de poder adicionar informações PMI no modelo geométrico. Essas funcionalidades geram arquivos que são armazenados no *Vault*, representados por *datasets* e relacionados a algum item no Teamcenter[®].

Os objetos do sistema são, por padrão, configurados com restrição de acesso de acordo com o perfil do usuário no modelo de gestão de acessos do Teamcenter[®]. Estes são hierarquizados em grupos e responsabilidades. Usuários de um mesmo grupo compartilham determinados direitos de acesso como: direito de leitura, edição e remoção do conteúdo nos objetos⁶.

Neste cenário, o gerente tem a responsabilidade de instanciar todos os itens da estrutura do produto, para que tenha total acesso a alterar e criar novas versões dos itens e do seu conteúdo. Por exemplo, durante a execução de um processo, o gerente tem acesso a alterar as propriedades dos itens.

O projetista tem direito de adicionar objetos na estrutura, visualizar os objetos referentes ao projeto e alterar os objetos criados por ele mesmo. Porém, nenhum

⁶ A regras de direitos de acesso podem ser alterados pelo administrador e o proprietário do objeto (criador) tem permissão de adicionar novas regras de acesso ao objeto que criou.

dos usuários tem acesso a modificação dos itens e conteúdo dos itens já liberados.

Tendo em vista o cenário apresentado, as etapas do processo de desenvolvimento do cilindro pneumático são apresentadas.

5.2. Planejamento e esclarecimento da tarefa

Na etapa de planejamento e esclarecimento da tarefa, a empresa MP recebe a tarefa de desenvolver um cilindro pneumático de acordo com as especificações enviadas pelo cliente. Essas especificações devem ser esclarecidas para especificações do produto, denominados requisitos. O gerente tem a responsabilidade de adicionar esses requisitos no Teamcenter®, para que no final da etapa, os avalie e aprove.

A Figura 22 apresenta a ordem em que as funcionalidades foram usadas nessa etapa. A ordem numérica em que foram realizadas as atividades será identificada ao longo do texto entre parênteses.

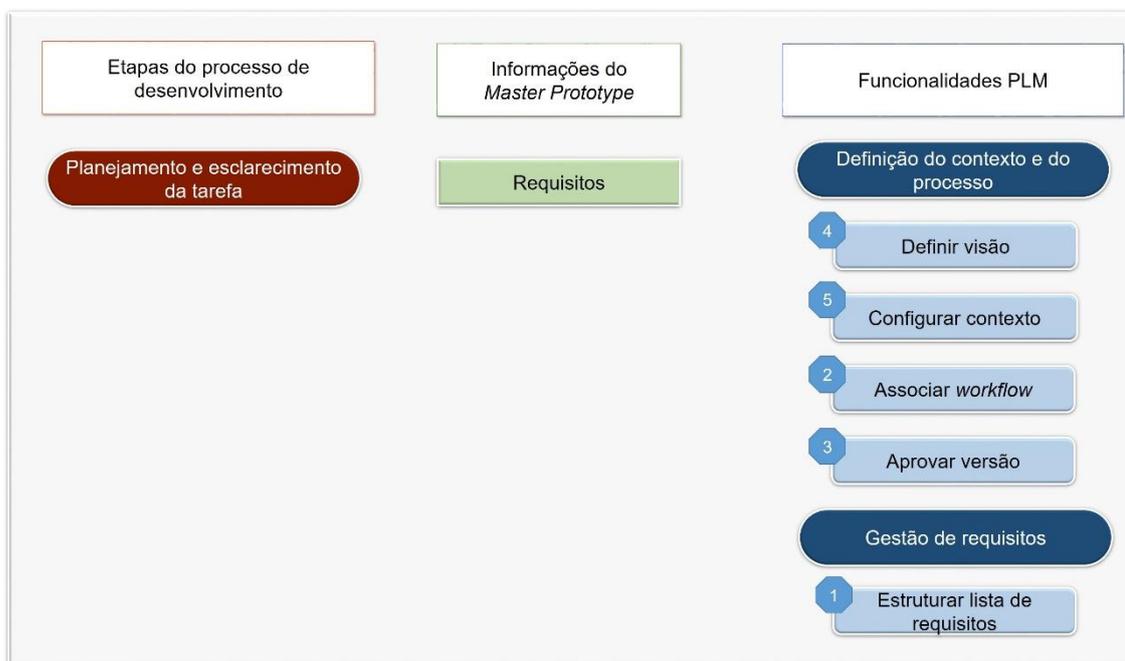


Figura 22: Representação do *framework* para a etapa de planejamento e esclarecimento da tarefa.

O gerente cria a pasta do produto e um objeto *folder* é instanciado para organizar todas as informações do processo de desenvolvimento do cilindro pneumático. Ele cria itens e uma estrutura contendo o título e tópicos da lista de requisitos (1).

As operações para criar a estrutura da lista de requisitos estão representadas na Figura 23.

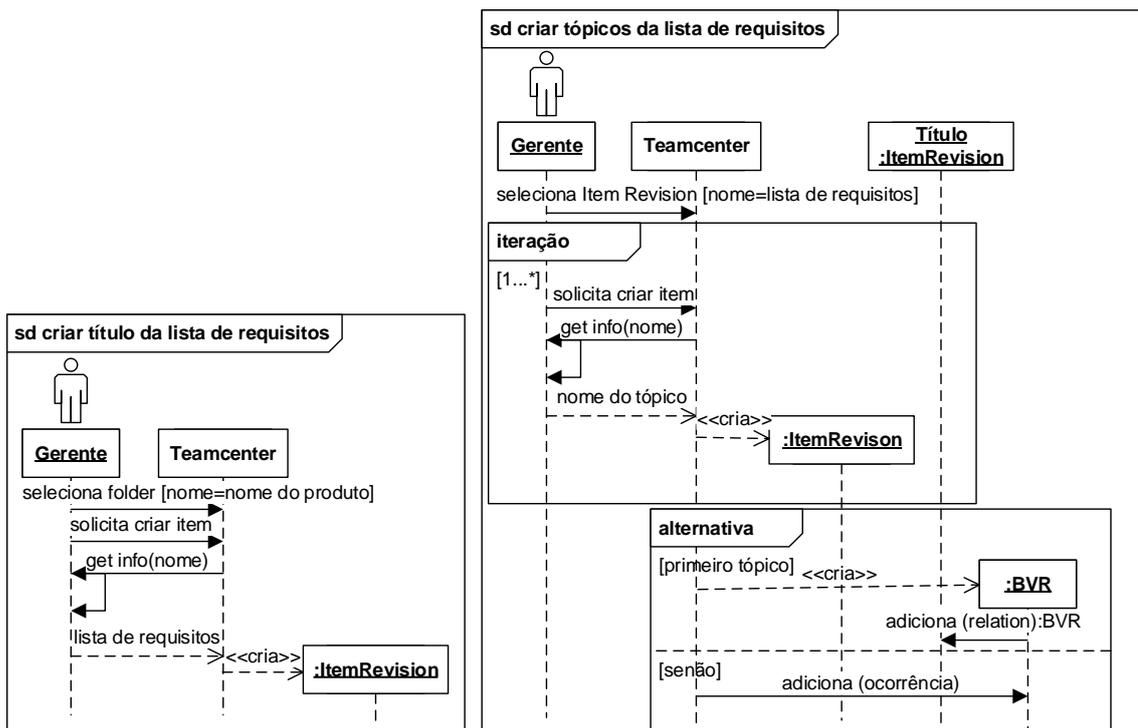


Figura 23: Sequência de mensagens para a estruturação da lista de requisitos.

Primeiramente, cria-se o título de lista de requisitos. O gerente seleciona a pasta do produto criada anteriormente e solicita ao Teamcenter® criar um item. O sistema envia uma mensagem ao gerente solicitando que ele preencha algumas informações daquele item.

Apesar de ser possível determinar muitas propriedades do item no momento da criação, as informações que darão significado para requisito no sistema são adicionadas nos campos de nome do item e descrição. No parâmetro “nome do item”, o gerente adiciona a descrição do requisito e no campo “descrição”, coloca o valor.

São atribuídos aos itens, automaticamente, uma identificação numérica de acordo com a sequência da criação de itens no Teamcenter® e uma identificação alfabética, que representa a criação de uma primeira versão do item, denominado *Item Revision*.

Para criar a estrutura da lista de requisitos (Figura 23, à direita), o gerente seleciona o item raiz da estrutura da lista de requisitos e solicita criar um novo item. O item criado representa um tópico da lista de requisitos.

No diagrama, depois que o gerente seleciona o item raiz, é iniciada uma iteração em um fragmento de mensagens. Essa iteração representa que são criados vários itens que representam os tópicos da lista de requisitos. Ao instanciar o primeiro item tópico da lista de requisitos, é instanciado automaticamente um objeto BVR, que armazena as relações imediatas entre um item pai e um item filho.

Depois de criar a estrutura, o gerente associa um *workflow* para que os requisitos sejam aprovados e congelados (2 e 3). Ele cria também uma representação da estrutura (4 e 5). Para isso solicita criar os objetos CCOBJECT, para criar um contexto de colaboração, e um StructureContext e ConfigurationContext para que a lista de requisitos seja manipulada na perspectiva *Multi-Structure Manager* e para que a visão configurada possa ser compartilhada com o projetista.

A estrutura da lista de requisitos é exibida no painel de colaboração do Teamcenter® na perspectiva do *Multi-Structure Manager*.

5.3. Projeto conceitual

A Figura 24 apresenta a visão do *framework* e a ordem de uso das funcionalidades no projeto conceitual. A ordem é indicada ao longo do texto, entre parênteses.

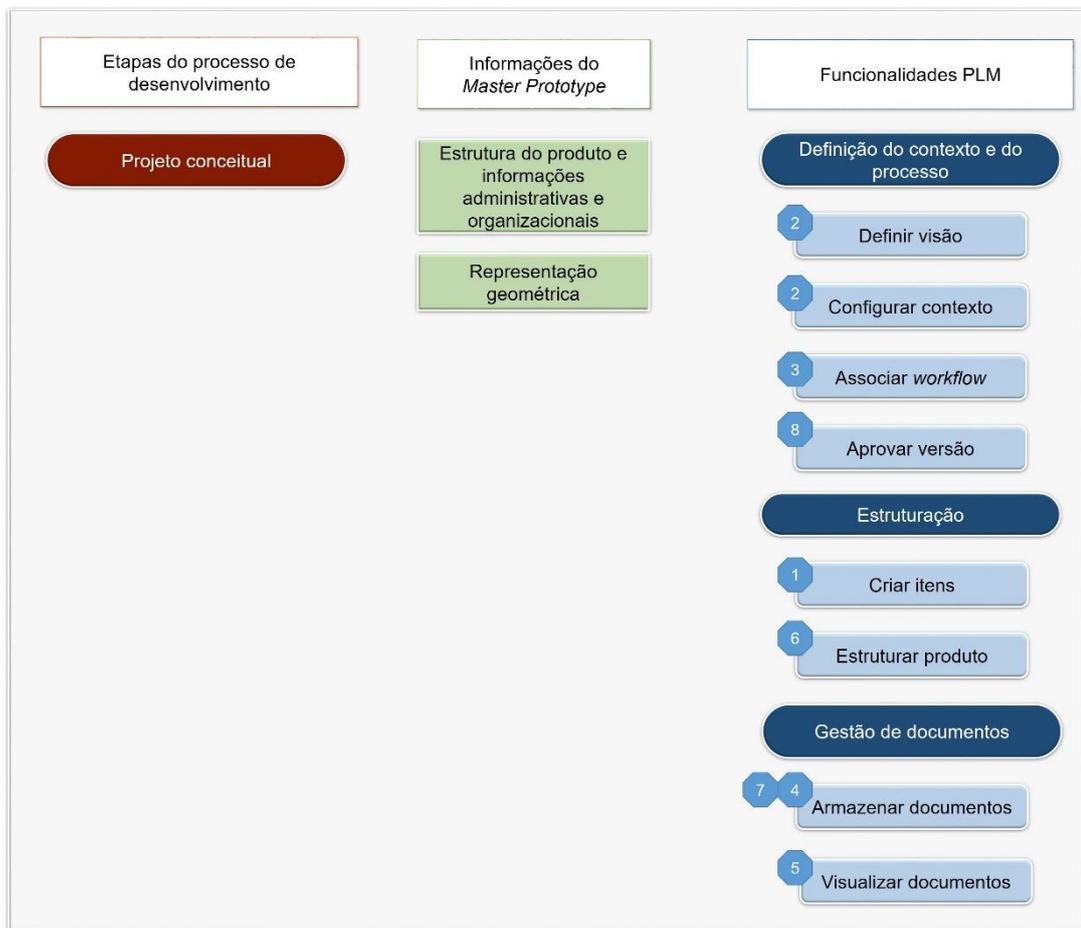


Figura 24: Representação do *framework* na etapa de projeto conceitual.

No início da etapa, o gerente prepara um contexto para a estrutura do produto. Para isso, cria um item para, posteriormente, representar o conjunto completo da estrutura do produto (1). Ao criar um item, a primeira versão desse item é instanciada. Uma identificação numérica individual é atribuída ao item e uma identificação em ordem alfabética que representa a sua versão como versão A.

Para criar o item raiz da estrutura do produto, o gerente seleciona a pasta do produto e solicita criar um item. O Teamcenter® atribui um número de identificação ao item e solicita que o gerente adicione as informações do produto (Figura 25, à esquerda). O gerente adiciona o nome da parte do produto e confirma. Com a confirmação, o Teamcenter® cria a primeira versão do item (*Item Revision*).

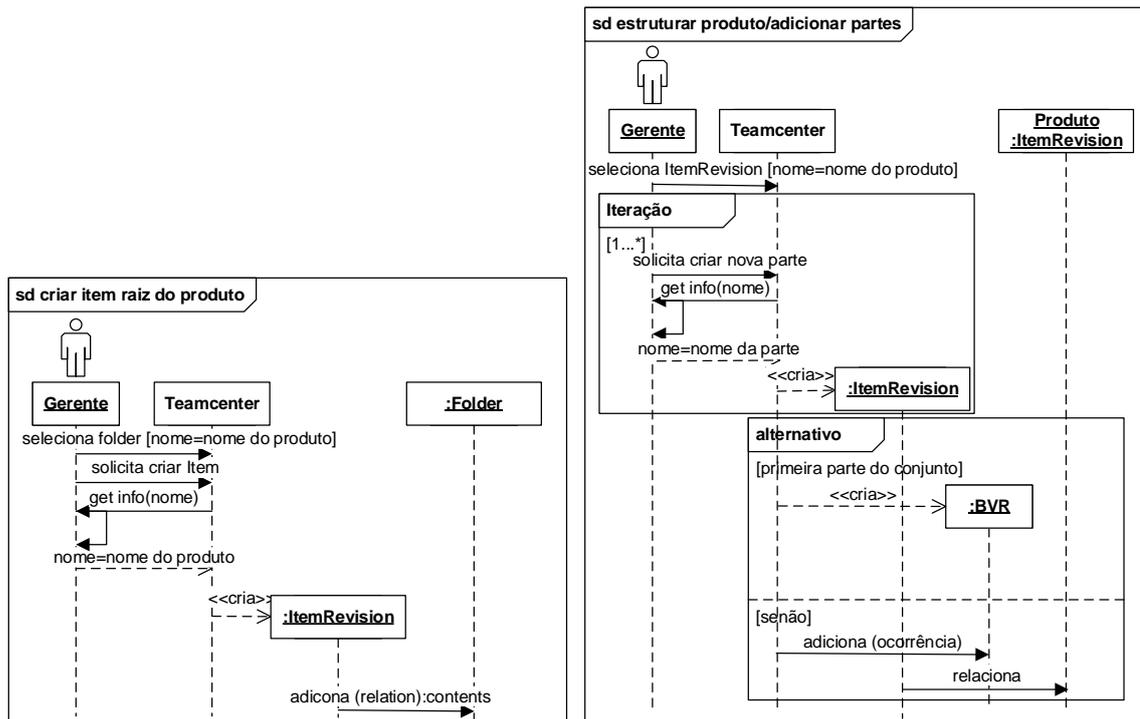


Figura 25: Sequência de mensagens para estruturar em conjuntos.

O gerente associa um *template* de *workflow* ao item raiz da estrutura do produto e cria uma visão para representar o produto (2 e 3). Para criar essa visão, são instanciados os objetos *StructureContext* e *ConfigurationContext* e uma referência⁷ do item raiz é inserida no contexto criado.

O projetista recebe as atividades via *workflow*. Ele determina a função global e as subfunções do produto. As subfunções são representadas por um diagrama de blocos em um documento Word[®] que é importado para o Teamcenter[®] (4). O projetista armazena também outro documento Word[®] contendo os parâmetros dos elementos dos princípios de solução selecionados e a definição da estrutura. Para representar esses documentos no Teamcenter[®], são instanciados dois *datasets* do tipo Word[®]. O projetista confirma o término da atividade e uma mensagem é enviada ao gerente via *workflow*.

⁷ Quando um objeto é instanciado, ele fica armazenado no banco de dados. Uma referência representa uma menção desse objeto em alguma estrutura, pasta ou outro container do Teamcenter[®]. Essa operação pode ser realizada copiando e colando o objeto no local desejado.

A partir da estrutura do produto, determinada pelo projetista, o gerente cria uma representação da estrutura do produto no Teamcenter® (5 e 6). Para esse fim, o gerente usa a funcionalidade “estruturar” com a criação de itens associados ao item raiz. Os itens filhos adicionados na estrutura representam as partes do produto. O gerente associa um *template* de *workflow* para cada um dos itens, para que sejam adicionadas as informações específicas das partes da estrutura. A Figura 25 apresenta a sequência de mensagens para criar os itens das partes da estrutura e estruturar um produto (à direita).

Para estruturar o produto, o gerente seleciona o item raiz do produto e solicita ao Teamcenter® criar um item. O gerente realiza esse procedimento várias vezes até que todas as partes da estrutura tenham sido adicionadas no produto (conjunto final)⁸. Um objeto BVR, com o registro de todas as ocorrências do produto, é instanciado e atribuído ao *Item Revision* do conjunto. Essa estrutura pode ser visualizada no painel de estrutura no *Multi-Structure Manager*.

Partindo dos parâmetros definidos até o momento, o projetista determina características básicas das partes, como geometria e possíveis materiais. Ele descreve isso em esboços. Os resultados são armazenados no Teamcenter® (7). A Figura 26 apresenta um dos esboços contendo dimensões básicas da tampa traseira do cilindro pneumático desenvolvidos na etapa.

No final da etapa, o projetista confirma que completou a atividade e uma mensagem é enviada ao gerente (8). O gerente avalia os princípios de solução e aprova a versão conceitual do cilindro pneumático. Ele seleciona a atividade de cada uma das peças e conjunto final e informa ao Teamcenter® que a atividade foi aprovada.

⁸ No contexto deste trabalho, optou-se por relacionar as peças diretamente ao item raiz, não foram criados subconjuntos.

Um *status* de liberação é relacionado ao item, mediante a aprovação do item em um processo de *workflow*. As operações para aprovar as partes e conjunto final são apresentadas na Figura 27.

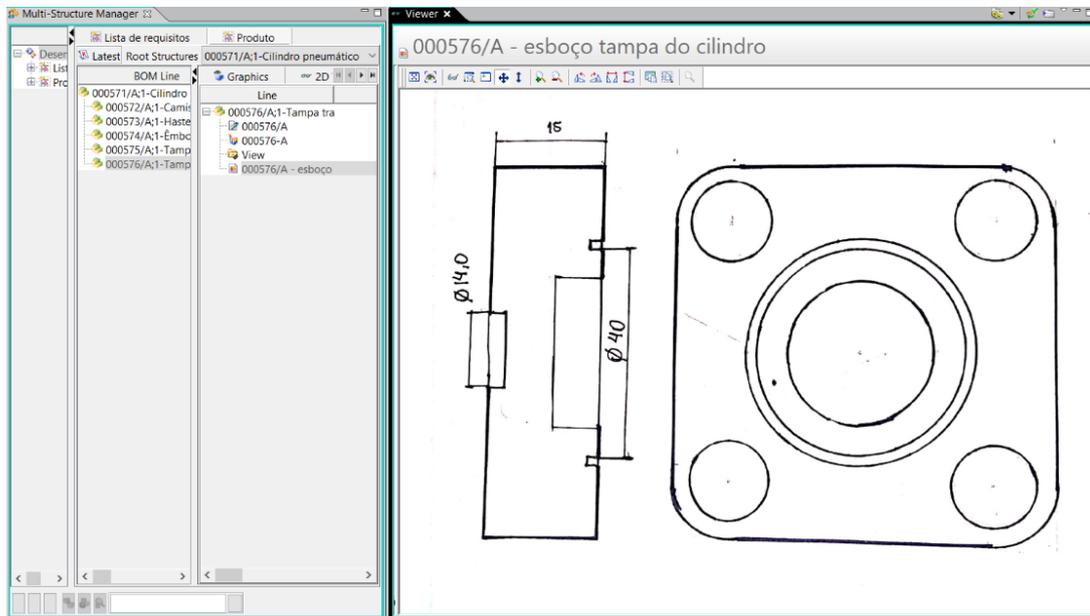


Figura 26: Visualização do esboço da geometria da tampa do cilindro armazenado no Teamcenter®.

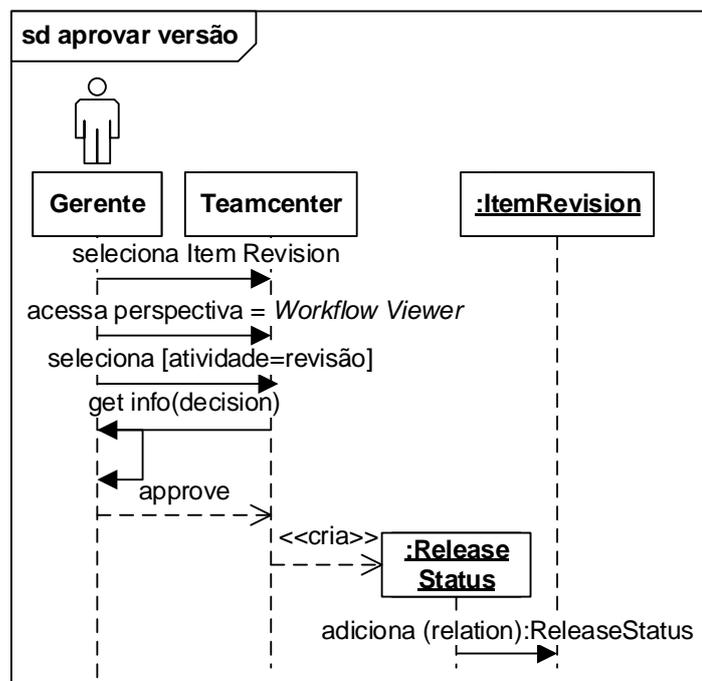


Figura 27: Sequência de mensagens para aprovar versão.

5.4. Projeto preliminar

A Figura 28 apresenta a visão do *framework* para a etapa de projeto preliminar.

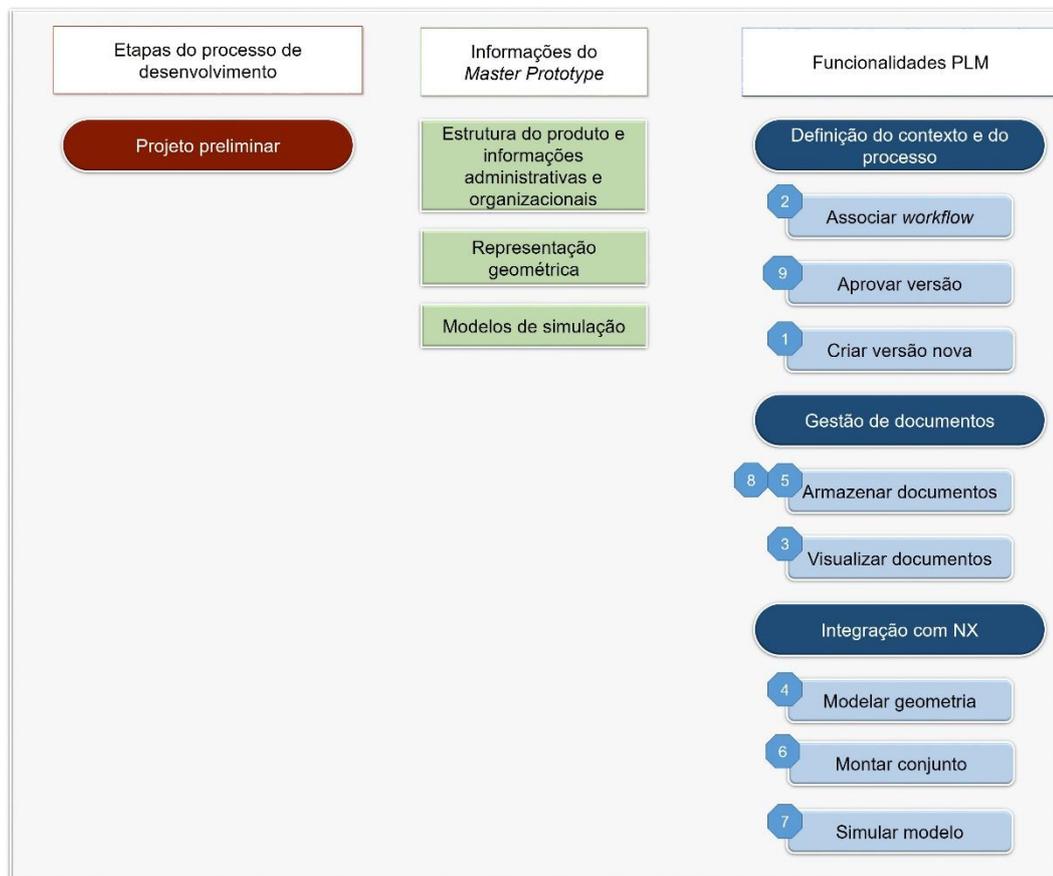


Figura 28: Representação do *framework* para a etapa projeto preliminar.

No início do projeto preliminar, o gerente cria uma nova versão das partes e do conjunto completo (1). A nova versão recebe uma identificação de versão B. O gerente associa *templates* de *workflow* à nova versão de cada parte, para que o projetista realize as atividades de determinação da arquitetura do produto (2). A Figura 29 apresenta a sequência para a criação da nova versão de cada um dos itens que representam as partes.

Para criar uma nova versão das partes e conjunto, o gerente seleciona os respectivos *Item Revision* e solicita revisar. O Teamcenter® requisita que o gerente determine quais objetos serão copiados para a nova versão. O gerente decide não copiar os objetos anexados aos itens da versão anterior, visto que

não tem a intenção de modificar nenhum dos objetos inseridos na versão de princípios de solução e os objetos da versão anterior dos itens podem ser visualizados a qualquer momento do processo.

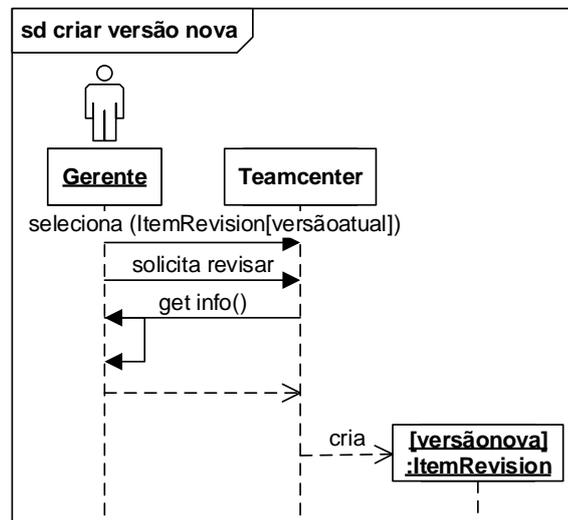


Figura 29: Sequência de mensagens para criar versão nova.

O projetista inicia a atividade de determinação da arquitetura preliminar do produto. Ele identifica os requisitos de geometria e arquitetura do produto e os parâmetros do produto determinados no projeto conceitual inseridos nos documentos (3).

Ele cria um modelo geométrico do produto no NX[®] (4). Para isso, ele inicia pelas peças principais e depois as auxiliares. Uma sequência de mensagens entre os objetos durante a modelagem da geometria é apresentada na Figura 30.

Para iniciar a modelagem geométrica das peças do produto, o projetista seleciona o item que representa peça na estrutura do produto que ele pretende modelar e solicita ao Teamcenter[®] acesso ao NX[®].

No NX[®] ele seleciona a opção de tipo de *template modeling*. Uma sessão do NX[®] é iniciada para que o projetista comece a modelar o produto. O NX[®] exibe as ferramentas para a criação das *features* e três abas no painel de navegação (*part navigator*): uma árvore que deverá apresentar a sequência das operações realizadas para a criação da parte, outra apresentará os detalhes dos parâmetros

e seus respectivos valores e outra exibirá as dependências entre as *features* criadas. As dependências representam a relação hierárquica de uma *feature* criada a partir da outra. Essa relação é denominada relação pai-filho (*parent-child*), e é relevante porque, dependendo do tipo de alteração realizada na *feature* pai, a *feature* filho também será alterada.

Além dos objetos geométricos, restrições que controlam a dimensão e a posição das *features* são também adicionadas e exibidas em uma aba do painel de navegação denominada restrições (*constraints*).

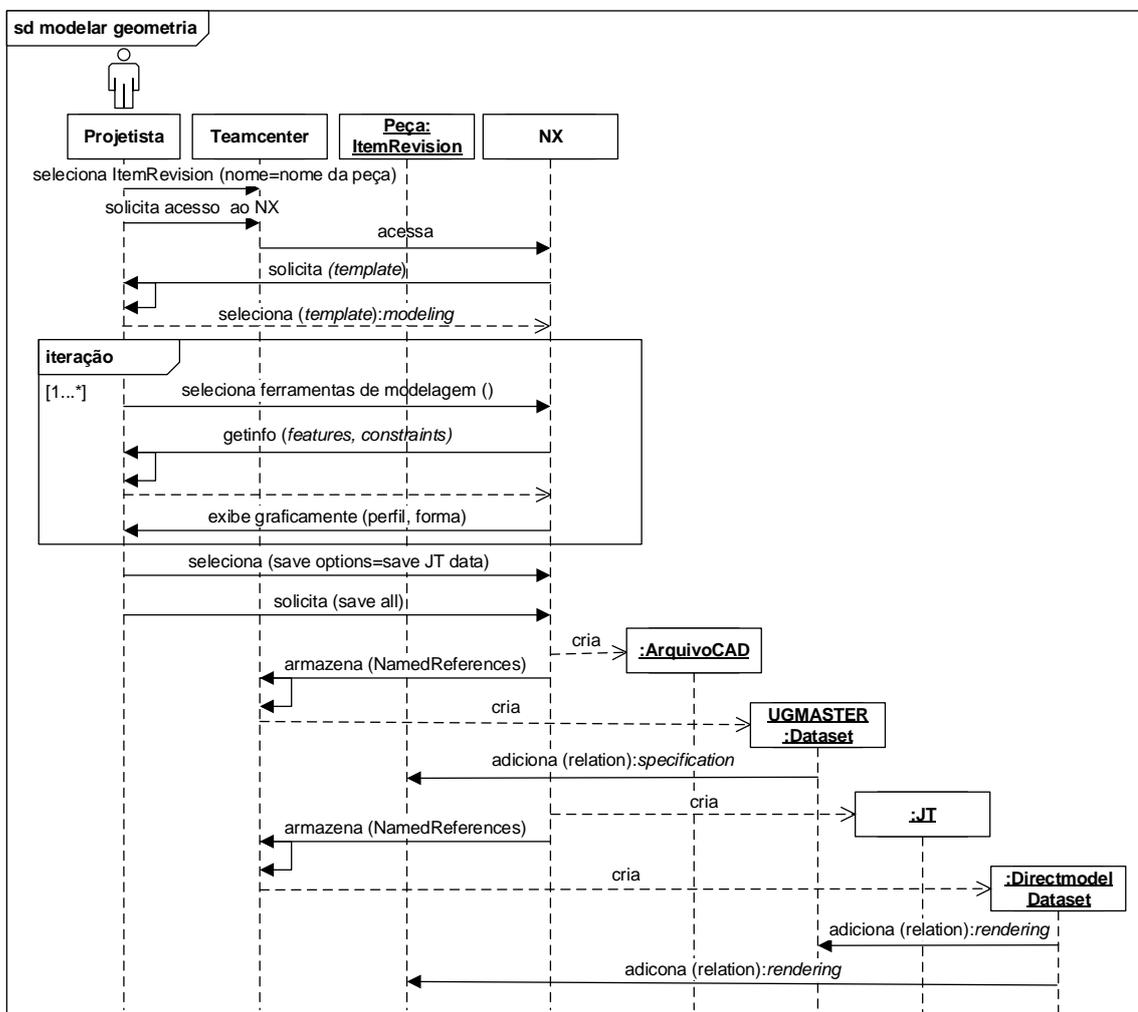


Figura 30: Sequência de mensagens para modelar geometria de uma peça.

Enquanto o projetista modela o produto, o NX[®] processa as intersecções entre as *features* para o método de representação pelas bordas (B-Rep) e a geometria da peça é exibida graficamente ao projetista. Para modelar uma peça, geralmente são criadas várias *features* e restrições. Portanto, esse processo deve ocorrer várias vezes como representado no diagrama da Figura 30 com o quadro iteração, significando que o número de interações é indeterminado, ou seja, ocorre de acordo com a peça que será criada.

Com a finalização do modelo, o projetista acessa o menu do NX[®] para configurar as opções de salvamento. Ele seleciona “salvar os dados JT” para que um arquivo JT seja armazenado no Teamcenter[®].

O projetista solicita ao NX[®] salvar o arquivo. O NX[®] gera dois arquivos, um arquivo CAD e um JT, e esses arquivos são armazenados no *Vault* do Teamcenter[®]. São instanciados dois *datasets* para representar esses documentos no ambiente do Teamcenter[®]: o UGMASTER para o arquivo CAD e o DirectModel para o arquivo JT. Ambos são relacionados ao item da peça, mas o tipo de relação difere: o arquivo CAD é relacionado como especificação (*specification*) e o JT como renderização (*rendering*). O JT é relacionado ao *dataset* UGMASTER também, com a mesma relação *rendering*.

O projetista cria uma referência de cada peça na pasta do produto⁹. Ele inicia então a combinação das peças no conjunto completo (5 e 6) (Figura 31). A combinação é feita a partir do item raiz do produto. O projetista combina todos os modelos das peças do cilindro pneumático no modelo de conjunto referenciado no item raiz.

A partir dos parâmetros funcionais definidos no projeto conceitual e do modelo geométrico construído, o projetista realiza análises estruturais e identifica os parâmetros do modelo de comportamento para as peças do cilindro.

⁹ A partes do produto foram criadas a partir do item raiz, seguindo uma abordagem *top-down*, porém para que as partes sejam visíveis para a montagem de conjuntos no NX[®] elas precisaram estar referenciadas em uma pasta. Nesse trabalho elas foram referenciadas na pasta do produto.

Ao final da atividade, o projetista confirma o término da atividade via *workflow*. O gerente avalia o produto e aprova. Um *status* de liberação é associado a cada um dos itens da estrutura usando o mesmo procedimento que foi anteriormente apresentado na Figura 27.

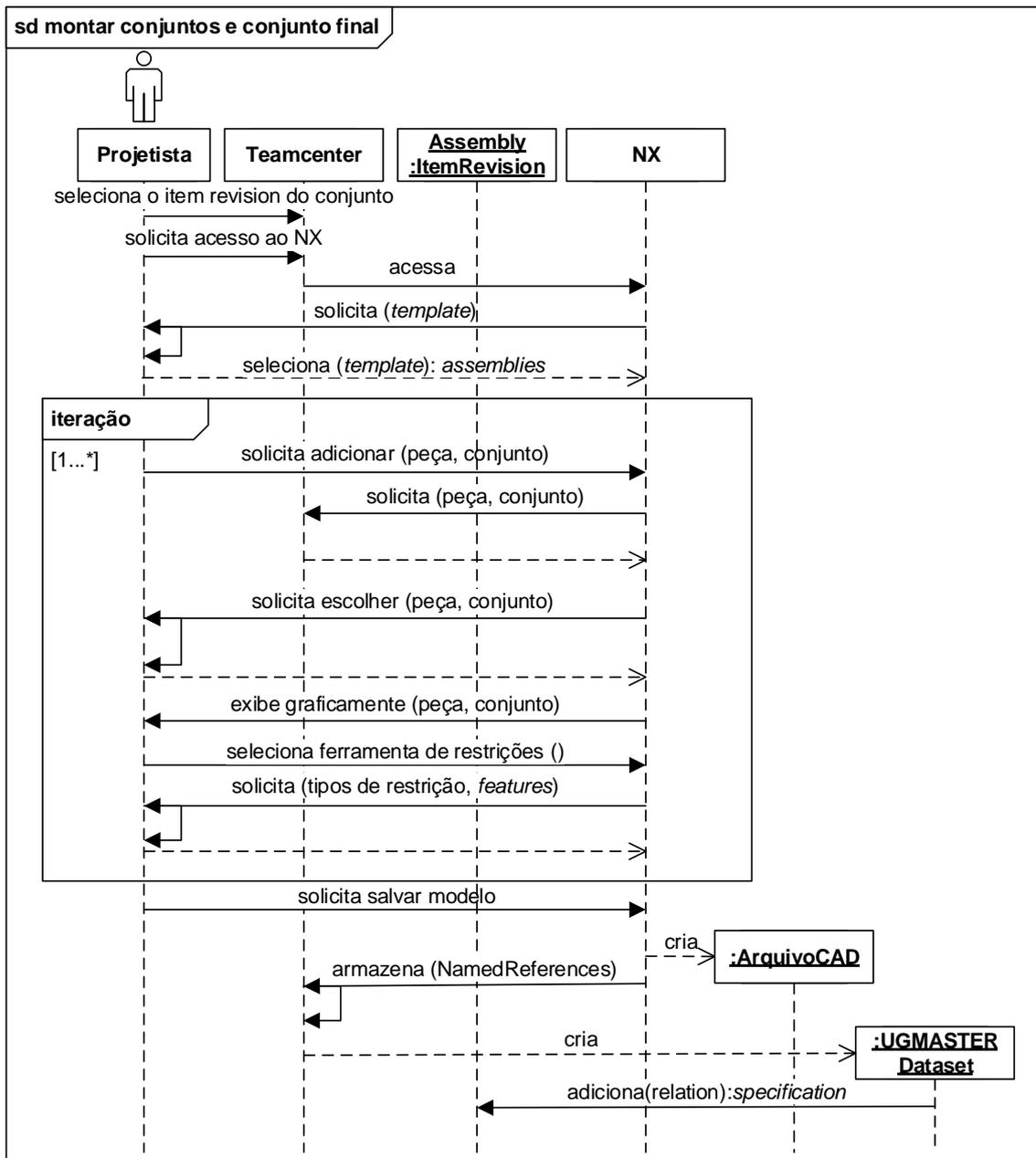


Figura 31: Sequência de mensagens para a montagem de conjuntos.

5.5. Projeto detalhado

A Figura 32 apresenta a visão do *framework* para a etapa de projeto detalhado. Primeiramente, o gerente cria uma nova versão dos itens das partes e conjunto completo – usando o mesmo procedimento apresentado na Figura 29 (1). Ele determina que somente os objetos *datasets* UGMASTER e DirectModel serão copiados para a versão final do produto.

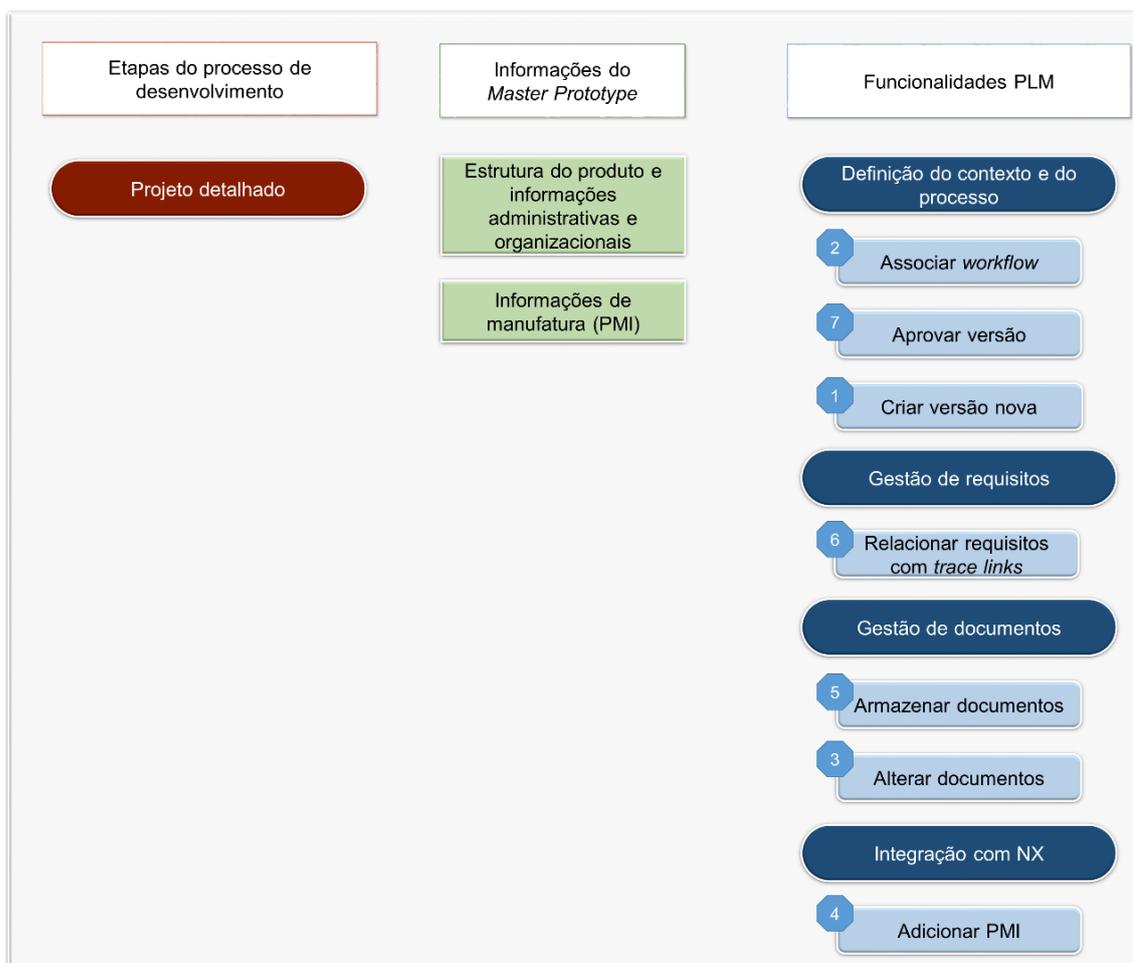


Figura 32: Representação de *framework* para etapa de projeto detalhado.

As atividades de finalizar e completar o produto são atribuídas ao projetista, via *workflow* (2). Para finalizar e completar, o projetista deve adicionar as informações PMI de cada peça do produto e para isso precisa acessar o NX para alterar o documento da peça (3). A sequência de operações para adicionar as informações de manufatura no está representada na Figura 33 (4).

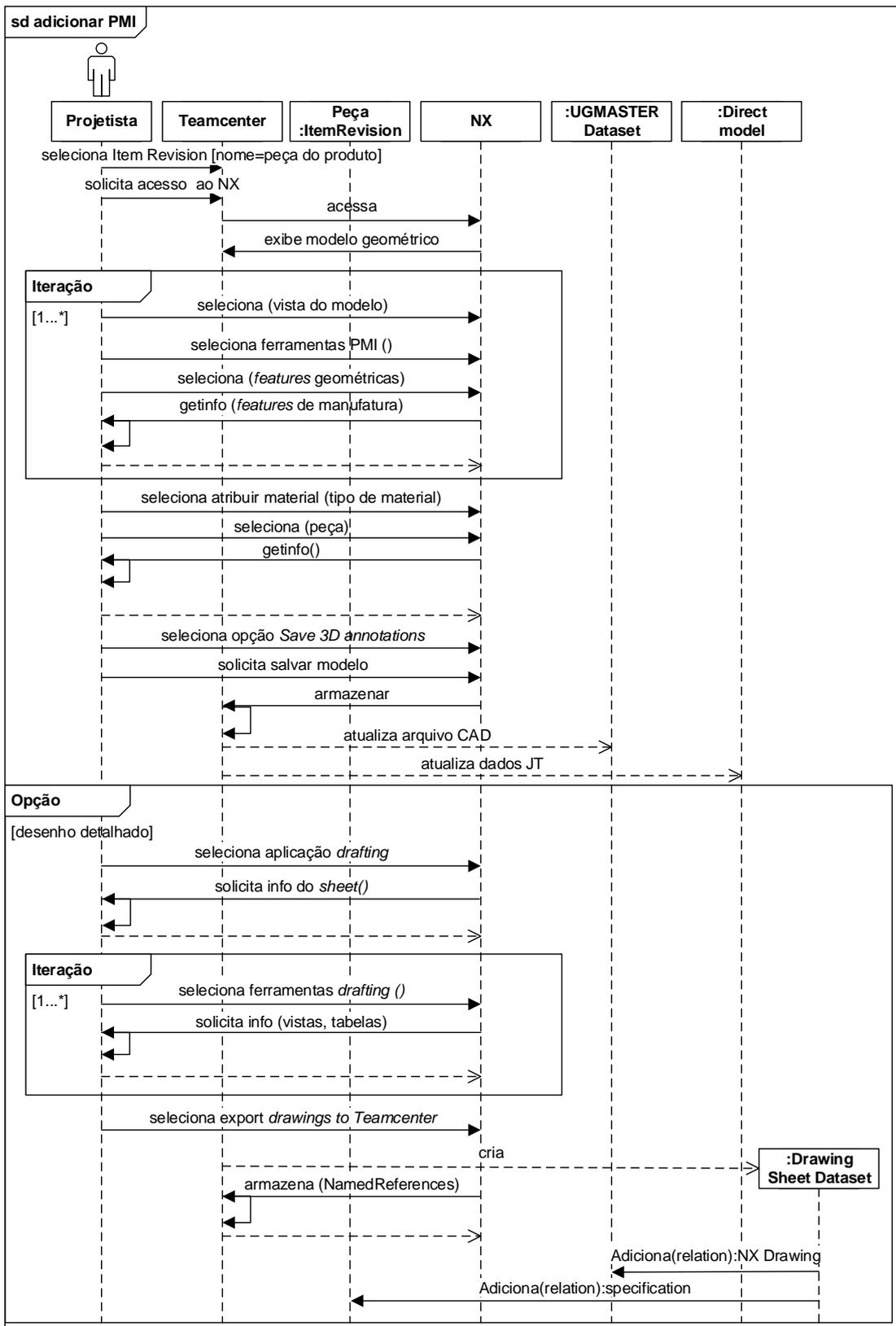


Figura 33: Seqüência de mensagens para adicionar PMI.

Para criar *features* de manufatura, o projetista seleciona o *Item Revision* da peça e acessa o NX®. O sistema exibe graficamente o modelo geométrico da peça já construído. O projetista escolhe uma perspectiva do produto que será usada para posicionar as *features* de manufatura. De antemão prevê quais as vistas e cotas do modelo serão necessárias para a criação do desenho detalhado.

O projetista seleciona as ferramentas na aba PMI e seleciona cada uma das *features* geométricas ou bordas que representarão os elementos tolerados, ou seja, aquelas que vão receber a definição de tolerância. O NX® solicita os parâmetros das *features* de manufatura. Esse fragmento de operações ocorre várias vezes até que todas as *features* de manufatura sejam definidas no modelo da peça, como representado pelo quadro iteração.

No momento em que as *features* de manufatura são instanciadas, elas são exibidas no *Part Navigator* (navegador de partes) em uma árvore individual. São também associadas à vista que foi escolhida do modelo. Os elementos tolerados são exibidos como pais das *features* de manufatura na aba de dependência. Informações quanto aos parâmetros e seus valores são exibidos na aba detalhes.

Quando o projetista termina de definir as PMI do produto, ele seleciona a opção de *save 3D annotations* (salvar os dados 3D) no menu do NX® em *save options* (opções de salvamento) (5). Como resposta, o NX® armazena as alterações do modelo no Teamcenter® e atualiza os *datasets* UGMASTER e Direct Model.

O projetista cria o desenho detalhado da peça. A adição das *features* de manufatura no modelo tridimensional anteriormente facilitou a criação do desenho detalhado. Para isso, o projetista seleciona o modo *drafting* do NX®. Ele seleciona a escala, depois as vistas e tabelas que gostaria de adicionar. Essas operações são realizadas até que a definição do produto esteja completa no desenho.

Tendo completado o desenho, o projetista seleciona a opção de *export drawings to Teamcenter* (exportar desenhos para o Teamcenter). Um *dataset* específico

para desenho é instanciado e ele pode então ser visualizado diretamente no Teamcenter®, conforme mostrado na Figura 34.

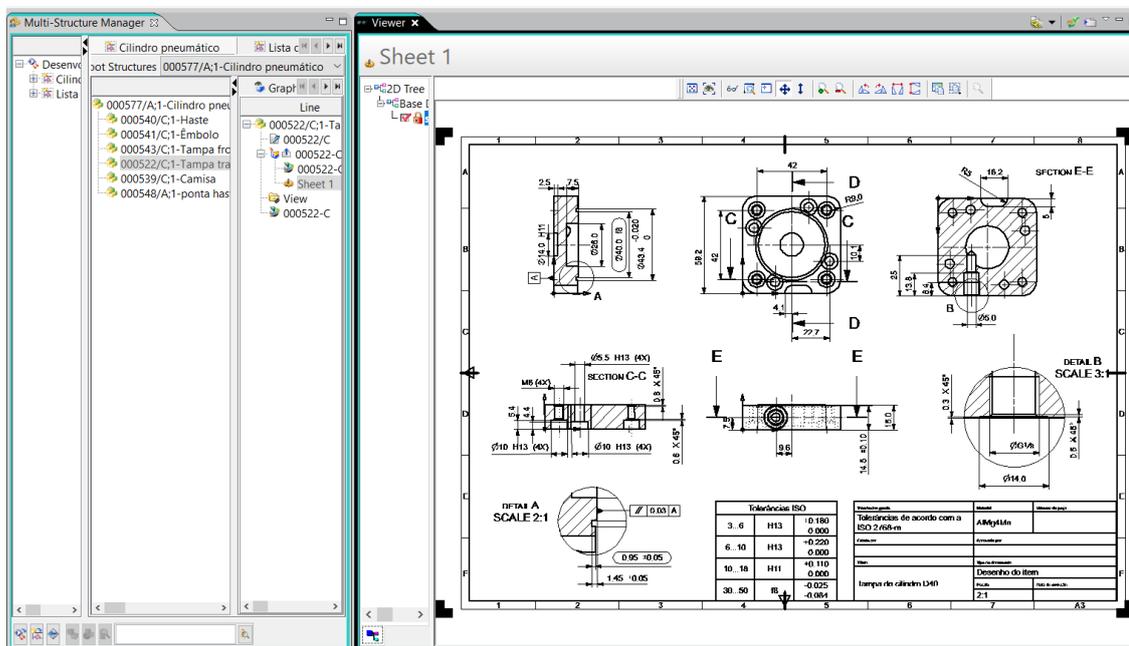


Figura 34: Visualização do desenho detalhado da tampa do cilindro pneumático no Teamcenter®.

O projetista relaciona os requisitos do produto com as partes que atendem a cada requisito (Figura 35) (6). Para os tópicos da lista de requisitos são definidos pontos de *definition* (definição), e às partes do produto, pontos de *compliant* (conformidade).

O gerente avalia a documentação final e aprova o produto final (7). Um *status* de liberação é atribuído a cada um dos itens da estrutura do produto. O produto está liberado para o planejamento da produção.

A Figura 36 apresenta o conjunto final do cilindro pneumático (à esquerda) e a estrutura do *Item Revision* da tampa do cilindro (à direita). Pode-se observar que a estrutura do *Item Revision* da peça “tampa traseira” corresponde a versão C.

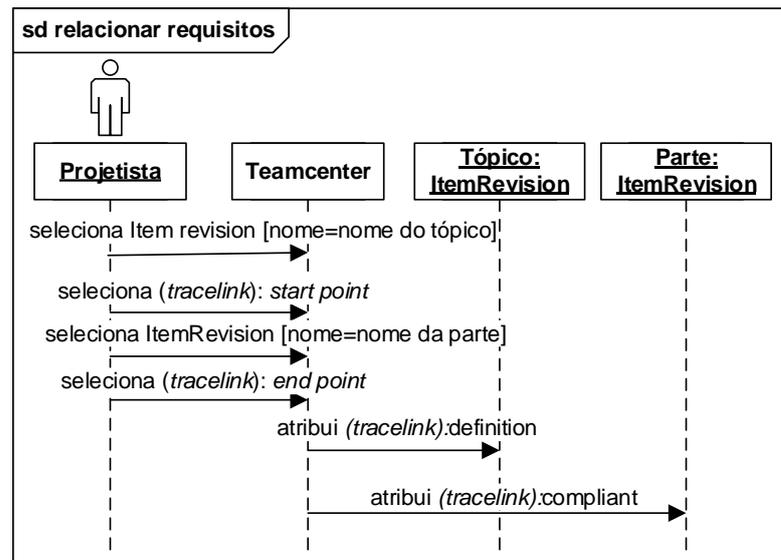


Figura 35: Sequência de mensagens para a criação de *trace links*.

A composição final da estrutura do item da peça compreende o desenho detalhado na segunda linha abaixo do *Item Revision*, com uma relação de *specification*. O arquivo CAD logo abaixo, armazenado no *dataset* com a mesma relação. O modelo JT com a relação de *rendering*. E por último o *status* de *release* (liberação).

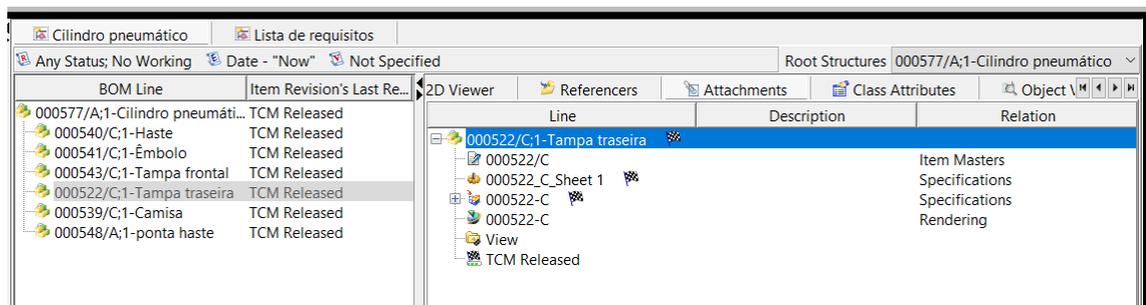


Figura 36: Estrutura do cilindro pneumático e do *Item Revision* da tampa traseira.

A Figura 37 mostra o modelo geométrico da tampa traseira do cilindro pneumático. Essa visualização é possível devido a conversão do formato CAD para JT. Ambos os formatos são armazenados no Teamcenter®.

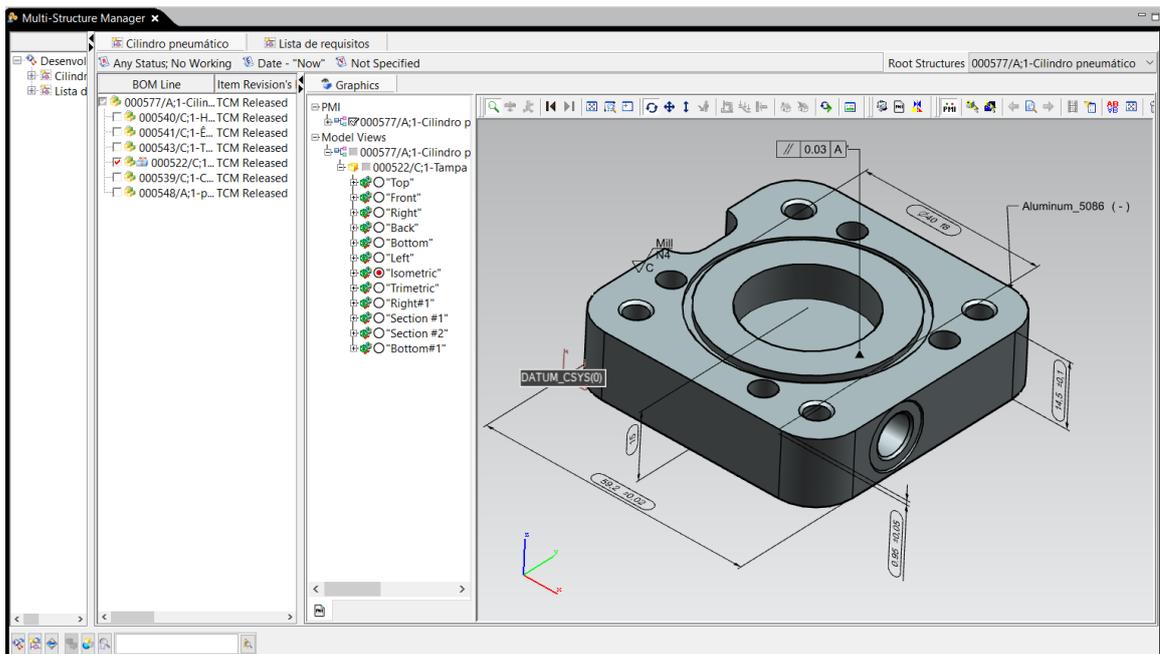


Figura 37: Visualização da geometria, PMI e da estrutura da tampa traseira no Teamcenter®.

6. Conclusões

Os avanços recentes nas tecnologias de comunicação e informação têm contribuído com o uso de modelos de informações do produto cada vez mais estruturados. O modelo de dados ICDM se baseia no modelo de dados integrado do produto STEP para a especificação do produto. Essa especificação se aplica ao conceito de classe de acordo com o que foi proposto no RAMI. Parte-se do princípio de que essa mesma especificação pode ser aplicada para a criação de um *Master Prototype* do produto, de acordo com o que foi encontrado na literatura, no que diz respeito ao Gêmeo Digital.

O Teamcenter® é uma ferramenta de PLM que suporta a criação de modelos de informações do produto dando significado para essas informações em um ambiente integrado e ao longo do ciclo de vida de um produto. Visto que a ferramenta fornece suporte para a gestão de informações e processos ao longo de ciclo de vida do produto, tem potencial para ser um meio de implementar o conceito do Gêmeo Digital. A fim de contribuir com essa ideia foram verificados quais os procedimentos usados para a criação das informações do *Master Prototype* durante o processo de desenvolvimento do produto.

Durante a verificação puderam ser identificadas as principais funcionalidades do Teamcenter® que determinam as informações do *Master Prototype*. Com a funcionalidade “criar itens” e “estruturar”, os itens foram criados para representar as partes de um produto e serem combinados em uma estrutura. Dessa forma foi possível criar uma visão do produto. Informações adicionais referentes a cada um dos itens puderam ser incluídas na estrutura individual de cada *item revision*. Isso possibilitou a organização das informações além de permitir trabalhar com cada peça de maneira integrada com as outras peças da estrutura e também individualmente, de acordo com a necessidade durante o processo.

Com a funcionalidade “armazenar documentos”, os documentos foram armazenados e referenciados por *datasets*. Os *datasets* foram relacionados aos itens das partes a que se referem. As ferramentas associadas a cada um dos

datasets permitiram acessar os arquivos armazenados. Algumas ferramentas puderam ser acessadas de forma integrada, como no caso do NX[®]. A partir da integração do Teamcenter[®] com o NX[®] foi possível definir a *features* geométricas e de manufatura e armazená-las automaticamente.

Por meio do visualizador de arquivos JT foi possível visualizar os documentos e a representação geométrica no formato JT diretamente no Teamcenter[®]. As *features* geométricas puderam ser identificadas no modelo e as *features* de manufatura também puderam ser acessadas e visualizadas claramente relacionadas as *features* geométricas do modelo.

As informações administrativas e organizacionais, que determinam a configuração do produto, como a identificação individual das partes, *status* de liberação e versões puderam ser geradas por meio das funcionalidades “aprovar item” e “criar uma nova versão”. Algumas funcionalidades adicionais, serviram de apoio ao processo e a configuração da exibição das partes do produto e da estrutura. Foram elas: “associar workflow”, criar visão” e “configurar estrutura”. As informações administrativas e organizacionais de cada item puderam ser visualizadas nas colunas da estrutura do produto no *Multi-Structure Manager*.

Como os requisitos foram representados por itens, puderam ser individualmente relacionados a cada uma das partes que atenderam a esses requisitos por meio de *trace links*. Essas relações puderam ser visualizadas no *data panel* (painel de dados) do *Multi-Structure Manager*.

Portanto, com os resultados da verificação do *framework*, pode-se dizer que o desenvolvimento do *Master Prototype*, no Teamcenter[®], tem potencial para atender ao modelo de referência proposto por Schleich et. al. [12] como um operador de especificação. Esse é um dos requisitos apresentados para esse *framework*. Isso porque o desenvolvimento de um *Master Prototype* compreende as operações necessárias para criar o modelo abstrato e conceitual do produto.

Por outro lado, fica como sugestão para pesquisas futuras determinar no Teamcenter[®] como esse modelo poderia atender aos requisitos para o operador

de verificação, que diz respeito as operações realizadas para verificar as especificações nas partes reais do produto.

Apesar de não ser o foco desse trabalho a instanciação do *Master Prototype*, algumas limitações podem ser salientadas. Por exemplo, não foram encontradas soluções voltadas para a criação de instâncias de *Master Prototype*, de acordo com os requisitos de modelo representação do Gêmeo Digital do produto, no Teamcenter®.

Ainda assim foram identificados alguns meios que se aproximam do que é esperado para a criação de uma instância do *Master Prototype*. No sistema é possível criar uma cópia do produto a partir de um *template* de uma estrutura do produto. Por exemplo, é possível criar uma (ou várias) cópias idênticas da estrutura do *Master Prototype*, por meio de um recurso do Teamcenter® denominado clonagem. Essa cópia representaria uma instância do *Master Prototype* e, apesar de ser independente, seria possível manter uma referência ao *Master Prototype* durante todo o ciclo de vida da instância.

Um meio de encontrar uma solução mais aproximada ainda poderia ser explorar os recursos de visualização, como as ferramentas do módulo *Lifecycle Viewer*, buscando entender como as informações do gêmeo físico poderiam ser relacionadas a sua respectiva representação no modelo JT. Podem ser também explorados os recursos de personalização do modelo de objetos de negócio do Teamcenter®, o BMIDE (*Business Modeler IDE*).

No que diz respeito a estrutura podem ser explorados os módulos do Teamcenter® *As Built Manager*, que instancia uma estrutura de produto configurada de acordo com um produto físico instanciado e o *Service Manager* que gerencia a configuração da estrutura de um produto “como mantida” ao longo do ciclo de vida.

Uma outra proposta de pesquisa poderia ser explorar como os modelos de simulação de comportamento do produto poderiam ser mapeados no *Master*

Prototype do produto. Por ser um assunto amplo, esse aspecto do *Master Prototype* não foi investigado, sendo, portanto, uma limitação desse trabalho.

Seria interessante investigar o mapeamento dos aspectos de comportamento do produto, visto que são altamente relevantes para a fase de uso do produto, uma vez que são usados como base para a simulação de dados reais de desempenho e comportamento do produto no ambiente. O Teamcenter® dispõe de soluções para modelos de simulação. O módulo CAE Manager pode ser explorado a fim de mapear as informações de comportamento das instâncias físicas do produto durante o seu ciclo de vida.

Esse trabalho pode ser aproveitado como ponto de partida para uma pesquisa sobre esse assunto, visto que, os modelos de representação de geometria são geralmente usados para a criação de modelos de comportamento do produto, como, por exemplo, para gerar os modelos de elementos finitos.

7. Referências

- /1/ PAHL, G. et al. *Engineering Design: A systematic Approach*. 3ª edição. ed. [S.l.]: Springer, 2007.
- /2/ ROZENFELD, H. et al. *Gestão de Desenvolvimento de Produto: Uma Referência para a Melhoria do Processo*. São Paulo: Saraiva, 2006.
- /3/ ANDERL, R.; TRIPPNER, D. *STEP STandard for the Exchange of Product Model Data*. Leipzig: B.G. Teubner Stuttgart, 2000.
- /4/ SAAKSVUORI, A.; IMMONEN, J. *Product Lifecycle Management*. 3. ed. Berlin: Springer, 2008.
- /5/ ZANCUL, E. D. S. *Gestão do ciclo de vida de produtos: seleção de sistemas PLM com base em modelos de referência*. Universidade de São Paulo. São Carlos, p. 227. 2009.
- /6/ ZAMAZAL, K.; DINGER, A. Product lifecycle management in automotive industry. *Systems Engineering for Automotive Powertrain Development*, 2020. 1-27.
- /7/ ANDERL, R. *Industrie 4.0 - Digital Transformation in Product Engineering and Production*. 21st International Seminar on High Technology - Smart Products and Smart Production. Piracicaba: [s.n.]. 2016. p. 15.
- /8/ VDI/VDE. *Reference Architecture Model Industrie 4.0 (RAMI4.0)*. Berlin, p. 25. 2015.
- /9/ PICARD, A.; ANDERL, R. *Integrated component data model for smart production planning*. 19º Seminário Internacional de Alta Tecnologia. Santa Bárbara d'Oeste: -. 2014. p. 6.
- /10/ PICARD, A. C. R. *Integriertes Werkstückinformationsmodell zur Ausprägung werkstückindividueller Fertigungszustände*. Technischen Universität Darmstadt. Darmstadt, p. i-183. 2015.
- /11/ STARK, R.; DAMERAU, T. Digital Twin. *CIRP Encyclopedia of Production Engineering*, 1, 2019. 1-8.
- /12/ SCHLEICH, B. et al. Shaping the digital twin for design and production engineering. *Cirp Annals*, 66, n. 1, 2017. 141-144.
- /13/ ABRAMOVICI, M.; GÖBEL, J. C.; SAVARINO, P. Reconfiguration of Smart Products during their Use Phase based on Virtual Product Twin. *CIRP Annals Manufacturing Technology*, 66, n. 1, 2017. 165-168.

- /14/ DURÃO, L. F. C. S. et al. Digital Twin Requirements in the Context of Industry 4.0. *IFIP International Federation for Information Processing*, 540, 2018. 204-214.
- /15/ DAVID, M.; ROWE, F. What does PLMS (product lifecycle management systems) manage: Data or documents? Complementarity and contingency for SMEs. *Computers in Industry*, 75, 2015. 140-450.
- /16/ SALLATI, C. *Proposta de um framework conceitual para o desenvolvimento de produtos inteligentes para idosos no contexto da indústria 4.0*. Universidade Metodista de Piracicaba. Santa Bárbara d'Oeste. 2019.
- /17/ STARK, J. *Product Lifecycle Management (Volume 1)*. 4. ed. Switzerland: Springer International Publishing, v. 1, 2020.
- /18/ VENGHAUS, E.; STARK, R. Understanding PLM and PLM Customizing: A Theoretical Fundament for a Conceptual Approach. *IFIP International Federation for Information Processing*, 540, 2018. 670-680.
- /19/ PINNA, C. et al. Effect of product lifecycle management on new product development Effect of product lifecycle management on new product development. *Computers in Industry*, 100, 2018. 184-195.
- /20/ ABRAMOVICI, M.; AIDI, Y. Next Generation Product Lifecycle Management (PLM). In: FATHI, M. *Integration of Practice-Oriented Knowledge Technology: Trends and Perspectives*. Siegen: Springer, 2013. p. 143-156.
- /21/ VIEIRA, D. R.; BOURAS, A.; DEBAECKER, D. *Gestão de projeto do produto*. Rio de Janeiro: Elsevier, v. 1, 2013.
- /22/ ALEMANNI, M.; DESTEFANIS, F.; VEZZETTI, E. Model-based definition design in the product lifecycle management scenario. *The International Journal of Advanced Manufacturing Technology*, 52, 2011. 1-14.
- /23/ BECKERS, R. et al. Interoperability and Visualization of Complex Products Based on JT Standard. *Transdisciplinary Engineering: Crossing Boundaries*, 4, 2016. 828-837.
- /24/ NZETCHOU, S. et al. Review of CAD Visualization Standards in PLM. *IFIP International Federation for Information Processing*, 565, 2019. 34-43.
- /25/ STEKOLSCHIK, A. *Approach to structure management and customer-related configuration in the product lifecycle in mechanical engineering*. Materials Science and Engineering. Shanghai: IOP. 2020. p. 6.
- /26/ CAMBA, J. D. et al. On the integration of model-based feature information in Product Lifecycle Management systems. *International Journal of Information Management*, 37, n. 6, 2017. 611-621.

- /27/ FRECHETTE, S.; HUANG, P.; CARLISLE, M. *Model Based Enterprise Technical Data Package Requirements*. National Institute of Standards and Technology. Gaithersburg, p. 100. 2011.
- /28/ SHEHAB, E. et al. Model-Based Enterprise Framework for Aerospace Manufacturing Engineering. *Advances in Manufacturing Technology XXXIII*, 2019.
- /29/ PFOUGA, A.; STJEPANDIC, J. Leveraging 3D geometric knowledge in the product lifecycle based on industrial standards. *Journal of Computational Design and Engineering*, 5, 2018. 54-67.
- /30/ SATO, R. S. *Diretrizes para Assegurar a Qualidade da Informação da Manufatura de Produtos*. Universidade Tecnológica Federal do Paraná. Curitiba. 2018.
- /31/ SALLATI, C.; BERTAZZI, J. D. A.; SCHÜTZER, K. Professional skills in the Product Development Process: the contribution of learning environments to professional skills in the Industry 4.0 scenario. *Procedia CIRP*, 84, 2019. 203-208.
- /32/ HERMANN, M.; PENTEK, T.; BORIS, O. *Design Principles for Industrie 4.0 Scenarios: A Literature Review*. Technische Universität Dortmund. Dortmund, p. 16. 2015.
- /33/ HERMANN, M.; PENTEK, T.; OTTO, B. *Design Principles for Industrie 4.0 Scenarios*. 49th Hawaii International Conference on System Sciences. Koloa: IEEE. 2016. p. 1-10.
- /34/ SOARES, M. F. *Proposta de Modelo para Gerenciamento de Informações Relacionadas aos Processos de Montagem Inteligente*. Universidade Metodista de Piracicaba. Santa Barbara D'Oeste. 2018.
- /35/ VDE/DKE. *Industrie 4.0 German Standardization Roadmap*. Germany, p. 146. 2018.
- /36/ OZTEMEL, E.; GURSEV, S. Literature review of Industry 4.0 and related technologies. *Journal of Intelligent Manufacturing*, 2020. 127-182.
- /37/ PERALES, D. P.; VALERO, F. A.; GARCÍA, A. B. Industry 4.0: A Classification Scheme. *Closing the Gap Between Practice and Research in Industrial Engineering (Lecture Notes in Management and Industrial Engineering)*, n. Springer, 2018. 343-350.
- /38/ WANG, S. et al. Towards Smart Factory for Industry 4.0: A Self-organized Multi-agent System with Big Data Based Feedback and Coordination. *Computer Networks*, 2016.
- /39/ ALCÁCER, V.; CRUZ-MACHADO, V. Scanning the Industry 4.0: A Literature Review on Technologies for Manufacturing Systems. *Engineering Science and Technology, an International Journal*, 22, n. 3, 2019. 899-919.

- /40/ BITKOM, VDMA, ZVEI. *Implementation Strategy Industrie 4.0. Plattform Industrie 4.0.* [S.l.], p. 96. 2016.
- /41/ PLATTFORM INDUSTRIE 4.0. *Relationships between I4.0 Components – Composite Components and Smart Production.* [S.l.]. 2017.
- /42/ CHILWANT, N.; KULKARNI,. Open Asset Administration Shell for Industrial Systems. *Manufacturing Letters*, 20, 2019. 15-21.
- /43/ STRANG, D.; ANDERL, R. *Assembly Process driven Component Data Model in Cyber-Physical Production Systems.* Proceedings of the World Congress on Engineering and Computer Science. San Francisco: [s.n.]. 2014.
- /44/ ANDERL, R. et al. Integriertes Bauteildatenmodell für Industrie 4.0 - Informationsträger für Cyber Physische Systeme. *cyber-physische produktionssysteme*, 2014.
- /45/ WAHLSTER, W. The Semantic Product Memory: An Interactive Black Box for Smart Objects. In: WOLFGANG, W. *SemProm: Foundations of Semantic Product Memories for the Internet of Things.* Saarbrücken, Germany: Springer, 2013. p. 3-20.
- /46/ DONOGHUE, I. et al. The Benefits and Impact of Digital Twins in Product Development Phase of PLM. *IFIP Advances in Information and Communication Technology*, 540, 2018. 432-441.
- /47/ SCHÜTZER, K. et al. Contribution to the development of a Digital Twin based on product lifecycle to support the manufacturing process. *Procedia CIRP*, 84, 2019. 82-87.
- /48/ RÍOS, J. et al. Product Avatar as Digital Counterpart of a Physical Individual Product: Literature Review and Implications in an Aircraft. *Advances in Transdisciplinary Engineering*, 2, 2015. 657-666.
- /49/ THARMA, R.; WINTER, R.; EIGNER, M. *An approach for the implementation of the digital twin in the automotive wiring harness field.* DS 92: Proceedings of the DESIGN 2018 15th International Design Conference. Dubrovnik: Design society. 2018. p. 3023-3032.
- /50/ ABRAMOVICI , M.; GÖBEL, J. C.; SAVARINO, P. Virtual Twins as Integrative Components of Smart Products. *IFIP International Federation for Information Processing 2016*, 492, 2016. 217–226.
- /51/ SHAFTO M. ET AL. *DRAFT Modeling, Simulation, Information Technology & Processing Roadmap. National Aeronautics and Space Administration.* National Aeronautics and Space Administration. Washington, p. 27. 2010.

- /52/ GRIEVES, M.; VICKERS, J. V. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. *Transdisciplinary Perspectives on Complex Systems*, 2017. 85-113.
- /53/ BOSCHERT, S.; ROSEN, R. Digital Twin—The Simulation Aspect. In: BOSCHERT, S.; ROSEN, R. *Mechatronic Futures*. Switzerland: Springer, 2016. Cap. 5, p. 59-74.
- /54/ VOELL, C. et al. How Digital Twins Enable the Next Level of PLM – A Guide for the Concept and the Implementation in the Internet of Everything Era. *IFIP International Federation for Information Processing*, 540, 2018. 238-249.
- /55/ WANG, Z. Digital Twin Technology. In: BÁNYAI, T. *Industry 4.0: Impact on Intelligent Logistics and Manufacturing*. [S.l.]: IntechOpen, 2020.
- /56/ EISENTRÄGER, M. et al. *Changeability in Engineering: Symbioses of Agile Methodologies and Virtual Engineering*. International Conference on Engineering, International Conference on Engineering. Stuttgart: IEEE. 2018. p. 1-8.
- /57/ BOSCHERT, S.; HEINRICH, C.; ROSEN, R. *Next Generation Digital Twin*. Proceedings of TMCE. Las Palmas de Gran Canaria: TMCE 2018. 2018. p. 209-217.
- /58/ ROSEN, R. et al. About The Importance of Autonomy and Digital Twins for the Future of Manufacturing. *IFAC-PapersOnLine*, 48, n. 3, 2015. 567-572.
- /59/ TETSUO, T. et al. Development capabilities for smart products. *CIRP Annals - Manufacturing Technology*, 68, 219. 727-750.
- /60/ LACERDA, D. P. et al. Design Science Research: método de pesquisa para a engenharia de produção. *Gestão da produção*, 20, 2013. 741-761.
- /61/ JOHANNESSON, P.; PERJONS, E. A Method Framework for Design Science Research. In: JOHANNESSON, P.; PERJONS, E. *An Introduction to Design Science*. Switzerland: Springer, 2014. Cap. 4, p. 75-89.
- /62/ CHANG, K.-H. *Product Design Modeling using CAD/CAE The Computer Aided Engineering Design Series*. 1. ed. Amsterdam: Elsevier, 2014.
- /63/ ZHAN, X.; LI, X.; LUI, X. *Study on Product Data Structure and Realization of Model Based on PLM*. Intelligence Computation and Evolutionary Computation. Wuhan: Springer. 2012. p. 1001-1008.
- /64/ KÖNIGS, S. F. et al. Traceability in Systems Engineering – Review of industrial practices, state-of-the-art technologies and new research solutions. *Advanced Engineering Informatics*, 26, n. 4, 2012. 924-940.

/65/ PEREIRA, L. A. D. M. *Análise e Modelagem de Sistemas com a UML*. 1. ed. Rio de Janeiro: [s.n.], v. 1, 2011.

/66/ SEIDL, M. et al. *UML @ Classroom An Introduction to Object-Oriented Modeling*. 1. ed. Heidelberg: Springer, v. 1, 2012.