MIGUEL ANGEL ORELLANA POSTIGO

Método para o Design Integrado de Sistemas Smart Grid: uma abordagem sistêmica baseada em Modelos

São Paulo

2021

MIGUEL ANGEL ORELLANA POSTIGO

Método para o Design Integrado de Sistemas Smart Grid: uma abordagem sistêmica baseada em Modelos

Versão Corrigida

Tese apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Doutor em Ciências.

Área de Concentração: Engenharia de Controle e Automação Mecânica.

Orientador: Prof. Dr. José Reinaldo Silva

Coorientador: Prof. Dr. Eduardo Lorenzetti Pellini

São Paulo

2021

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, _____ de _____ de _____

Assinatura do autor: _____

Assinatura do orientador: _____

Catalogação-na-publicação

Orellana Postigo, Miguel Angel
Método para o Design Integrado de Sistemas Smart Grid: uma abordagem sistêmica baseada em Modelos / M. A. Orellana Postigo, J. R. Silva, E. Lorenzetti Pellini -- versão corr. -- São Paulo, 2021.
127 p.

Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1.smart grid 2.systems design 3.model-based systems engineering 4.service systems 5.goal-oriented requirements engineering I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II.t. III.Silva, José Reinaldo IV.Lorenzetti Pellini, Eduardo

Este trabalho é dedicado à memória de minha avó Salomé Amado Valdivia, que sempre me iluminou e me deu forças nos momentos mais difíceis da minha vida.

AGRADECIMENTOS

Principal agradecimento a meu orientador o Prof. Dr. José Reinaldo Silva por sua confiança, oportuna orientação, e constante apoio durante o desenvolvimento deste trabalho.

Especial agradecimento ao Laboratório de Pesquisa Design-Lab e ao Departamento de Engenharia Mecânica da Escola Politécnica da Universidade de São Paulo pelos recursos fornecidos durante minha formação.

Aos professores do corpo docente do programa de pós-graduação da Universidade de São Paulo.

A minha mãe Haydee por ser meu farol e guia, a ela devo tudo o que sou.

A minha amada esposa Ana Maria, por ter me apoiado nessa empreendida, me dando força e inspiração para superar todos os obstáculos durante minha jornada, para realização desse sonho e superação dos desafios encontrados.

Dedico também as minhas filhas Mishell, Mellanie e Micknie, que sempre estiveram presentes me inspirando.

"Para nos tornarmos pessoas de mérito e de valor,
o que há de mais certo em nós é confiarmos em nós mesmos"
(Michelangelo)

RESUMO

ORELLANA POSTIGO, M.A.. **Método para o Design Integrado de Sistemas Smart Grid: uma abordagem sistêmica baseada em Modelos**. 2021. 127 f. Tese (Doutorado em em Ciências. Área de Concentração: Engenharia de Controle e Automação Mecânica) – Escola Politécnica (POLI/USP), São Paulo.

O design de sistemas modernos de energia aumentou em complexidade devido às suas novas características: sustentável, distribuído, heterogêneo, escalável, colaborativo, sempre preservando o alto desempenho. Novas abordagens, métodos, conceitos e ferramentas surgiram para caracterizar, compreender e especificar sistemas híbridos de produção/consumo de energia totalmente adaptados ao ambiente de entorno onde deveriam funcionar. Este trabalho apresenta uma proposta de método sistêmico para o desenvolvimento de requisitos aplicado a Smart Grid (SG), baseado em uma abordagem de serviço. Portanto, privilegia o acoplamento entre o sistema, o usuário e o domínio. A proposta enfatiza formalização, já na especificação de requisitos para redes distribuídas Smart Grid usando análise de workflow baseadas em Redes de Petri. A rede é sintetizada de um processo de Engenharia de Requisitos Orientada a Objetivos (GORE), baseada em diagramas semi-formais KAOS (Aquisição de Conhecimento de Especificações Objetivas), capaz de superar o dilema da abordagem funcional/ não-funcional, pelo menos na fase preliminar. O ciclo de requisitos consiste em eliciação, modelagem, análise, modelagem formal e verificação e pode ser iterado por refinamentos. O método foi aplicado a um estudo de caso real, baseado na evolução de uma microgrid instalada em uma comunidade isolada na floresta amazônica.

Palavras-chave: smart grid, systems design; model-based systems engineering; service systems; goal-oriented requirements engineering.

ABSTRACT

ORELLANA POSTIGO, M.A.. **Método para o Design Integrado de Sistemas Smart Grid: uma abordagem sistêmica baseada em Modelos**. 2021. 127 f. Tese (Doutorado em em Ciências. Área de Concentração: Engenharia de Controle e Automação Mecânica) – Escola Politécnica (POLI/USP), São Paulo.

The modern design of energy systems grew in complexity due to emerging characteristics as sustainability, distribution, heterogeneity, scalability, collaboration while preserving high performance. New approaches, methods, concepts, and tools appeared to characterize, understand and specify hybrid production/consumption grids wholly adapted to the application environment. This work presents a proposal for a service-oriented systemic method to develop requirements that could be applied to Smart Grids (SG). Therefore, the method prioritizes the coupling between system, users, and domain. The proposal also emphasizes a formal approach, used in the early phase and applied to a distributed network of SGs, based on workflow analysis and Petri Nets (PN). A PN schema is synthesized after a Requirements Engineering process that is also goal-oriented (GORE), using semi-formal KAOS (Knowledge Acquisition on Objective Specifications). The schema can obliviate the dilemma functional/non-functional, at least in the first steps. The requirements cycle proposed comprised the steps: elicitation, modeling, analysis, formal modeling, and verification and was applied to a case study of an SG in an isolated community in the Amazon forest.

Key-words: smart grid; systems design; model-based systems engineering; service systems; goal-oriented requirements engineering.

RESUMEN

ORELLANA POSTIGO, M.A.. **Método para o Design Integrado de Sistemas Smart Grid: uma abordagem sistêmica baseada em Modelos**. 2021. 127 f. Tese (Doutorado em em Ciências. Área de Concentração: Engenharia de Controle e Automação Mecânica) – Escola Politécnica (POLI/USP), São Paulo.

El diseño de sistemas energéticos modernos ha aumentado en complejidad debido a sus nuevas características: sustentables, distribuidas, heterogéneas, escalables, colaborativas, conservando siempre un alto rendimiento. Han surgido nuevos enfoques, métodos, conceptos y herramientas para caracterizar, comprender y especificar los sistemas híbridos de producción / consumo de energía totalmente adaptados alrededor donde deberían operar. Este trabajo presenta una propuesta de método sistémico para el desarrollo de requisitos aplicados a los Smart Grid (SG), basado en un enfoque de servicio. Para favorecer el acoplamiento entre sistema, usuario y dominio. La propuesta enfatiza la formalización, ya desde la fase de especificación de requisitos para redes distribuidas Smart Grid, utilizando análisis de workflow basado en Petri Nets. La red se sintetiza a partir de un proceso de Ingeniería de Requisitos Orientados a Objetos (GORE), basado en diagramas KAOS semiformales (Adquisición de Conocimientos de Especificaciones Objetivos), capaz de superar el dilema del enfoque funcional / no funcional, por lo menos en etapa preliminar. El ciclo de requisitos consiste en la obtención de requisitos, modelado, análisis, modelo formal y la verificación; y que puede repetirse mediante refinamientos. El método se aplicó a un estudio de caso real, basado en la evolución de una microrred instalada en una comunidad aislada de la selva amazónica.

Palabras clave: redes inteligentes, diseño de sistemas; ingeniería de sistemas basada en modelos; sistemas de servicio; ingeniería de requisitos orientada a objetivos.

LISTA DE ILUSTRAÇÕES

Figura 1 – Ciclo de Design-V 32
Figura 2 – Metodologia LCA 33
Figura 3 – Formalismo baseado em grafos 34
Figura 4 – Smart Grid Roadmap 36
Figura 5 – Arquitetura Intelligrid 38
Figura 6 – Arquitetura SGAM 39
Figura 7 – Mapeamento de Use-Case SGAM 39
Figura 8 – Proposta Design 44
Figura 9 – Smart Grid SoS 47
Figura 10 – Smart Grid 49
Figura 11 – Diagrama Unifilar de um SEP 50
Figura 12 – Smart Grid Presente e Futuro 51
Figura 13 – Arquitetura IEC 61850 53
Figura 14 – Classificação Objetivo 56
Figura 15 – Modelo KAOS 61
Figura 16 – Elementos KAOS 62
Figura 17 – Objetivos para o Elevador 63
Figura 18 – Objetivo solicitação de transporte 64
Figura 19 – Objetivo requisição de serviço 65
Figura 20 – Objetivo chamar ao elevador 65
Figura 21 – Modelo de responsabilidade 66
Figura 22 – Modelo de objetos 67
Figura 23 – Modelo de Operações 68
Figura 24 – RdP transição habilitada 69
Figura 25 – Ciclo Criativo 76
Figura 26 – Smart Grid SoS 78
Figura 27 – Domínio Usuário 79
Figura 28 – Ambiente de Modelagem 80
Figura 29 – Ciclo de desenvolvimento de requisitos. 81
Figura 30 – Algoritmo de transferência ReKPlan 84
Figura 31 – Microgrid Legada - Arquitetura AC-DC 88
Figura 32 – Sistema-as-is. 91
Figura 33 – Sistema-to-be (Principal) 94

Figura 34 – Refinamento Objetivos Informação 95
Figura 35 – Operação IEC 61850 97
Figura 36 – Modelo de Objetivos da operação baseado na IEC-61850 98
Figura 37 – Modelo de Objetos 100
Figura 38 – Diagrama de responsabilidades 101
Figura 39 – Modelo de Operações 102
Figura 40 – Rede de Petri (Operação IEC-61850) 103
Figura 41 – Arvore de alcançabilidade 104
Figura 42 – Analise de resultados do espaço de estados da RdP 105

LISTA DE TABELAS

Tabela 1 – Modelos de referência SG em diferentes regiões do mundo. 35
Tabela 2 – Temporal Logic Operators 62
Tabela 3 – Dados do Projeto 89
Tabela 4 – Possíveis novos objetivos para o projeto 90
Tabela 5 – Objetivos Emergentes 96
Tabela 6 – Requisitos da Operação de força baseado na IEC 61850. 99
Tabela 7 – Sentenças LTL do Objetivo de "Reduzir o consumo de energia" 121
Tabela 8 – Joutnal Sustainable Energy, Grids and Networks - Elsevier 123
Tabela 9 – Cap-15 Livro: Eng. Elétrica 124
Tabela 10 – INDUSCOM-2021 124
Tabela 11 – CBA-2020 125
Tabela 12 – SBSE-2020 125
Tabela 13 – Journal Brazilian Applied Science Review 126
Tabela 14 – SBAI-2019 126
Tabela 15 – CBA-2018 127
Tabela 16 – SBSE-2018 127

LISTA DE ABREVIATURAS E SIGLAS

ANSI ..... American National Standards Institute
CLP's .... controladores de lógica programável
CTL ..... Computation Tree Logics
EPRI ..... *Electric Power Research Institute*
ER ..... Engenharia de Requisitos
ESOs ..... European Standardization Organizations
ETSI ..... European Telecommunications Standards Institute
GBRAM . *Goal Based Requirements Analysis Method*
GD ..... Geração Distribuída
GOOSE .. *Generic Object Oriented Substation Event*
GORE .... *Goal Oriented Requirements Engineering*
IEA ..... *International Energy Agency*
IEDs ..... Dispositivos Eletrônicos Inteligentes
IEEE ..... Instituto de Engenheiros Eletricistas e Eletrônicos
IHM ..... Interface Homem-Máquina
INCOSE .. International Council of Systems Engineering
ISO ..... *International Organization for Standardization*
KAOS .... *Knowledge Acquisition in Automated Specification*
LNs ..... Nodos Lógicos
LTL ..... Logica temporal linear
MBSE .... *Model-Based Systems Engineering*
MDE ..... *Model Driven Engineering*
MMS ..... Mensagem de Manufatura
NIST ..... National Institute of Standards and Technology
OCL ..... Object Constraint Language
OMG ..... *Object Management Group*
RdP ..... Rede de Petri
RED ..... Recursos Energéticos Distribuídos
RF ..... Requisitos Funcionais
RNF ..... Requisitos Não-Funcionais
RTU ..... Unidade Terminal Remota

RUP ..... *Rational Unified Process*
SAE ..... Sistemas de Armazenamento de Energia
SCADA .. *Supervisory Control and Data Acquisition*
SCL ..... Substation Configuration Language
SDOs .... Organizações de Desenvolvimento de Padrões
SE ..... Engenharia de Sistemas
SEBoK ... *Systems Engineering Body of Knowledge*
SEMD ... *Service Engineering Modeling and Design*
SEP ..... Sistema Elétrico de potência
SG ..... Smart Grid
SGAM ... Smart Grid Architecture Model
SGFC .... Grupo com Foco em Smart Grid
UIT-T .... União Internacional de Telecomunicações
UML ..... *Unified Modeling Language*
XML ..... Extensible Markup Language

SUMÁRIO

1 INTRODUÇÃO . . . 23
1.1 Introdução . . . 23
1.2 O problema . . . 25
1.3 Objetivo . . . 27
1.3.1 Objetivos específicos . . . 27
1.4 Contribuições . . . 27
1.5 Metodologia . . . 28
1.6 Organização do Trabalho . . . 29

2 PESQUISA BIBLIOGRÁFICA . . . 31
2.1 Perspectivas de Design em Sistemas de Energia . . . 31
2.2 Modelos de Referência para sistemas Smart Grid . . . 34
2.3 Síntese do Capítulo . . . 40

3 DESCRIÇÃO DA PROPOSTA PARA O DESIGN DE SISTEMAS DE ENERGIA . . . 43
3.1 Conceitos de Engenharia de Sistemas aplicado a Sistemas de Energia 43
3.1.1 Sistemas-de-Sistemas (SoS) . . . 46
3.1.1.1 Características dos SoS . . . 46
3.1.1.2 Taxonomia dos SoS . . . 46
3.1.2 Abordagem orientada a Serviços . . . 48
3.2 Visão Geral dos Sistemas Smart Grid . . . 48
3.2.1 Desafios e Oportunidades das Smart Grid . . . 49
3.3 A IEC 61850 no contexto das microgrids . . . 52
3.4 Engenharia de Requisitos no Design de Sistemas . . . 55
3.4.0.1 Métodos para obtenção de Requisitos . . . 56
3.4.0.1.1 Unified Modeling Language - UML . . . 56
3.4.0.1.2 Engenharia de Requisitos Orientada a Objetivos . . . 57
3.4.0.1.3 Método KAOS . . . 59
3.4.0.1.4 Lógica Linear Temporal-LTL . . . 61
3.4.0.1.5 Exemplo de aplicação . . . 63
3.5 As Redes de Petri no Design de Sistemas . . . 67
3.5.1 Definição formal . . . 69

3.5.2 Propriedades das Redes de Petri . . . 70
3.5.2.1 Propriedades funcionais . . . 70
3.5.2.2 Propiedades estruturais . . . 71
3.6 Síntese do Capítulo . . . 72

4 MÉTODO PROPOSTO . . . 75
4.1 Abordagem proposta para o Design de sistemas SG . . . 75
4.1.1 Smart Grid como um Serviço de Serviços (SoS+) . . . 77
4.2 Ambiente de Design proposto . . . 78
4.3 Ciclo de requisitos para design de sistemas SG . . . 80
4.3.1 Etapa 1: Coleta de dados do Domínio . . . 82
4.3.2 Etapa 2: Especificação de Requisitos do Sistem-as-is) . . . 82
4.3.3 Etapa 3: Especificação de Requisitos Sistem-to-be . . . 82
4.3.4 Etapa 5: Formalização dos Requisitos . . . 83
4.3.5 Etapa 6: Verificação formal dos Requisitos . . . 85
4.4 Síntese do Capítulo . . . 86

5 ESTUDO DE CASO . . . 87
5.1 Modelando o sistem-as-is . . . 89
5.2 Modelando o sistem-to-be . . . 92
5.2.1 Modelo de Objetivos . . . 93
5.2.1.1 Modelando o sistema de Informação . . . 93
5.2.1.2 Modelando o sistema de Força . . . 96
5.2.2 Modelo de Objetos . . . 98
5.2.3 Modelo de Responsabilidades . . . 99
5.2.4 Diagrama de Operações . . . 101
5.3 Formalização dos Requisitos . . . 101
5.3.1 Verificação formal dos Requisitos . . . 104
5.4 Síntese do Capítulo . . . 105

6 CONCLUSÕES E TRABALHO FUTURO . . . 107
6.1 Conclusões . . . 107
6.2 Trabalhos futuros . . . 109

REFERÊNCIAS . . . 111

Glossário . . . 119

APÊNDICE A FORMALIZAÇÃO DOS REQUISITOS EM LTL . . . 121

APÊNDICE B PUBLICAÇÕES DECORRENTES DA PESQUISA . . . 123

CAPÍTULO

1

INTRODUÇÃO

1.1 Introdução

A energia elétrica é um item fundamental para melhorar a qualidade de vida da população. No entanto, segundo o Instituto Brasileiro de Geografia e Estatística, (IBGE, 2018) em 2010 cerca de 729 mil famílias brasileiras não tinham acesso a nenhum recurso energético, principalmente por residirem em locais muito afastados dos centros de distribuição.

No cenário amazônico, a situação não é diferente, apesar de ser uma região de grande potencial energético, características geográficas, questões ambientais e técnicas inviabilizam a implantação de redes de distribuição convencionais. Este fato é um agravante no quadro atual, tornando o equacionamento do problema de fornecimento de energia um grande desafio.

Para atender esse novo cenário surgem os sistemas Smart Grid (SG) com geração de energia renovável, e tem se mostrado uma alternativa viável para diversificar e racionalizar as necessidades de suprimento de energia, especialmente se baseada na integração de diferentes fontes alternativas de geração (distribuída e sustentáveis). A integração de diferentes fontes de geração de energia para suprir a demanda constitui os denominados sistemas híbridos de geração de energia (FALCÃO, 2010).

Por outro lado, verifica-se que os sistemas SG possuem uma complexidade inerente por serem sistemas eminentemente heterogêneos, abertos e distribuídos, e possuírem ambientes para desenvolvimento (*design*) e aplicação disjuntos, ou seja, com domínios de conhecimento com pouca ou nenhuma interação, pelo que o processo de design se torna bastante complexo (OLIVEIRA; SILVA, 2015).

Em contrapartida os tradicionais sistemas de energia são centralizados por meio da concessionaria de serviços de públicos de energia responsável pela geração, transmissão e distribuição de energia (ELÉTRICA, 2018). O que significa que mudanças e inovações no projeto elétricos devem ser compatíveis com sistema centralizado já existente (ferramentas,

fornecedores etc.), ou seja, um sistema com pouca flexibilidade no processo de design.

Portanto, o modelo tradicional centralizado bem sendo modificado, por causa do surgimento dos sistemas SG e microgrid, que possuem um tipo de gestão de consumo não centralizada, onde o usuário consumidor pode também ser produtor (*prosumer*). Neste novo cenário os novos sistemas SG tem que se adaptar ao já existente que pode ser centralizado (Concessionaria), ou sistemas distribuídos (ELÉTRICA, 2018); (RODRIGUEZ *et al.*, 2007). Por conseguinte, o domínio de design deve ser mais flexível, ou seja, não apenas considerando os aspectos técnicos (equipamentos, entre outros), se não também, como acoplar os subsistemas com um sistema de energia maior. O projeto de sistemas tradicionais, preocupados apenas com aspectos técnicos tendem a ser mau sucedidos.

Nesse contexto, o ciclo de vida de sistemas SG tem uma fase inicial importante, caracterizada por um conhecimento ainda incompleto sobre o sistema - fase onde estrategicamente é mais fácil tratar as dificuldades e mais fácil introduzir mudanças. Estamos falando da fase de especificação de requisitos, fase que influenciará muito no sucesso das fases posteriores, e é condição necessária para determinar o bom andamento do projeto (KENDAL; CREEN, 2007).

Cabe ressaltar que o termo de *"Design"* utilizado neste trabalho e substancialmente diferente da sua simples tradução para "Projeto" o que de fato seria comum, nesse contexto cabe destacar que um determinado Projeto está formado uma serie de fases (ciclo de vida): Requisitos, *design*, implementação, manutenção e descomissionamento, isso significa, que o *Design* e uma etapa das etapas do Projeto, previa à implementação, portanto, trata-se de definições diferentes (INCOSE, 2015); (SILVA; NOF, 2015).

Por outro lado, verificamos que todo sistema artificial (feito pelo homem) tem um ciclo de vida, ainda que não esteja formalmente definido (INCOSE, 2015). Na nomenclatura moderna chama-se design a fase que leva em consideração o processo como um todo, desde a especificação de requisitos, passando pelo desenvolvimento, implantação, manutenção, comissionamento do projeto, até a fase final de aposentadoria, ou seja, o descarte do sistema (SILVA; NOF, 2015).

Uma pesquisa recente de (BUEDE; MILLER, 2016) mostra a importância das etapas iniciais do processo de design, onde é feito um estudo comparativo mostrando que o comprometimento atinge aproximadamente 80 % no final do design detalhado para uma taxa de ocorrência de 20 %, enquanto na final da fase de design preliminar (final da fase de requisitos) o comprometimento seria de 55 % para uma taxa de ocorrência de 3 %. Isso significa que os erros encontrados nessas últimas etapas do ciclo terão impactos substancialmente negativos sobre o custo global do sistema, além de inviabilizar respostas às necessidades dos usuários.

O *Design* de projetos de sistemas SG não escapa dessa problemática. A etapa inicial que envolve a especificação de requisitos também precisa ser melhorada, especialmente, devido à peculiar complexidade do sistema. Por isso é necessário introduzir métodos, ferramentas e conceitos que ajudem a caracterizar o ambiente de desenvolvimento do sistema, e convergir

design confiável.

Na verdade, a introdução de uma fase de requisitos estruturada e que antecipe a formalização no ciclo de vida dos sistemas SG é uma tendência recente em sistemas elétricos e tem chamado muito a atenção de pesquisadores e implementadores, conforme mostra a literatura (ROHJANS; DÄNEKAS; USLAR, 2012) (ROBOAM, 2012), (GERBER, 2014). Alguns métodos existentes até consideram uma fase de requisitos no ciclo de vida deste tipo de sistema, mas não cumprem plenamente a fase de requisitos (composta de eliciação, análise, validação, modelagem formal, verificação), apenas satisfazem algumas etapas preliminares de modo informal (POSTIGO; SILVA; SILVA, 2020) (POSTIGO; SILVA, 2018).

Recentemente foi proposto um método que introduz a formalização para resolver problemas de Design, na qual, primeiramente representa-se os requisitos através de um modelo que é analisado para posteriormente projetar uma solução. Esta solução pode ser testada e verificada para posteriormente ser apresentada como solução final caso o resultado for satisfatório (SILVA *et al.*, 2012).

Embora se existam métodos para *Design* de sistemas de energia, verifica-se que geralmente a etapa inicial não recebe a atenção necessária. A especificação e análise de requisitos carecem de métodos específicos de tratamento e, consequentemente, resultam em dificuldades e custos no restante do ciclo de vida (MAZZOLINI; BRUSAFERRI; CARPANZANO, 2011).

Além disso, trabalhar com requisitos informais, embora facilite o processo de design fora da academia, pode levar a falhas não detectadas durante a especificação de requisitos (onde os custos dos erros são menores), onerando o projeto como um todo (SILVA, 2016b). A complexidade inserida por lidar com um arranjo distribuído de sub-sistemas (nem todos feitos ao mesmo tempo), é outro fator importante para justificar o uso de métodos formalmente fechados e consistentes que podem ser analisados e verificados antes da implementação.

Neste trabalho apresentamos uma proposta de método de análise formal de requisitos que possa ser aplicado a um domínio flexível de sistemas elétricos especialmente SG. A proposta se baseia em uma abordagem sistêmica e de sistemas de sistemas e antecipa a formalização através de uma linguagem esquemática, dentro do escopo do *Model Driven Engineering* (MDE) (CRETU; DUMITRIU, 2014).

1.2 O problema

Após revisão da literatura, observa-se que não é comum ter uma fase de requisitos mais extensa e formal no desenvolvimento de sistemas elétricos ou em sistemas Smart Grid (GERBER, 2014); (ROHJANS; DÄNEKAS; USLAR, 2012). A fase de especificação de requisitos é particularmente importante, como vimos na seção anterior, pelo fato de os projetos modernos de geração distribuída e sustentável serem eminentemente sistêmicos e baseados na composição de

subsistemas, que além de tudo devem se integrar ao sistema legado de geração e distribuição de energia elétrica. Nesse contexto, verifica-se que os métodos existentes não abrangem a plenitude a etapa de requisitos para o design de sistemas SG, restringindo-se às etapas preliminares, de modo informal, ou complemente funcionais, como na proposta recente da arquitetura SGAM e Intelligrid (COMMISSION *et al.*, 2008); (USLAR *et al.*, 2012)

Desta forma, e considerando a importância da fase de requisitos, entende-se que é necessário fornecer métodos, ferramentas e conceitos para melhor caracterizar, compreender e especificar o domínio de aplicação de sistemas SG (RODRIGUEZ *et al.*, 2007).

Portanto, o escopo desta tese é resolver o seguinte problema: Suprir ausência de um método formal, que não seja estritamente funcional, para especificação de requisitos aplicado a um domínio flexível de modernos sistemas elétricos (SG), sistemas eminentemente automatizados.

Assim, o principal objetivo deste trabalho é propor um ciclo de requisitos para redes de SG que inclua a formalização já nos passos iniciais e que possua:

- uma etapa de eliciação modelada com uma linguagem semi-formal como *Unified Modeling Language* (UML) (funcional estruturado, orientado a objetos) ou KAOS (orientado a objetos e a objetivos);
- um ciclo de requisitos baseado em uma sistêmica, e distribuída, ou baseada em Sistemas de Sistemas;
- um ciclo orientado a serviços, ou a Serviços de Serviços (SoS+);
- um processo de formalização dos requisitos, através da sintetização de um modelo formal em Redes de Petri, baseado em trabalhos já desenvolvidos no D-Lab (Laboratório de Design de Sistemas da Escola Politécnica da USP) (SILVA, 2016a);
- estrutura para a inclusão da reutilização de projetos pelos requisitos.

Portanto, existem contribuição em todas as etapas, utilizando métodos com reconhecida base teórica (como a análise de *workflow* e de propriedades nas Redes de Petri), uma parte metodológica inovadora, incluindo *Goal Oriented Requirements Engineering* (GORE), a inserção em métodos da Teoria de Sistemas ou *Model-Based Systems Engineering* (MBSE), e de *Service Engineering Modeling and Design* (SEMD). Baseados nestes métodos e teoria de suporte a proposta deste trabalho é compor um ciclo fechado de requisitos, baseado na análise de domínios do MBSE e na Ciência de Serviços. O trabalho foca na análise formal dos requisitos que pode ser iterada refinando o modelo a cada ciclo.

1.3 Objetivo

O Objetivo principal é propor um método método de Design sistêmico para especificação e análise formal de requisitos aplicado a domínios flexíveis mais especificamente a sistemas SG's colaborativas e conectadas a provedores comunitários locais ou globais, baseado em uma abordagem de serviço.

Assim, o sistema deve fazer a ponte entre a etapa mais abstrata, onde o conhecimento sobre o sistema que se quer desenvolver é ainda difuso e etapas onde se pode precisar o seu comportamento através de modelos, dentro do escopo do MDE (CRETU; DUMITRIU, 2014). Para ter um vínculo maior com a aplicação prática, o método proposto deve também admitir o uso de arquiteturas de referência consolidadas em sistemas elétricos, e especialmente em SGs.

A hipótese base da proposta é que, a partir de uma arquitetura de referência consolidada em sistemas SG seja possível reduzir o esforço na representação dos requisitos usando uma abordagem orientada a objetivos. Nesse caso, diagramas visuais baseados no método *Knowledge Acquisition in Automated Specification* (KAOS) que serão transformados em uma representação de Redes de Petri ISO/IEC 15.909 (para modelagem e análise), onde objetivos restrições e propriedades podem ser extraídos de normas como a IEC 61850.

1.3.1 Objetivos específicos

- Estudar e avaliar arquiteturas de referência compatíveis a sistemas SG e microgrid, e satisfaçam as orientações da norma IEC 2030 adaptadas a sistemas automatizados;
- Estudar métodos modernos de engenharia de requisitos, especialmente os orientados a objetivos GORE;
- Modelar a automação e *workflow* de redes de SG's usando Redes de Petri, definidas dentro da norma ISO/IEC 15.909;
- Validar o modelo com técnicas de verificação formal, e análise das propriedades das Redes de Petri;
- Mostrar a aplicabilidade do método proposto em um estudo de caso, especificamente uma microgrid para região Amazônica, considerando as características de sustentabilidade locais e a necessidade de um acoplamento com o usuário.

1.4 Contribuições

A principal contribuição acadêmica e tecnológica deste trabalho é dar uma resposta metodológica para a crescente complexidade e flexibilidade do projeto de sistemas SG, preservando sua natureza incremental e composicional. Esta contribuição se materializa na proposição de

um ciclo fechado de análise formal de requisitos aplicável a um domínio flexível de sistemas energia baseados em fontes distribuídas, sustentáveis e com elevado grau de automação. Este ciclo é amparado por uma visão sistêmica, no contexto de sistemas de sistemas. O ciclo de requisitos proposto está ligado ao processo de eliciação e portanto usa uma primeira modelagem semi-formal, onde se introduz (o que é uma novidade no design de sistemas elétricos) métodos baseados em objetivos ao invés de métodos funcionais. A formalização inserida no processo de análise e verificação de requisitos, usa as redes de Petri inseridas no padrão ISO/IEC 15.909.

Finalmente, este trabalho vai explorar o *design* de sistemas elétricos como serviços e portanto introduzir de forma prática o *Service of Services* (SoS+). Especial atenção deve ser dada à integração do SoS com o usuário final, que é a base dos serviços. Tanto em sistemas urbanos como em comunidades isoladas (com sistema comunitário local) e participação do usuário é fundamental para que se atinja metas de eficiência e sustentabilidade das redes de SGs.

Fornecer uma resposta metodológica sistêmica para a crescente complexidade e flexibilidade do projeto SG, preservando sua natureza incremental e composicional;

Proposição de um ciclo fechado de engenharia de requisitos , baseados em uma abordagem formal de SoS, aplicável a um domínio flexível de sistemas SG;

Introduz um ciclo semiformal de requisitos, baseado em objetivos ao invés de métodos funcionais (o que é uma novidade no projeto de sistemas elétricos);

Explorar o design de sistemas elétricos como serviços e introduzindo de forma prática o *Service of Services* (SoS+).

Propor um método que reúne de forma harmônica o GORE, *Model Based System Engineering* (MBSE), e *Service Oriented Engineering Design* (SOED) em um mesmo ciclo, que se fecha com a formalização utilizando a Teoria de Sistemas Discretos e as Redes de Petri.

Espera-se, ainda que o ciclo de desenvolvimento de requisitos proposto proposta possa ser utilizada para outros processos de design de sistemas elétricos em geral.

1.5 Metodologia

O método de pesquisa utilizado neste trabalho é do tipo exploratório e dedutivo, aplicando uma visão sistêmica a partir da representação de modelos constituídos de uma estrutura observando o seu desempenho (SEVERINO, 2017). O primeiro passo empreendido, foi realizar uma revisão bibliográfica abrangente sobre o cenário atual no que diz respeito aos métodos e ferramentas de design em sistemas energéticos disponível em bancos de dados de produção científica. Também foram pesquisadas normas internacionais e livros acadêmicos relacionados ao tema, além de anais de congressos de temas correlatos. Seguiu-se o estudos das normas existentes e a familiarização com o design de sistemas existentes e com dificuldades para aplicar os mesmos métodos aos sistemas modernos.A busca e o estudo de novos métodos foi o passo

seguinte, agora com um problema adicional que seria propor um modelo de desenvolvimento que pudesse integrar as vantagens de vários deles de forma harmônica.

Houve uma preocupação constante com o lado prático, o que qualifica o trabalho como pesquisa aplicada, ou ainda teórico-experimental, baseada no levantamento teórico de métodos e soluções, e na proposição de procedimentos e de modelos matemáticos para melhorar o desempenho destes métodos. Acrescente-se a isso a aplicação da proposta a um caso real, para a avaliação dos resultados e da viabilidade do método proposto (SEVERINO, 2017).

As etapas do método aplicado neste trabalho são:

1- Estudo dos aspectos teóricos relacionados ao Design de sistemas de energia, conceitos da engenharia de requisitos, método GORE, modelagem em Redes de Petri, métodos de verificação formal, Sistemas Smart Grid (SG), Arquiteturas de Referência para SG, Arquitetura IEC 61850. Escolha das técnicas para a fase seguinte.

2- Estudo específico técnicas e ferramentas escolhidas: Métodos GORE e sua versão de uso prático, o KAOS, implementado no sistema Objectiver 3.0; estudo da UML 2.5 e comparação dos métodos funcionais na sua capacidade máxima em comparação aos métodos orientados a objetivo; técnicas de modelagem e verificação formal baseadas em Redes de Petri implementadas na ferramenta clássica PIPE 4.3 e no sistema GHENeSys.

3 -Integração das técnicas acima para compor um método fechado. Avaliação de performance em problemas de resultado conhecido.

4 - Aplicação em estudo de caso real de um microgid a ser desenvolvida na região Amazônica.

1.6 Organização do Trabalho

Além desta introdução, o trabalho se encontra estruturado na seguinte sequência:

- Capítulo 2: Descreve a fundamentação teórica sobre o trabalho, as discussões correntes, e as contribuições mais relevantes relacionadas aos métodos de *Design* praticados em diferentes áreas, em especial na área de sistemas de energia, evidenciando necessidades e oportunidades de pesquisa;

- Capítulo 3: Baseado no capítulo anterior o este capítulo apresenta uma visão específica das técnicas e métodos (inclusive formais) que serão utilizados para dar suporte à proposta deste trabalho;

- Capítulo 4: Apresenta o método proposto, baseado na integração SoS/SoS+ bem como as vantagens e principais avanços que visam as dificuldades para compor um ciclo de desenvolvimento de requisitos para redes de SG's identificadas nos capítulos anteriores;

- Capítulo 5: Para mostrar a capacidade de aplicação do método proposto, este é aplicado um estudo de caso real com a evolução de um microgrid implementado em uma região isolada da floresta amazônica como um sistema funcional dependente da rede comunitária (local). Esta evolução vislumbra a possibilidade de usuários diretos ser tornarem produtores/consumidores em uma rede de SG's ou microgrids SoS/SoS+.
- Capítulo 6: apresenta as principais conclusões, contribuições e trabalhos futuros.

Finalmente são apresentadas as referências bibliográficas consultadas e os anexos.

CAPÍTULO

2

PESQUISA BIBLIOGRÁFICA

Nesta secção, iremos a apresentar as discussões correntes e as contribuições mais relevantes relacionada aos métodos de *Design* praticados em diferentes áreas, em especial na área de sistemas de energia. Apresenta-se um breve resumo das abordagens, seguidos pelos métodos, ferramentas mais destacadas, segundo as publicações relevantes na área.

2.1 Perspectivas de Design em Sistemas de Energia

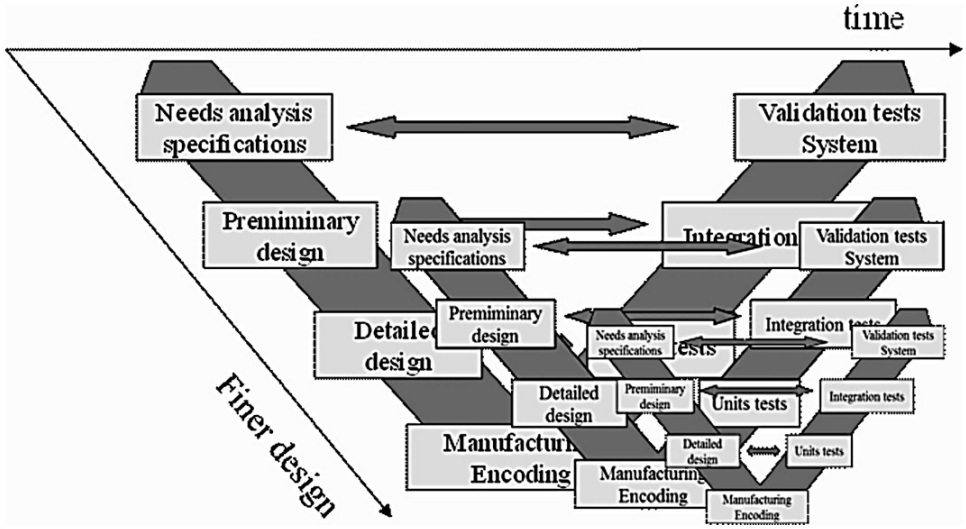
A introdução de redes inteligentes (SG) está levando pesquisadores e desenvolvedores a sistemas de energia complexos, fortemente acoplados a elementos pertencentes a diferentes áreas do conhecimento (física, engenharia, entre outros) (ELÉTRICA, 2018); (NYSO, 2017). A pesquisa sobre métodos de design (modelagem, análise, simulação, verificação) está crescendo e explorando perspectivas distintas. A seguir, destaca-se algumas dessas perspectivas.

Uma primeira perspectiva consiste em tratar os sistemas de energia como sistemas ao invés de produtos ou um arranjo de dispositivos (ROBOAM, 2012). Normalmente, essa nova tendência se adapta aos métodos clássicos de projeto de sistemas, como o modelo “V” (ROBOAM, 2012), amplamente utilizado pela indústria de manufatura, mesclando a abordagem “top-down” e linear - baseada puramente em funcionalidade - como mostrado na Figura 1.

O método "V"clássico é aprimorado pela divisão do ciclo do projeto primário em vários ciclos "mini-V", com refinamento sucessivo, chamado de Design interativo tipo "V". Este método tem duas etapas principais: análise e decomposição da demanda, e especificação, separando os aspectos "funcionais"dos "estruturais". O foco do método é o desempenho técnico e as avaliações de custo ao longo do ciclo de vida.

Em 2012 Roboam (ROBOAM, 2012) já se preocupava com a abordagem sistêmica, na qual utilizou uma abordagem clássica tipo "V", funcional, que considera fases de requisitos teste e implementação, conectando as fases equivalentes. Um problema com o método "V"utilizado

Figura 1 – Ciclo de Design-V



Fonte: baseada em (ROBOAM, 2012)

neste trabalho é que não foi feito para sistemas distribuídos, e a alternativa encontrada pelo autor foi fazer uma interação de ciclos V, o que lhe permitiu aumentar flexibilidade no design. No que diz respeito à fase inicial de requisitos, normalmente o método "V"é associado a uma linguagem semi-formal, a UML (muito adaptada para esse tipo de sistemas).

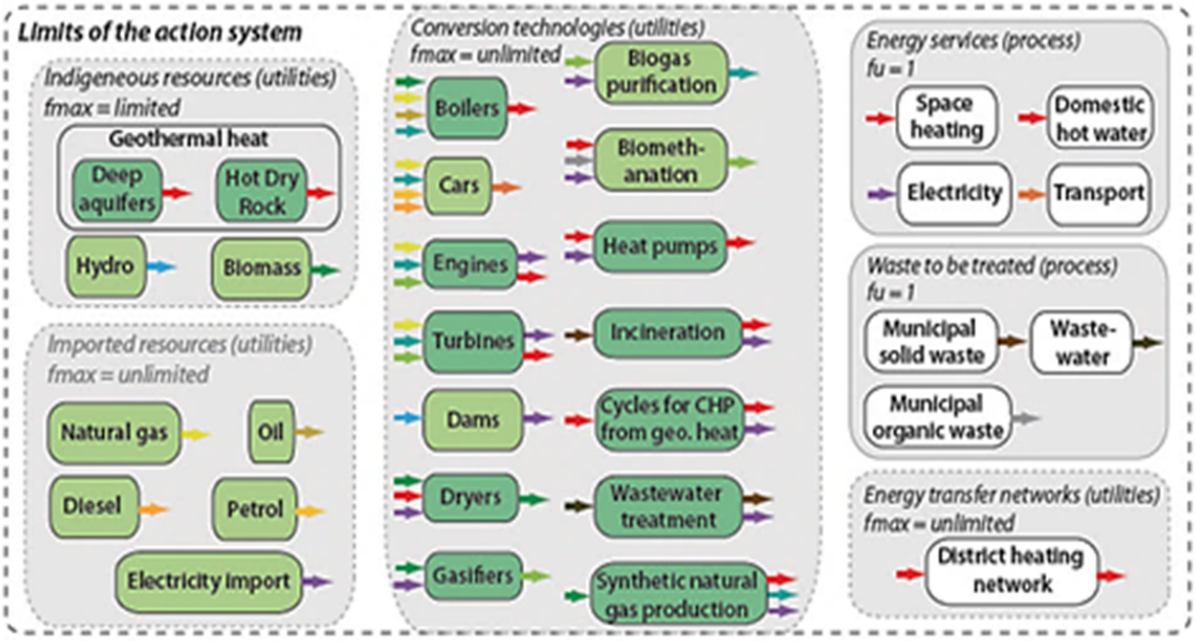
Entretanto, na maior parte dos casos práticos somente o diagrama de casos de uso é usado e não uma combinação de diagramas UML. Mesmo com um conjunto de diagramas, tem-se o problema de ter que balancear requisitos funcionais e não-funcionais logo no início do design.

Outra perspectiva é a utilização da Avaliação do Ciclo de Vida (LCA) utilizando técnicas de otimização Multi-objetivo para o *Design* de sistemas de energia, especialmente para sistemas baseados em energia renovável (GERBER, 2014). Trata-se de uma abordagem interdisciplinar, que integra conceitos das áreas de projeto, aspectos ambientais, e os relaciona com os aspectos técnicos e econômicos. Este método é aplicado à tomada de decisão relacionada ao planejamento territorial e a inclusão de sistemas de energia, objetivando a identificação de melhores formas de utilização dos recursos energéticos.

O método é baseado nas normas do *International Organization for Standardization* (ISO) mostrado na Figura 2, na qual o LCA, e estruturado em quatro etapas: a definição de objetivo e escopo, o inventário do ciclo de vida, a avaliação do impacto regional e a interpretação. O autor utiliza a modelagem do sistema para estruturar, organizar a informação, e representar as interações entre os diferentes componentes considerados no projeto.

Gerber se preocupa com a sustentabilidade, pensando no processo de projeto de sistemas

Figura 2 – Metodologia LCA



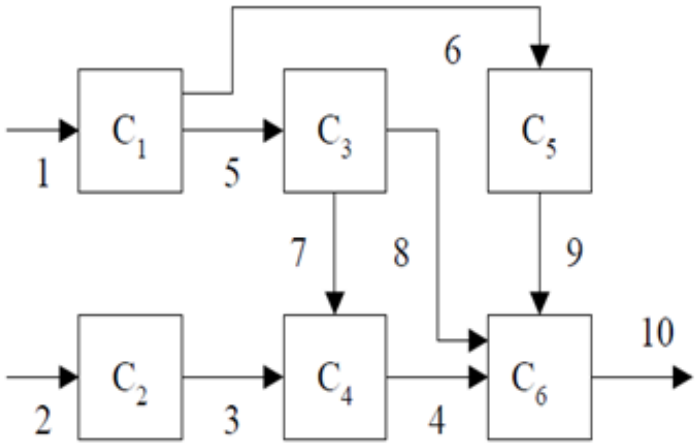
Fonte: baseada em (GERBER, 2014)

de energia renováveis, acoplados a um domínio maior, utilizando técnicas multi- objetivo. Um complicador desta perspectiva é que se trata de uma técnica muito precisa, entrando em detalhes de funcionamento, já nas etapas iniciais, o que não permite uma visão mais ampla e sistêmica, importante para o projeto e para a tomada de decisão - e para a sobrevivência do acoplamento regional. O que significa que a manutenção do sistema será muito mais complicada. Por outro lado, o método proposto se adapta bem a sistemas com provimento centralizado, vindo de concessionárias.

Uma outra perspectiva aborda o formalismo de especificação técnica de requisitos baseada na teoria de grafos proposta por *Bondy & Murty* para realizar a modelagem e verificação, conforme mostrado na Figura 3. O trabalho foi apresentado por Frangopoulos (FRANGOPOULOS; SPAKOVSKY; SCIUBBA, 2002) no artigo *Methods for the Design and Synthesis Optimization of Energy Systems*. Apresenta-se um método para resolver problemas energéticos com um grande número de graus de liberdade não-lineares e complexos, classificados em três níveis: síntese, design e operação. No trabalho o autor foca no nível da síntese do sistema de energia, os quais são classificados em três grupos, baseados em heurística, métodos para atingir metas, e configuração ideal do sistema.

Frangupoulos, mostra preocupação com a análise formal e verificação dos requisitos. Esta proposta mostra um Design preocupado com aspectos exclusivamente técnicos, sem pensar em um domínio mais amplo e flexível ou adaptação regional (FRANGOPOULOS; SPAKOVSKY; SCIUBBA, 2002). Portanto, embora reconheçamos que a preocupação com a modelagem formal e com a captura do *workflow* baseado em teoria de grafos seja uma boa alternativa, seria preciso recuperar a abrangência e adaptação ao domínio (regional).

Figura 3 – Formalismo baseado em grafos



Fonte: baseada em (FRANGOPOULOS; SPAKOVSKY; SCIUBBA, 2002)

Outro ponto importante, além da conjunção das tendências mencionadas, seria a adequação aos modelos de referência já existentes, baseados em longos estudos sobre o problema.

Portanto, para ter um vínculo maior com a aplicação prática, o método propõe o uso de arquiteturas de referência reconhecidas pela indústria de energia. As arquiteturas de referência fornecem parâmetros que ajudam a obter uma visão geral sobre o domínio, e incorporam as melhores práticas, e enfoques de soluções aceitas no setor, ajudado a reduzir o tempo para o desenvolvimento de projetos.

2.2 Modelos de Referência para sistemas Smart Grid

Uma enorme variedade de modelos e arquiteturas para o desenvolvimento de projetos SG têm surgido, endereçados à fase de design (EIJKELHOF *et al.*, 2020), e propondo padrões reconhecidos mundialmente pelas agências reguladoras de eletricidade em diferentes países.

As pesquisas relacionadas a modelos e arquiteturas de referência para desenvolvimento de projetos na área das SG têm ocorrido desde o início do século XXI e continuam até os dias de hoje (NAFI *et al.*, 2016). Entretanto, o desenvolvimento das referidas arquiteturas de referência é seguir as orientações do Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) para sistemas elétricos, sistemas embarcados, e comunicação digital. O intuito é desenvolver um modelo padrão que seja mundialmente reconhecido e utilizado pelas agências reguladoras de energia elétrica dos países, e permita a unificação de métodos e critérios.

O modelo IEEE 2030, é considerado o principal esforço de padronização para desenvolvimento de sistemas SG, (SATO *et al.*, 2015) (NAFI *et al.*, 2016). A Tabela 2 resume o esforço de padronização do SG realizado por várias organizações em diferentes regiões do mundo, (NAFI *et al.*, 2016).

Tabela 1 – Modelos de referência SG em diferentes regiões do mundo.

Location	SG standardization Organizations
United States of America	- National Institute of Standards and Technology (NIST) (NIST Framework and Roadmap for Smart Grid Interoperability Standards, 2010)
	- American National Standards Institute (ANSI) (Association, 2006)
European Countries	- European Standardization Mandate M441 - (SGAM)
	- Smart Meter Co-ordination Group ("Final Report of the CEN/CENELEC/ETSI Joint Working Group On Standards for Smart Grids,", 2011)
	- European Telecommunications Standards Institute (ETSI)
	- European Standardization Organizations (ESOs) (Standardization Mandate to European Standardization Organizations (ESOs) to Support European Smart Grid Deployment, 2011)
Japan	Japanese Industrial Standards Committee (JISC) (Standardization Mandate to European Standardization Organizations (ESOs) to Support European Smart Grid Deployment, 2011)
China	China State Grid Corporation of China (Uslar et al., 2010)
Australia	Standard Australia (Australian Standards For Smart Grids-Standards Roadmap, Standards Australia (SA), 2012)

Fonte: baseada em (NAFI *et al.*, 2016)

O Setor de Normalização da União Internacional de Telecomunicações (UIT-T) estabeleceu um grupo chamado Grupo com Foco em Smart Grid (SGFC) para desenvolver recomendações, avaliar o impacto das normas SG e fortalecer a relação entre as autoridades da UIT-T e empresas do setor elétrico.

Hoje em dia, várias Organizações de Desenvolvimento de Padrões (SDOs) e indústrias estão trabalhando para desenvolver tecnologias e padrões relacionados à SG, enquanto governos em todo o mundo estão gradualmente emitindo diretrizes e contribuindo para a modernização de suas redes de energia (SATO *et al.*, 2015).

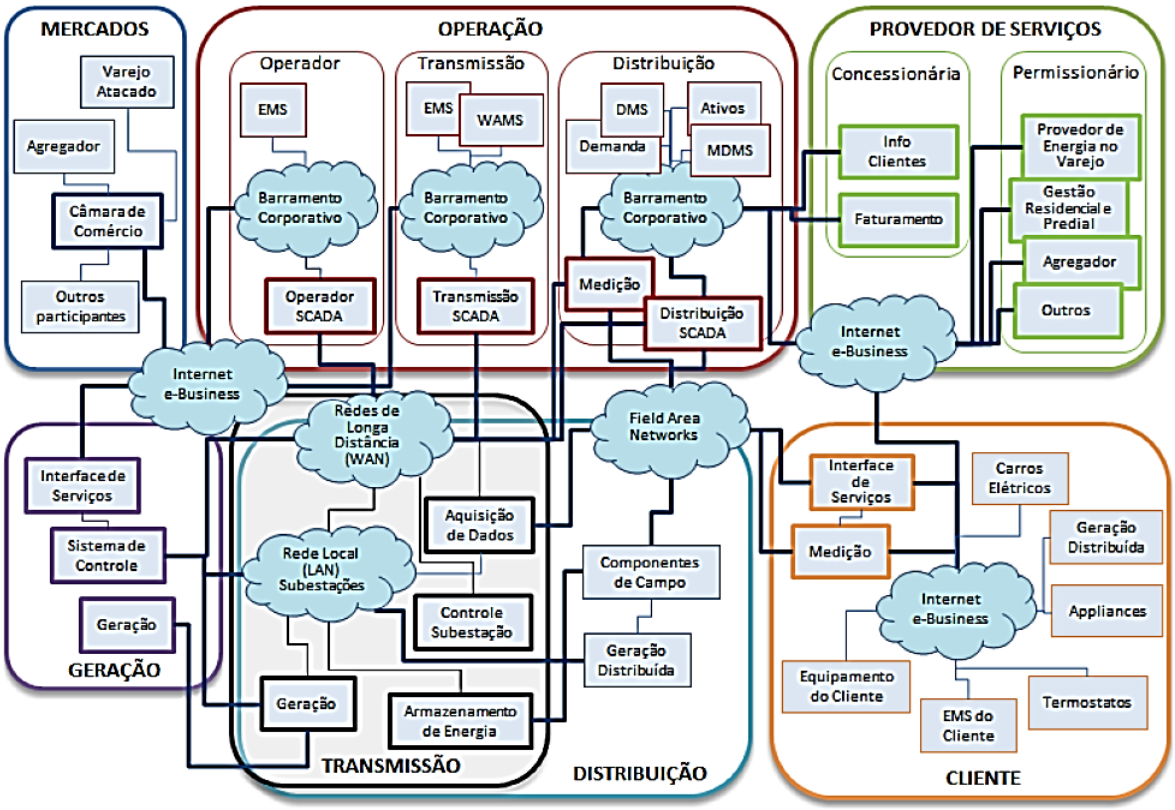
A IEC está entre as maiores e mais bem estabelecidas SDOs, junto com a ISO e a União Internacional de Telecomunicações (ITU). Trata-se de uma SDO internacional, não governamental, que prepara e publica padrões internacionais para sistemas elétricos e eletrônicos, geração de energia, transmissão, distribuição e tecnologias associadas, (SATO *et al.*, 2015) (RUTA *et al.*, 2011).

A Organização ISO envolveu um esforço no desenvolvimento de padrões SG eletrodomésticos (definido pela ISO / IEC 14543-3) (RUTA *et al.*, 2011) e sistemas inteligentes de design para controle de edifícios (definidos pela ISO 16484 -5) (ISO, 2007). Existem também esforços de padronização nacional e internacional por diferentes agências, incluindo o IEC, o NIST, o

ANSI, o CIGRE, o ISO e o ESO, especificando uma ampla gama de atributos para SGs (SATO *et al.*, 2015).

A IEC publicou o Smart Grid Standardization, que descreve os requisitos para SGs nos padrões existentes e define a comunicação, segurança e planejamento como os três requisitos básicos para o design de SGs. O NIST *Framework and Roadmap* para padrões de interoperabilidade da SG, é considerado um roteiro que pode ser usado por engenheiros especialmente para implementação de projetos SG. Conforme mostrado na Figura 4, o *Roadmap* identifica os domínios e os principais blocos funcionais do SG e fornece um meio para analisar casos de uso e identificar interfaces e padrões necessários para garantir a interoperabilidade, (ROHJANS; DÄNEKAS; USLAR, 2012).

Figura 4 – Smart Grid Roadmap



Fonte: baseada em (ARNOLD *et al.*, 2010)

Segundo Fitzpatrick (FITZPATRICK; WOLLMAN, 2010) para o NIST, são oito os itens identificadas como de alta prioridade para padronização:

- Resposta à demanda e eficiência energética do consumidor;
- Consciência Situacional de Área Ampla;
- Armazenamento de energia;

- Transporte elétrico
- Infraestrutura de medição avançada;
- Gestão da rede de distribuição;
- Segurança cibernética;
- Comunicações de rede.

Sem dúvida o modelo Americano do NIST, junto às metodologias, ferramentas e recomendações do *Electric Power Research Institute* (EPRI) dos USA, é considerado umas das referenciais mais fortes no desenvolvimento e normatização de projetos para sistemas SG (FITZPATRICK; WOLLMAN, 2010) (COMMISSION *et al.*, 2008). Reiteramos que é um modelo muito importante, especialmente para prover o devido acoplamento dos SGs com os sistemas legados.

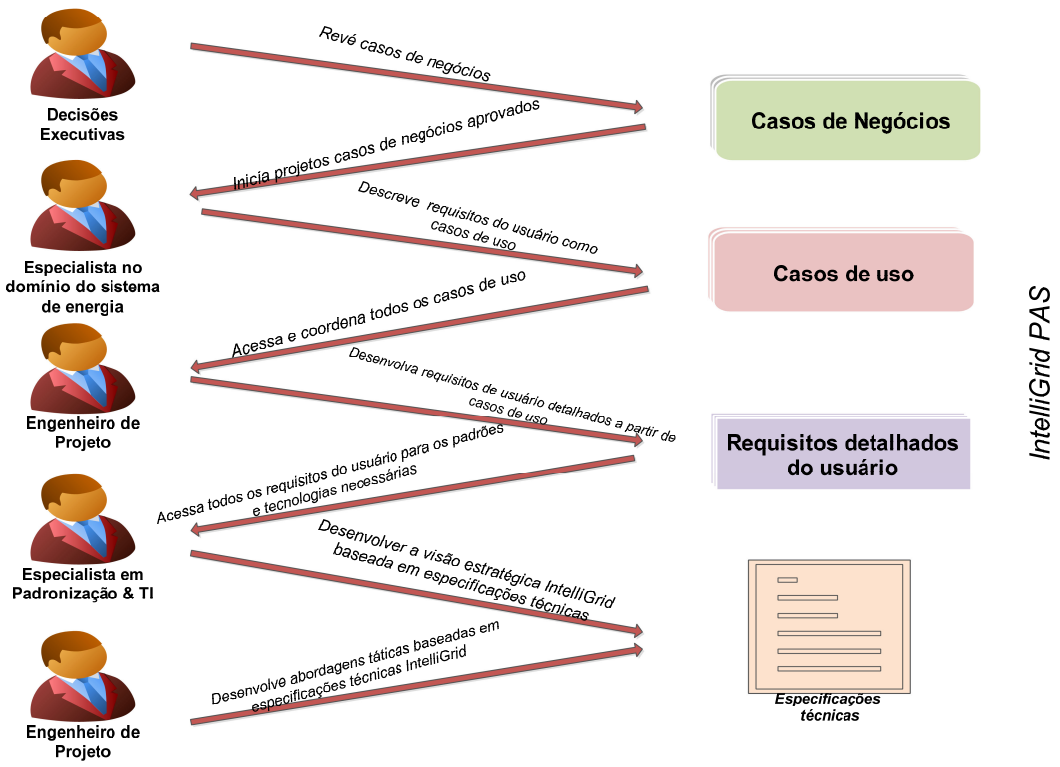
A arquitetura IntelliGrid patrocinado pela EPRI, fornece os princípios de design para troca de informações, integração de sistemas e uso de padrões para a indústria de sistemas de energia. Esta arquitetura é baseada em um conjunto de requisitos funcionais do sistema de energia para suportar sistemas de energia de *self-healing*, transações do mercado de energia, compartilhamento e sinergia entre funções operacionais de concessionárias e serviços integrados ao cliente. Identifica todos os agentes, humanos ou máquinas, envolvidos no projeto para estruturar uma comunicação adequada (ROHJANS; DÄNEKAS; USLAR, 2012). (USLAR *et al.*, 2012) (COMMISSION *et al.*, 2008).

Esta Arquitetura aborda os conceitos de "requisitos do usuário"de "especificações técnicas": os requisitos do usuário definem "o que"é necessário sem referência a quaisquer projetos ou tecnologias específicas, enquanto as especificações técnicas definem "como"implementar os sistemas de automação para atender os requisitos do usuário, (ROHJANS; DÄNEKAS; USLAR, 2012). Para a especificação dos requisitos utiliza a linguagem UML, mas é focada na utilização dos diagramas de casos de uso para a captura das funcionalidades da SG, Os Casos de Uso, representam operações relacionadas à transmissão, distribuição, e conexão com o sistema elétrico de potência, (USLAR *et al.*, 2012) (COMMISSION *et al.*, 2008); (GOTTSCHALK; USLAR; DELFS, 2017); (ANDRÉN; STRASSER; KASTNER, 2017).

A metodologia IntelliGrid baseado na IEC/PAS 62559 é ilustrada na Figura 5, o método, é dividido em cinco fases incrementais, relacionando as pessoas envolvidas e etapas de projeto (ROHJANS; DÄNEKAS; USLAR, 2012).

A Smart Grid Architecture Model (SGAM), ilustrado na figura 6 é um modelo de referência destinado à análise e visualização, de forma tridimensional de maneira tecnologicamente neutra. É baseada em arquiteturas já existentes misturando diferentes perspectivas e abordagens

Figura 5 – Arquitetura Intelligrid



Fonte: baseada em (ROHJANS; DÄNEKAS; USLAR, 2012)

na construção de um modelo comum, dando uma visão holística dos diferentes aspectos que integram as SG.

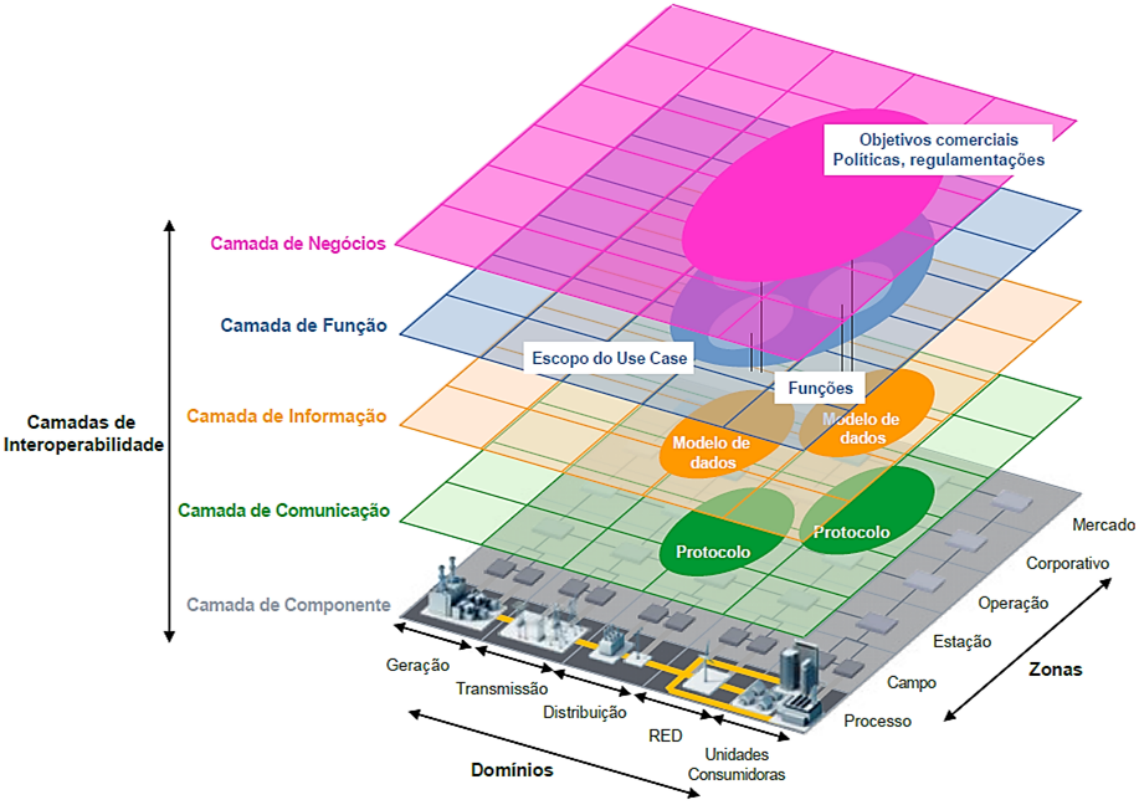
Através de uma abordagem sistêmica, possibilita soluções de projeto para SG por meio de representações visuais do estado atual e das evoluções esperadas em cenários futuros. A arquitetura SGAM é resultado do trabalho de grupos constituído pelos comitês europeus de normalização, CEN, CENELEC e ETSI, mandato M/490 da Comissão Europeia (USLAR *et al.*, 2012); (ROHJANS; DÄNEKAS; USLAR, 2012); (COMMISSION *et al.*, 2008).

A arquitetura SGAM, mostrada na Figura 6, é dividida em dois eixos: a gestão dos processos energéticos (relacionados aos domínios físicos do Sistema Elétrico de potência (SEP)), e o eixo da gestão da informação (níveis hierárquicos correspondentes ao gerenciamento do SEP). Ademais, existe um terceiro eixo, perpendicular aos dois primeiros, que modela cinco camadas de interoperabilidade entre componentes e sistemas.

O processo de mapeamento de *Use-Case* no SGAM é ilustrado na Figura 7, e serve como ferramenta para a escolha de normas, interfaces e protocolos, assim como a definição de perfis para cada camadas de interoperabilidade, permitindo o mapeamento de cada entidade no plano de domínios e zonas.

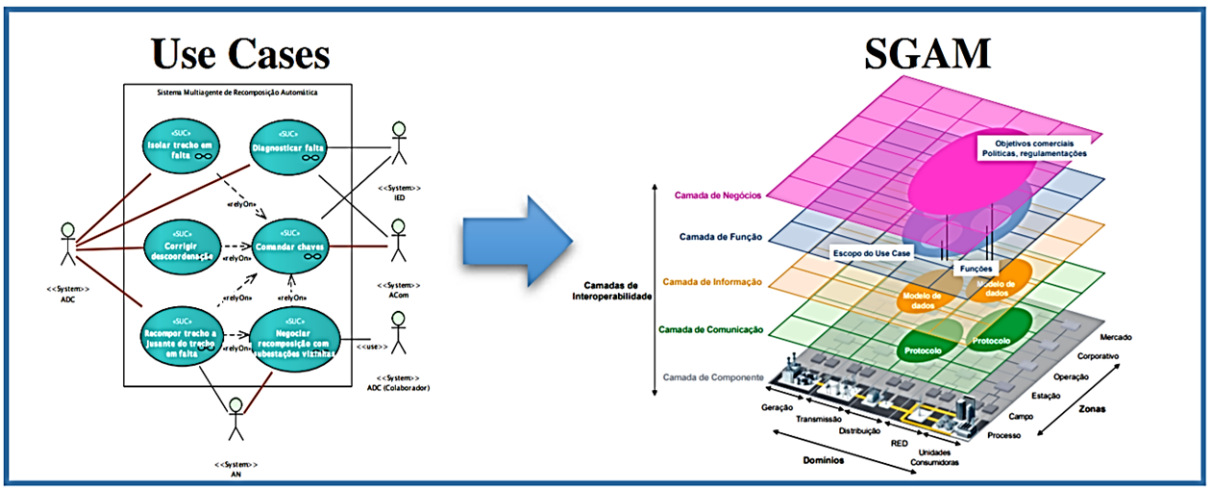
Esta arquitetura foi a responsável pela introdução da modelagem usando casos de uso em

Figura 6 – Arquitetura SGAM



Fonte:(USLAR; ROHJANS *et al.*, 2019)

Figura 7 – Mapeamento de Use-Case SGAM



Fonte: baseada em (USLAR *et al.*, 2012)

processos de normalização. Segundo(USLAR; ROHJANS *et al.*, 2019); (ROHJANS; DÄNEKAS; USLAR, 2012); (USLAR *et al.*, 2012), com a aplicação dos *Use-Cases* da IEC 62559 e do SGAM em projetos de SG a o método mostrou resultados positivos na a especificação requisitos de sistemas SG.

Entretanto, embora as arquiteturas SGAM e IntelliGrid sejam muito difundidas, devemos ressaltar que elas consistem em uma abordagem puramente funcional e que embora tenham mais cuidado com os requisitos, usam somente casos de uso na modelagem, concebida apenas para elicitação de requisitos, mas não para análise. Como se trata de abordagem semiformal, não pode dar suporte à verificação, (USLAR *et al.*, 2012); (ROHJANS; DÄNEKAS; USLAR, 2012).

Embora a abordagem de caso de uso proposta se destine a capturar requisitos de operação e funcionalidades do sistema SG, estas são usadas para compor um repositório de *rationales* de projeto que podem ser posteriormente reutilizados. A documentação dos modelos desenvolvidos é armazenada em um banco de dados UML (GOTTSCHALK; USLAR; DELFS, 2017); (ANDRÉN; STRASSER; KASTNER, 2017); (USLAR; ROHJANS *et al.*, 2019).

Assim, a capacidade de reutilização foi considerada através do uso de um repositório, baseado em IEC/PAS 62559, de histórico de modelos de casos de uso (GOTTSCHALK; USLAR; DELFS, 2017); (ROHJANS; DÄNEKAS; USLAR, 2012) (USLAR; ROHJANS *et al.*, 2019) (COMMISSION *et al.*, 2008).

No entanto, é importante observar que os diagramas visuais UML representam os requisitos eliciados predominantemente de modo "funcional", focado apenas em Casos de Uso, o que é questionado em outros trabalhos sobre design de sistemas (SILVA; SILVA, 2015). Por outro lado, a cardinalidade de um conjunto mínimo de diagramas necessários para representar um determinado sistema não é consensual. Portanto não existe uma alternativa clara para o uso da UML neste projetos.

Por outro lado, é importante salientar que a UML foi proposta por Ivar Jacobson para compor o processo de elicitação no *Rational Unified Process* (RUP) e posteriormente englobado pelo *Object Management Group* (OMG). Portanto, sem o restante do RUP faltam elementos para análise - hoje fornecidas pela SysML - e uma linguagem formal para a documentação, verificação e validação do design, o que não poderia ser feito sem uma linguagem formal (DOBING; PARSONS, 2008).

2.3 Síntese do Capítulo

A pesquisa mostrou que existe uma preocupação da comunidade acadêmica em prover uma fase de requisitos para o ciclo de vida dos sistemas de energia. (ROBOAM, 2012) (COMMISSION *et al.*, 2008). No entanto, verificou-se que a estrutura desta fase de requisitos ainda não está clara, especialmente no que diz respeito ao papel da verificação, com base em requisitos

formais. O projeto do sistema SG, assim como o projeto dos sistemas de energia elétrica em geral, segue as especificações e diretrizes do IEEE, (MARZAL *et al.*, 2017), que são úteis, mas não incluem a personalização do usuário, acoplamento do usuário e orientação para serviços. Assim, seguem um modelo mais orientado a "produto"do que a "serviço"que é o que realmente são.

O modelo IEEE 2030 é considerado um esforço de padronização recente para o desenvolvimento de sistemas SG (NAFI *et al.*, 2016). Nesse contexto, o modelo clássico “V” usado na área de sistemas recebeu a contribuição adicional de uma abordagem funcional semiformal baseada em UML (RANGANATHAN; NYGARD; MAGEL, 2011); (COMMISSION *et al.*, 2008), introduzindo a necessidade de inserir diagramas visuais desde a fase de eliciação e análise de requisitos.

Portanto, as tendências metodológicas atuais para o *design* do sistemas de energia podem resumidas, da seguinte forma:

- Os métodos são geralmente prescritivos, baseados em regulamentações metodológicas ou locais, mas o domínio (a região ou entorno onde o sistema será implementado) não faz parte do *Design*;
- A maioria dos métodos são funcionais, ou seja, priorizam a funcionalidade do sistema, e geralmente os requisitos se restringem a isso, correndo sério risco de consistência, pela falta (ou incompletude) dos requisitos não-funcionais;
- Muitos métodos são orientados a dispositivos, ou seja, não há uma visão do sistema e de sua integração com sistemas legados, condições regionais ou com os usuários;
- Muitos métodos consideram o SG como um produto, ou um arranjo de subprodutos que compõem o sistema elétrico;
- Algumas abordagens consideram sistemas centralizados mestre-escravo, onde SG’s são recursos individuais que respondem a um sistema mestre que centraliza todo o controle ou informação;
- A maioria das abordagens não são flexíveis, ou seja, a evolução dos sistemas não faz parte do design - como no *Design* de Sistemas - o que torna mais complicado fazer mudanças ou inovações (ou mesmo atender a requisitos emergentes);
- Apesar de todo o esforço na padronização e na confecção de repositórios de projetos para reutilização, a reutilização não faz parte dos métodos de *design*;

Por conseguinte, concluímos que uma abordagem moderna para o Design se sistemas de energia exige um ciclo de requisitos, que deve ser fechado (*closure*) e realimentado, e que

possibilite uma análise formal. Considerando todos esses aspectos, apresentaremos no próximo capítulo, a nossa proposta para o *design* de sistemas energia.

CAPÍTULO

3

DESCRIÇÃO DA PROPOSTA PARA O DESIGN DE SISTEMAS DE ENERGIA

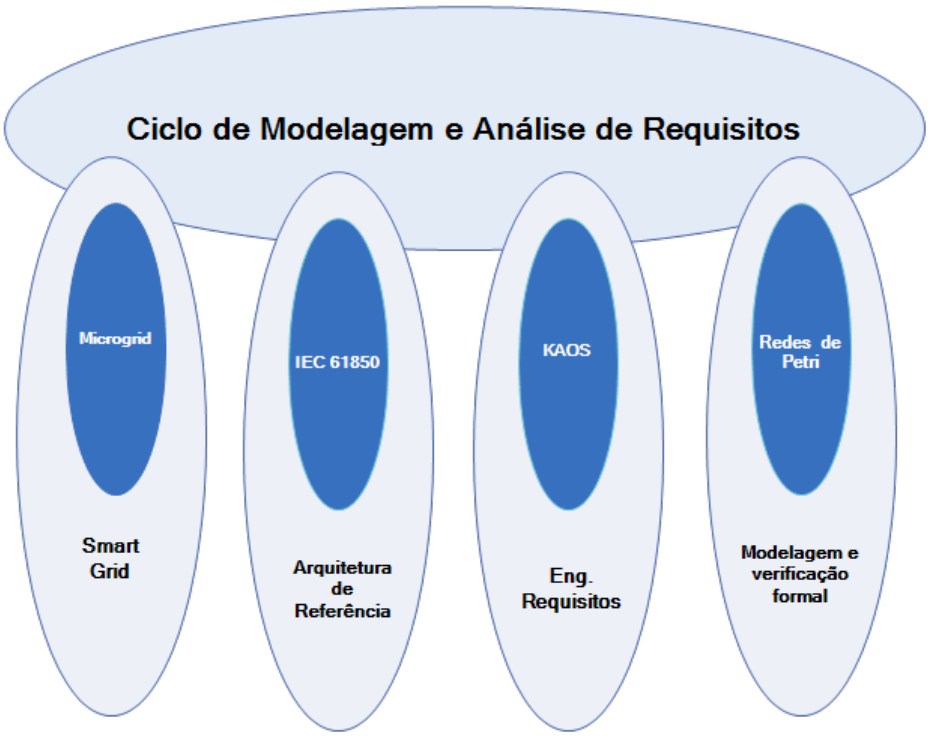
Neste capitulo iremos a abordar a descrição de uma proposta para o design de sistemas elétricos em domínios flexíveis, começando por elucidar os conceitos, métodos e ferramenta que embasam a proposta.

O intuito da proposta é atender a um ciclo de modelagem de requisitos que se inicia com uma descrição semi-formal dos requisitos capturados. A proposta enfatiza a fase de design que começa com modelagem e análise destes requisitos, atendendo a restrições regionais (do domínio), para sistemas distribuídos dos recursos energéticos e fontes sustentáveis, onde os elementos de composição podem ser microgrids acoplados a sistemas maiores, com um sistema de comunicação - fundamental, uma vez que estes subsistemas precisam estar interconectados. Além disso, a proposta incluirá alguma arquitetura de referência consagrada na área, para ter um maior grau de aplicabilidade. A figura 8, mostra o ciclo de modelagem e análise de requisitos proposto para design de sistemas de energia.

3.1 Conceitos de Engenharia de Sistemas aplicado a Sistemas de Energia

Na ótica da engenharia de sistemas, todo sistema tem um ciclo de vida que possui várias etapas que começa na definição de requisitos, passando pelo design, construção, integração, teste, e operação, até o descomissionamento do sistema(WALDEN *et al.*, 2015); (INCOSE, 2015). Nesse contexto, o Ciclo de Vida de Desenvolvimento de Sistemas (CVDS) é um processo que tem por objetivo produzir sistemas de alta qualidade que atendam às expectativas do cliente (*stakeholder*) e do usuário, que possa ser concluído dentro do tempo e custos estimados, e que ao mesmo tempo funcione de forma eficaz com menor custo de manutenção (INCOSE, 2015).

Figura 8 – Proposta Design



Fonte: Própria

De acordo com, o International Council of Systems Engineering (INCOSE), que elaborou o *Systems Engineering Body of Knowledge* (SEBoK) - a Engenharia de Sistemas (SE) tem uma abordagem interdisciplinar que permite o desenvolvimento de sistemas de elevada complexidade (ISO/IEC/ IEEE 15288) (INCOSE, 2015). O Objetivo é definir, de maneira precoce no ciclo de desenvolvimento de um sistema, as necessidades do usuário, assim como as funcionalidades requeridas, documentando os requisitos de forma sistemática, tratando a síntese de projeto e a etapa de validação, de maneira a considerar o problema como um todo para obter um produto de qualidade que atenda às necessidades do cliente/usuário (INCOSE, 2020); (WALDEN *et al.*, 2015).

A engenharia de sistemas introduz métodos e ferramentas que que facilitam o trabalho projetos complexos que fundem diferentes ambientes de desenvolvimento e aplicação como: computação, engenharia, logística e gestão, entre outros (INCOSE, 2020); (INCOSE, 2015); (MARTIN, 2004). O propósito ES é o de abranger representações gráficas e computacionais de um sistema, utilizadas para realizar a comunicação entre os requisitos de dados e suas funcionalidades (essenciais na fase subsequente de design).

Algumas das técnicas utilizadas pela Engenharia de Sistemas são:

- Conceituação: Uma técnica utilizada pela ES em projetos complexos, como por exemplo o design de aeronaves, design de chips, integração de sistemas distribuídos, entre outros. A ES utiliza uma variedade de ferramentas para modelagem, simulação, e análise de

requisitos no intuito de tratar a complexidade;

- Visão Holística: A ES enfatiza na análise e resposta às necessidades dos usuários / clientes além da funcionalidade desde o início do ciclo de desenvolvimento, gerando documentação e requisitos e processamento (*workflow*) associado ao sistema como um todo e não resultado da composição das partes;

- Um determinado sistema pode aumentar sua complexidade se aumentar de tamanho, se aumentar o volume de dados, a quantidade de variáveis, ou aumentar o número de relações entre parâmetros e variáveis. A evolução de algoritmos de controle cada vez mais inteligentes visam a recuperação de erros e situações críticas, manutenção de invariantes, ou o aumento da colaboração entre sistema e usuário;

- Modelos: Umas das técnicas mais importantes utilizadas pela ES é a orientação a modelos, definidos como:

  • uma abstração da realidade capaz de responder a questões específicas sobre o mundo real;

  • uma imitação, analogia ou representação de uma estrutura ou processo real;

  • uma ferramenta conceitual, matemática ou física para apoiar na tomada de decisão.

A razão principal para se utilizar modelos e diagramas em estudos de viabilidade de projetos é prover estimativas para a efetividade, performance e atributos técnicos dos sistemas e custo (NIELSEN *et al.*, 2015).

Por outro lado, é importante salientar que a melhor maneira de entender um sistema é dividi-lo em partes recursivamente até chegar a elementos simples que não precisam fazer parte do processo de design (podem ser reutilizados, por exemplo). Portanto o foco é na funcionalidade (ou nos objetivos) do todo (*holón*) e não nos recursos e elementos constituintes, ou na composição deles.

Seguindo a máxima do *"dividir para conquistar"*, a composição hierárquica de um Sistema é uma estrutura organizacional que pode ser usada para lidar com a complexidade, sem abandonar os conceitos principais enumerados anteriormente. A ISO/IEC/IEEE 15288 define a hierarquia no ciclo de vida de sistema como um conjunto de elementos (ou sub-sistemas) implementados para atender a requisitos especificados (e completos) para cada um destes elementos (INCOSE, 2015). Portanto a indicação é seguir uma abordagem de sistemas composto de sistemas.

A proposta para o design de sistemas de energia mostrado na figura 8, começa com modelagem e análise de requisitos, atendendo a restrições do domínio, para sistemas distribuídos SG e microgrid, onde a comunicação é fundamental, uma vez que estes subsistemas precisam estar interconectados. A proposta inclui alguma arquitetura de referência consagrada na área

de atuação, para ter um maior grau de aplicabilidade, também, incluirá métodos modernos de requisitos orientados a objetivos GORE, especificamente KAOS. E a formalização deve estar presente desde a fase de requisitos utilizando RdP.

3.1.1 Sistemas-de-Sistemas (SoS)

Um Sistema de Sistemas (SoS) é uma forma de integrar sistemas independentes (conhecidos como constituintes) e seus relacionamentos, formando um todo maior que a soma das partes (MAIER, 1998). Consequentemente, um SoS possibilita alcançar objetivos complexos que não poderiam ser alcançados pela soma dos seus sistemas constituintes, ou seja, trabalham de forma colaborativa para obter funcionalidades coletivas. Em um SoS, comportamentos emergentes podem aparecer a qualquer momento e devem ser igualmente previstos e modelados (NIELSEN *et al.*, 2015).

3.1.1.1 Características dos SoS

Um SoS deve idealmente satisfazer a diversas características como (NIELSEN *et al.*, 2015):

- Independência Operacional: constituintes têm suas próprias operações embora trabalhem colaborativamente, como serviços, no escopo do SoS;
- Independência Gerencial: constituintes podem ser gerenciados independentemente e localmente e efetivamente se auto-gerenciam, respectivamente;
- Desenvolvimento Evolucionário: um sistema de sistemas está em constante mudança por adaptação de seus objetivos e funcionalidades; a evolução de cada constituintes é suficiente para denotar a evolução do SoS (marcando uma transição para um novo estado). Os colaboração entre os constituintes é feita pela troca de mensagens e menos pela troca de recursos ou de energia;
- Distribuição geográfica: constituintes possuem localidade (física e topológica) e interagem segundo uma tecnologia um protocolo de comunicação;
- Comportamento Emergente: SoS's possuem funcionalidades que emergem do comportamento coletivo dos constituintes. Essas funcionalidades não podem ser derivadas da conjunção direta do comportamento dos constituintes individualmente.

3.1.1.2 Taxonomia dos SoS

Para efeitos de design é importante discernir entre as diversas possibilidades de SoS's disponíveis. Maier (MAIER, 1998) definiu basicamente três tipos de SoS's (ASSAAD; TALJ; CHARARA, 2017):

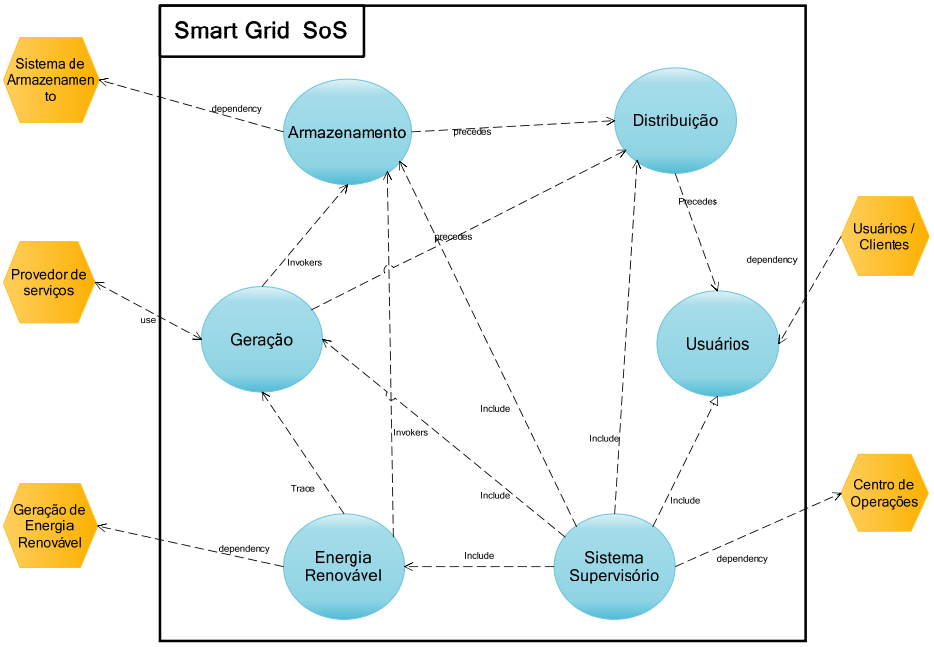
- Dirigido: O sistema é construído para um propósito específico e controlado por uma entidade central, um tipo comumente escolhido para sistemas de defesa ou sistemas críticos;

- Colaborativo: Constituintes colaboram voluntariamente para atingir o objetivo global, mas, ainda que exista um sistema de coordenação central, este não tem poder coercitivo ou controle direto sobre os componentes;

- Virtual: Não há controle central e a colaboração surge da interação entre os componentes. Não existe um objetivo definido univocamente (*social goal*) mas estratégias, que também podem ser classificadas como objetivos ou até invariantes de design que devem ser mantidos.

Os conceitos de SoS são recentes e apresentam uma serie de tópicos de pesquisa em aberto, tais como, a integração entre os sistemas constituintes de um SoS. Algumas questões a serem exploradas são: a complexidade das interações, a complexidade dos sistemas constituintes, incompatibilidade entre interfaces - que podem afetar o funcionamento como serviço - e a evolução dos SoS - que demandam um tratamento formal. A Figura 9, mostra um SoS no contexto das SG's

Figura 9 – Smart Grid SoS



Fonte: Própria

3.1.2 Abordagem orientada a Serviços

Uma abordagem orientada a serviços significa ter um acoplamento adequado entre o Sistema e Usuário que é a base dos serviços. O design como serviços, introduzem o conceito de (SoS+), isso, significa que cada subsistema e agora um serviço (ou integração de vários serviços), onde especial atenção deve ser dada ao acoplamento entre o SoS com o usuário final, devido ao surgimento dos requisitos emergentes. Os requisitos emergentes que surgem quando o sistema está projetado e implementado, ou seja, trata-se das necessidades que aparecem pelo fato de o sistema existir, e que pode mudar os comportamentos dos usuários.

O Figura 9, mostra os (SoS) como serviços da SG, interconectados que compartilham recursos e possuem objetivos comuns. Os elementos no interior do quadro representam os SoS provedores dos serviços e os elementos externos representam os agentes receptores do serviço. Por exemplo o usuário, faz parte do sistema como produtor energia, mas também é um agente consumidor e receptor desses serviços. O que implica em uma dupla responsabilidade tanto no fornecimento, quanto no consumo de energia.

Este tipo de análise considerando (cliente/ provedor) já na fase de requisitos, permite mostrar as interações com o usuário, isto é, suas demandas e responsabilidades para atender os requisitos emergentes.

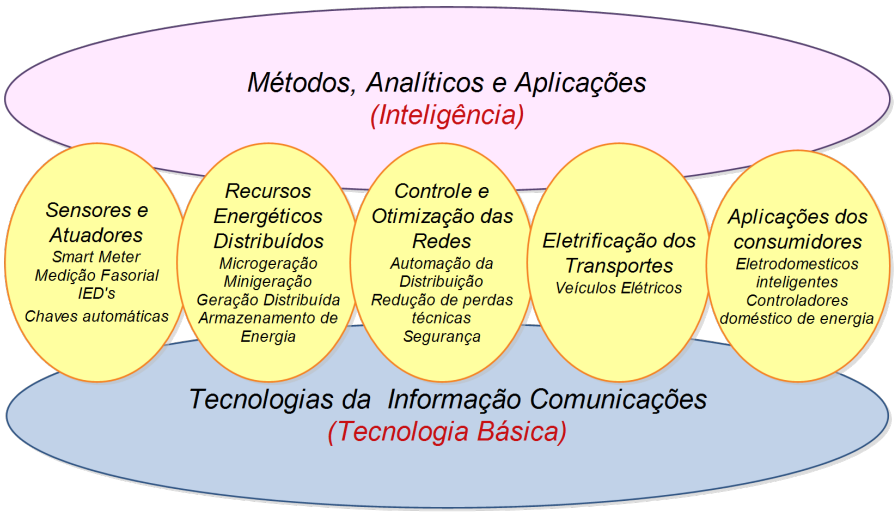
3.2 Visão Geral dos Sistemas Smart Grid

Um sistema SG pode ser definido como a sobreposição de um sistema de comunicação e controle sobre a infraestrutura existente de rede de energia, para permitir a colaboração entre as entidades interessadas (COMMISSION *et al.*, 2008). Segundo (HEFFNER, 2011) a *International Energy Agency* (IEA) considera um SG como uma rede elétrica que usa tecnologias digitais para supervisionar e controlar o transporte de energia elétrica a partir de fontes de geração encontrando uma variedade de demandas e usuários. A rede em objetivos gerais de manutenção do fornecimento com um mínimo de interrupção, evitando desperdícios, minimizando custos, e privilegiando sustentabilidade. Portanto pode ser associada a um SoS virtual, na classificação acima.

Assim, existe uma forte integração tecnológica na comunicação, informação, e em sistemas de automação, sensoriamento e medição, visando aumentar consideravelmente a capacidade de atender domínios com fontes intermitentes e distribuídas de energia, com altos requisitos de confiabilidade, e baixo impacto ambiental, e que seja compatível com os novos mercados de energia (SECHILARIU; LOCMENT, 2016).

Em termos gerais, uma SG é considerada como um conglomerado de tecnologia que abrange desde a produção de energia, passando pela transmissão, até o consumo (FALCÃO, 2010). Compreende a convergência do fluxo de energia elétrica, do fluxo de informações, assim

Figura 10 – Smart Grid



Fonte: adaptado de (RODRIGUEZ *et al.*, 2007)

como do fluxo de negócios com base na automação, informação e comunicação, conformo mostrado na Figura 10. Estas redes estão ganhando popularidade em empresas de serviços públicos de energia e institutos de pesquisa, e outras aplicações(SILVA *et al.*, 2014).

A principal vantagem das SG é a capacidade de geração dos Recursos Energéticos Distribuídos (RED) e a forma de gerenciar o consumo e a produção de energia, graças a um fluxo bidirecional de energia e dados entre geração, transmissão, distribuição e consumo de energia (FALCÃO, 2010).

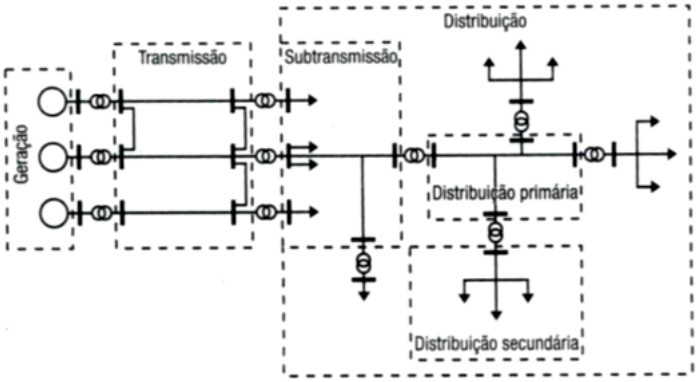
A geração de energia é o primeiro passo na cadeia de valor do SG. Inclui diferentes tipos de fontes de energia renovável e não renováveis. A transmissão de energia usa tecnologias de monitoramento e controle que permitem a comunicação com a etapa de distribuição (RODRIGUEZ *et al.*, 2007). O consumo é o último elo na cadeia de valor da SG e envolve os usuários de qualquer categoria: residencial, comercial, e industrial. Assim, é cada vez mais comum para o consumidor gerar energia elétrica, tornando-se um *prosumer*, especialmente através de métodos alternativos de produção de energia renovável. Por esse motivo, é muito importante supervisionar o consumo e a produção para otimizar o serviço e evitar desperdícios (PAPAEFTHYMIOU *et al.*, 2010).

3.2.1 Desafios e Oportunidades das Smart Grid

Na atualidade os Sistemas elétricos de potência (SEP) são constituídos por centrais elétricas, subestações de transformação e de interligação, linhas e receptores, interligados entre si. Estes sistemas englobam geração, transmissão e distribuição de energia elétrica. O sistema elétrico de potência compreende todas as formas de geração de energia elétrica e sua transmissão até os consumidores (FILHO; MAMEDE, 2000). A Figura. 11, mostra um diagrama unifilar

típico de um SEP, destacando o sistema de transmissão operando radialmente.

Figura 11 – Diagrama Unifilar de um SEP



Fonte: adaptado de (FILHO; MAMEDE, 2000)

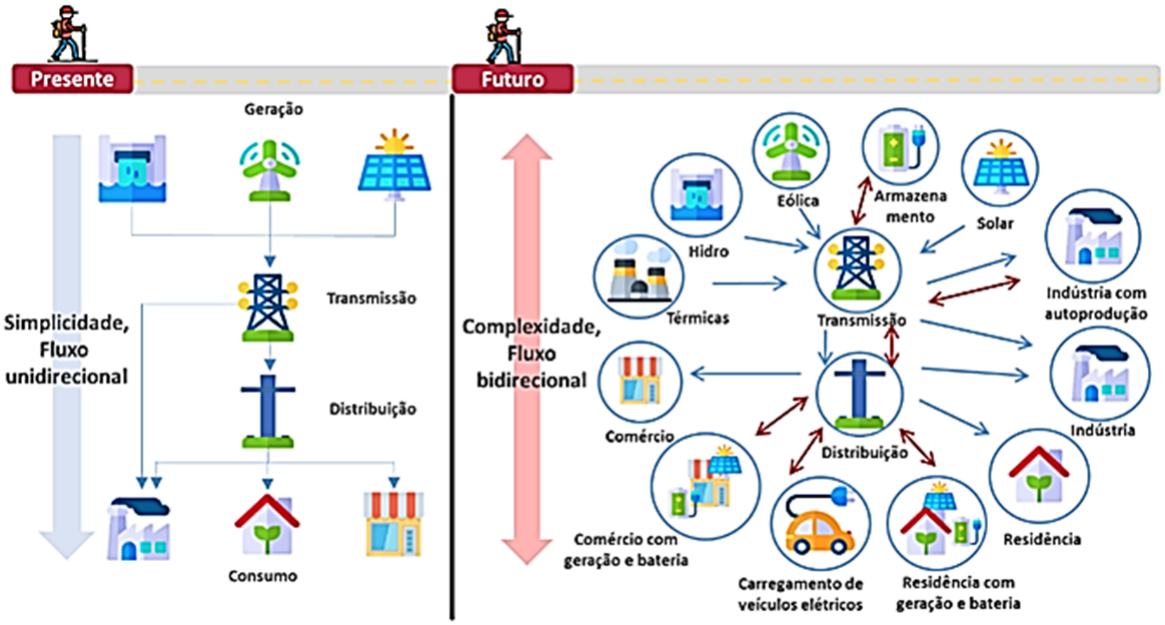
Sistemas tradicionais de energia são caracterizados por serem centralizados normalmente através de uma concessionaria com vistas a explorar a economia de escala; o fluxo energético é unidirecional, isto é, energia gerada é transportada da geração através de linhas de transmissão de alta tensão, passando pelas redes de distribuição até os consumidores finais, ou seja, a geração segue a carga. Os usuários tem comportamento passivo, são apenas consumidores. Existe necessidade de equilíbrio instantâneo entre a carga e a geração devido à impossibilidade de estocar energia (NYSO, 2017).

Este modelo tradicional de centralizado vem sendo modificado devido ao surgimento dos sistemas SG, estes modernos sistemas de energia se caracterizam por serem distribuídos onde a produção de energia é baseada especialmente em GD e combinadas a centrais geradoras. O fluxo energético é bidirecional, da geração até os usuários, e dos usuários para a rede, aqui a comunicação também é bidirecional. Existe capacidade de armazenamento, pois a energia pode ser estocada normalmente e sistemas de banco de baterias. Neste novo modelo os consumidores tornam-se prosumidores, participando ativamente e contribuindo para o equilíbrio do sistema elétrico (ELÉTRICA, 2018); (NYSO, 2017); (BLAABJERG *et al.*, 2006).

A auto-restauração (*Self-Healing*) é outra característica deste sistema que é possível devido à automação da distribuição, especialmente através da proteção inteligente, que reduz o número de interrupções do consumidor durante uma condição de contingência (ELÉTRICA, 2018). A flexibilidade deste sistema depende da sua capacidade de responder as variações bruscas da oferta e demanda, para garantir assim a continuidade e estabilidade do fornecimento de energia. Neste novo cenário o sistema legado deve coexistir com novas tecnologias de controle e operação que pode ser centralizado, ou distribuídos. Por isso, o domínio de design deve ser mais flexível, ou seja, não apenas considerando os aspectos técnicos, se não também, a forma de integrar outros subsistemas e especialmente acoplamento com o usuário (POSTIGO; SILVA, 2020).

Para que isso se torne realidade, será necessária a aplicação massiva de diversas tecnologias, tais como: sistemas de sensoriamento, o design de modelos verificados formalmente - em alguns casos em tempo real - e tecnologia que permita a comunicação bidirecional entre a concessionária (ou gestor centralizado) e os usuários, e também entre dispositivos ao longo da rede. Na Figura 12 mostra a comparação entre as redes elétricas tradicionais e as novas tendência de estruturação para as redes elétricas do futuro (SG). Também, é possível observar dispositivos para dar suporte ao novo mix de geração e integração da energia, visando o uso das energias renováveis e diferentes formas de armazenamento (BERNAL-AGUSTÍN; DUFO-LÓPEZ; RIVAS-ASCASO, 2006) (VÁZQUEZ; MÁRQUEZ; MANZANO, 2008).

Figura 12 – Smart Grid Presente e Futuro



Fonte: baseada em (NYSO, 2017)

Nessa nova conjuntura temos a um mercado de energia aberto, onde se destaca uma nova configuração de serviços associado ao consumo, e portanto a ao usuário final. Este novo consumidor/produtor (*prosumer*) possui novas responsabilidades, tais como o geração e armazenamento de energia local, o que, além de apontar para um SoS virtual afeta o processo de negócios e o precificação da energia (BLAABJERG *et al.*, 2006).

Nesse contexto, para atender o aumento da demanda de energia nos próximos anos, espera-se um crescimento expressivo da geração distribuída proveniente de fontes renováveis de energia. Inclusive no futuro próximo, os veículos elétricos exercerão a função dupla de carga, com suas baterias carregadas pela rede, e de fonte de energia, enquanto acumuladores de energia, poderão injetar energia na rede elétrica. Existe interesse dos governos por tecnologias limpas, motivo pelo qual estas têm recebido um maior incentivo, especialmente tecnologias como a solar-fotovoltaica, eólica e biomassa (FALCÃO, 2010).

A ênfase em comunicação traz à tona a consideração da norma IEC 61850 no design de sistemas SGs.

3.3 A IEC 61850 no contexto das microgrids

Inicialmente, a Arquitetura IEC 61850 foi desenvolvida com o intuito de ser uma arquitetura de comunicações para facilitar o design de sistemas com funções de proteção, controle, monitorização e diagnósticos em subestações (NAUMANN *et al.*, 2014). Esta norma foi desenvolvida pela International Electrotechnical Commission, (IEC), Technical Commitee 57, composto por um grupo de fabricantes (ABB, Alstom, Schneider, SEL, Siemens, Toshiba, etc.) e por provedores (Electricité de France, Iberdrola, Hydro-Quebec, etc.), com o objetivo de melhorar a interoperabilidade dos equipamentos (CLAVEL *et al.*, 2015).

A referida arquitetura não trata apenas da comunicação, mas também da modelagem das informações sob medida para as necessidades da indústria de energia elétrica (BERGER; INIEWSKI, 2012). Adicionalmente, define uma linguagem de configuração baseada em Extensible Markup Language (XML), padronizado para a engenharia/configurações de dispositivos de automação das subestações.

Embora a arquitetura IEC 61850 originalmente abordasse unicamente à automação de subestações, já existem modelos de informação adicionais, definidos com base na arquitetura IEC 61850, para outros domínios como energias renováveis (BERGER; INIEWSKI, 2012). Em 2010, o Instituto Nacional de Padrões e Tecnologias dos Estados Unidos (NIST) reconheceu a IEC 61850 como um dos principais facilitadores da implementação dos sistemas SG. (ZHAO *et al.*, 2015).

Um dos objetivos principais da arquitetura IEC 61850 era resolver problemas de interface e padronizar a comunicação para evitar a dependência de protocolos específicos dos fabricantes. Visava também criar um mecanismos para o processo de automação e funções de controle que permitissem a sistemas de elétricos evoluir para um Smart Grid (NAUMANN *et al.*, 2014).

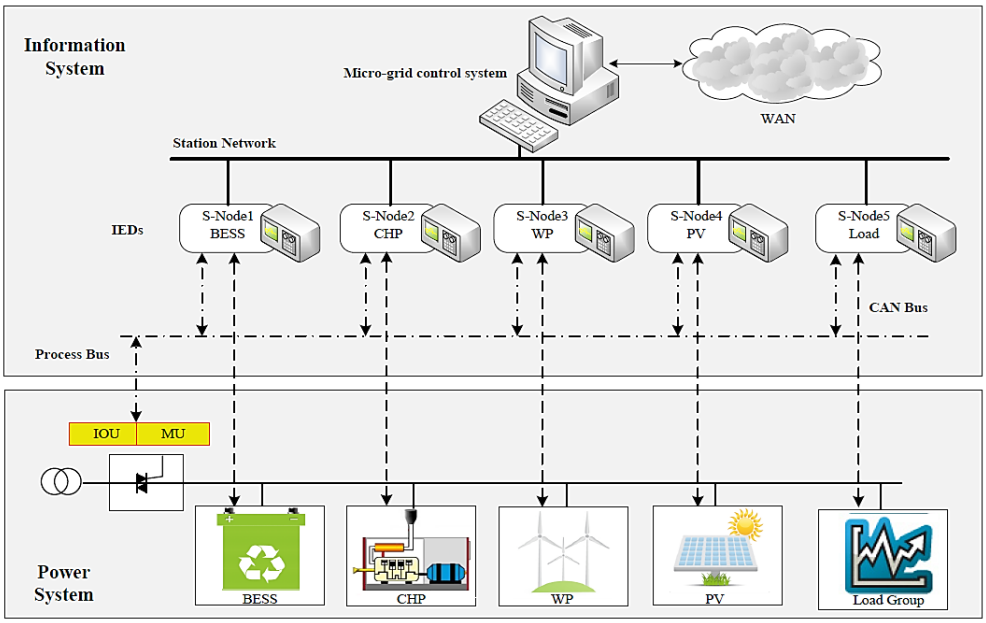
A principais vantagens no uso desta norma são (VASUDEVAN *et al.*, 2012) (MACKIEWICZ, 2006):

- Interoperabilidade: onde um sistema tem interfaces previamente preparadas para trabalhar com outros sistemas, sem qualquer restrição (uma aplicação especial desta propriedade seria em sistemas orientados a serviço);
- Substituição de todas as ligações por cabos Ethernet (no caso de algumas aplicações é usada fibra ótica), o que fez reduzir quantidade de cabos, a estruturação, e os custos;
- Metodologias orientadas a objeto e o uso da Substation Configuration Language (SCL) usada como linguagem de referência (no caso de substações).

O SCL é colocado em arquivos em formato XML, que podem ser usados para troca de informação entre dispositivos. Com a configuração disponível em um formato padronizado, é possível usar essa informação fora da subestação (BRUNNER, 2008).

No contexto das microgrids, a arquitetura IEC 61850 é composta por duas partes: um sistema de informação e um sistema de potência, conforme mostrado na figura 13. O sistema de potência é a base física para realizar a função da microgrid, e é composto por unidades de Geração Distribuída (GD), unidades de carga, Sistemas de Armazenamento de Energia (SAE), e unidades auxiliares. O sistema de potência, segundo a IEC 61850, orienta o modo de operação, isto é, como cada RED acessa o barramento do microgrid.

Figura 13 – Arquitetura IEC 61850



Fonte: (DENG *et al.*, 2015)

O sistema de informação da microgrid, conforme a IEC 61850, é dividido em três níveis: o nível de estação, o nível de vão, e o nível de processo. A extensão da norma IEC 61850-7-420 fornece um modelo de informação e Nodos Lógicos (LNs) típicos para RES's para o nível de processo, que incluem controladores, geradores, conversores de energia, inversores, e sistemas auxiliares (como dispositivos de medição, e proteção) (BERGER; INIEWSKI, 2012).

As extensões da norma IEC 61850 7-1, 7-2, 7-3 e 7-4 fornecem métodos de modelagem de comunicação e informação e LNs para os equipamentos físicos da microgrid. A IEC/TR 61850-90-7, fornece modelos orientados a objetos de inversores, SAE, etc. Desta forma, baseado nas informações fornecidas pela norma IEC 61850, a modelagem das informações dos dispositivos e equipamentos de microgrid no nível de processo pode ser alcançada.

O nível de vão é composto principalmente de S-Nodes, como mostrado na figura 13 . Cada nó tipo-S é um IED, responsável pela troca de informações entre o nível estacionário e

o nível do processo. Assim, o S-Node recebe e analisa as informações de estado e os valores medidos do nível do processo, e os transmite- para o nível de estação. Também, o S-Node recebe e analisa e controla as configurações do nível de estação, e os transmite para o nível de processo para poder gerir os equipamentos da microgrid. A extensão IEC 61850 7-4 fornece LNs fundamentais para o controle, monitoramento, proteção e outras funções, de modo que o S-Node pode construir o modelo de informação de acordo com os requisitos e aplicações dos equipamentos.

O nível de estação corresponde ao sistema de controle da microgrid com o *Supervisory Control and Data Acquisition* (SCADA) na estação de controle, e é responsável pela operação de monitoramento da microgrid, pelo controle de estabilidade, otimização de energia, pela troca de instruções, planejamento, e gestão da geração harmonizada com o centro de programação. IEC 61850-6 apresenta a linguagem de configuração e procedimentos de trabalho das IED assim como, ferramentas de configuração, e proporciona um *framework* que inclui os equipamentos do nível de processo, os S-Nodes do nível de vão e o sistema de comunicação.

Em torno da estrutura de três níveis, apresentados na figura 13 , pode haver diferentes redes de comunicação,

- Barramento de Estação localizada entre o nível de estação e nível de vão: Troca mensagens de velocidade média (tais como o controle automático e dados de energia e de operação), arquivos (como eventos de proteção e valor de configurações) e mensagens de sincronização, especificação e de Mensagem de Manufatura (MMS), ou seja, mensagens comuns.

- Barramento de Processo localizada entre o nível de vão e o nível de processo: trocas de mensagens de alta-velocidade (como comandos de *trip* e instruções de bloqueio) e mensagens de medição como parâmetros elétricos usando mensagens do tipo *Generic Object Oriented Substation Event* (GOOSE).

- Barramento (BAY) localizada entre o nível de vão e os equipamentos da microgrid no nível de processo: fornece comunicação de dados em tempo real, e garante a confiabilidade das transmissões de dados pela sua estrutura redundante.

  Os objetos - podem ser Dispositivos Eletrônicos Inteligentes (IEDs) e Unidade Terminal Remota (RTU) e controladores de lógica programável (CLP's). Estes objetos desempenham controlo local e enviam/recebem sinais para/de outros objetos ou mestre SCADA. Podem dividir-se em dois níveis: equipamento primário e bay level (inteligência local em forma de IED's, RTU's (ELGARGOURI; ELFITURI; ELMUSRATI, 2013).

3.4 Engenharia de Requisitos no Design de Sistemas

O processo de design de sistemas em geral, compreende um conjunto de atividades que engloba métodos, ferramentas e procedimentos, com o objetivo de produzir sistemas que atendam aos requisitos. A Engenharia de Requisitos (ER) é a primeira etapa do processo de desenvolvimento de um sistema, e está associada à identificação dos objetivos a serem atingidos pelo sistema a ser desenvolvido (LAMSWEERDE, 2004). Um dos principais objetivos da ER é melhorar a modelagem de sistemas e a capacidade de analisá-los, possibilitando maior compreensão de suas características, antes da implementação.

As atividades da ER clássica são:

- eliciação dos requisitos: responsável pela aquisição do comportamento pertinente às funcionalidades do sistema e à sua relação com o ambiente ao redor;
- modelagem: descreve as informações dos requisitos através de uma representação específica, que pode ser dividida na modelagem do problema de planejamento a ser resolvido, e na modelagem do mundo que interage com este planejamento, definida como domínio de trabalho.

A análise do domínio consiste em verificar e validar a consistência dos modelos de requisitos obtidos. (WERNECK; OLIVEIRA; LEITE, 2009); (LAPOUCHNIAN, 2005); (SOMMERVILLE, 2007).

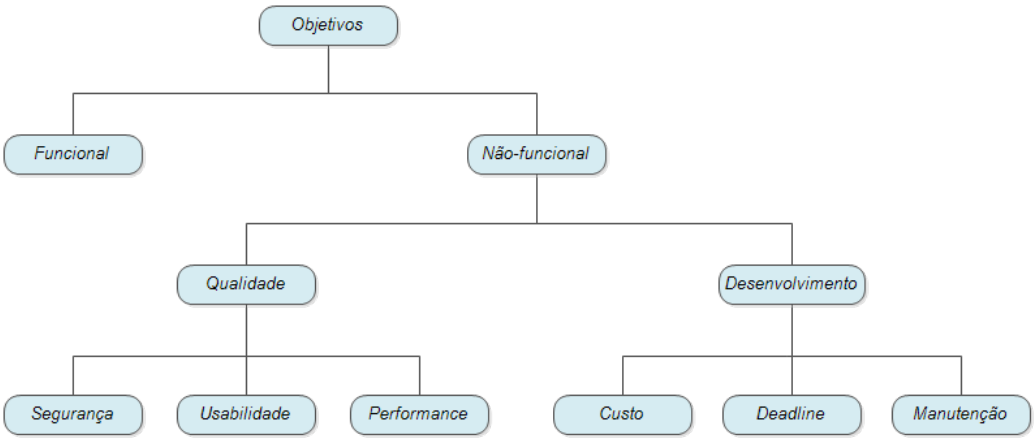
Os problemas mais comuns que podem surgir na fase de requisitos, estão associados à complexidade, à volatilidade, à ambiguidade e ao desentendimento entre *stakeholders*. De fato, a obtenção (eliciação) de requisitos é considerada uma das tarefas mais complexas no desenvolvimento de um sistema. Isso se deve principalmente à dificuldade dos analistas em extrair informação dos usuários, as quais podem estar incompletas.

Outro fator que torna complexo o processo de obtenção dos requisitos, é o fato de que estes requisitos são volateis; uma vez que a visão do usuário e dos stakeholders muda diante da perspectiva de ter um sistema automatizado, gerando requisitos emergentes. Por outro lado, dificilmente as especificações permanecem inalteradas durante o processo, e os requisitos mudam durante o processo de modelagem e análise (o chamado *scope creep*).

Existem dois tipos fundamentais de classificação de requisitos: Requisitos Funcionais (RF) e Requisitos Não-Funcionais (RNF) (LAMSWEERDE, 2004) como mostrado na figura14. Os RF especificam as funções que o sistema ou seus componentes deverão ser capazes de realizar. Podem ser definidos como atividades que o software ou sistema faz, ou fará, com o objetivo de capacitar os usuários a realizar suas funções (ou receber o serviço). Esses requisitos devem ser bem definidos e descritos claramente para que não exista ambiguidade na sua funcionalidade (DORFMAN; THAYER, 1990).

Os RNF descrevem as restrições e condições dos requisitos funcionais, como por exemplo: adaptabilidade, confiabilidade, eficiência, performance, portabilidade, segurança, flexibilidade, usabilidade, entre outros (CHUNG *et al.*, 2012). Os requisitos não-funcionais são geralmente mensuráveis, e podem estar associados a uma medida ou valor de referência.

Figura 14 – Classificação Objetivo



Fonte: baseado em (LAMSWEERDE, 2009)

3.4.0.1 Métodos para obtenção de Requisitos

Dentro da Engenharia de Requisitos, existem vários métodos para obter requisitos, alguns mas atuais como os orientados a objetivos e outros muito utilizados como o *Use-Case*. Cada um desse métodos envolve um conjunto de atividades interligadas, que tem a função de criar um documento de requisitos do sistema.

3.4.0.1.1 Unified Modeling Language - UML

A UML é considerada por muitos pesquisadores como uma linguagem sólida para modelar sistemas, e tem sido usada com sucesso em diferentes projetos, muitos deles ligados ao desenvolvimento de software, mas também em projetos de natureza mais geral envolvendo sistemas automatizados, onde predomina o controle baseado em software (RUMBAUGH; JACOBSON; BOOCH, 2004).

Assim, a UML oferece um conjunto de notações e diagramas para representar requisitos eliciados observando diferentes aspectos (onde predomina o funcional) sem que haja uma forte interferência entre eles. Por outro lado, é possível observar tanto no mercado quanto na academia o excesso de zelo funcional, dando um poder maior aos chamados casos de uso, cuja inclusão no conjunto de diagramas base é questionado (DOBING; PARSONS, 2008).

A UML não é uma metodologia de desenvolvimento, portanto não oferece um método para projetar sistemas. Porém, serve de ajuda para modelar, visualizar e documentar os mesmos. Assim, são necessárias técnicas para verificação e validação para assegurar a consistência das

descrições (DOBING; PARSONS, 2008). A verificação e a validação de sistemas complexos desenvolvidos em UML não é um processo simples, seja pela dificuldade em tratar sistemas complexos com diagramas onde existe uma certa superposição de informação, seja pela dificuldade em escolher um conjunto mínimo de diagramas. Esse último aspecto pode dificultar o reuso, quando repositórios de projetos são disponibilizados, como apregoado por várias arquiteturas já descritas.

Embora a UML possua um alto poder de expressão gráfica, existem propriedades e restrições de sistemas que não podem ser expressas adequadamente em um diagrama visual (SALOMON, 2017). Mesmo os mecanismos de extensão da UML podem ser insuficientes para compor uma especificação de requisitos completa (RUMBAUGH; JACOBSON; BOOCH, 2004). Para enfrentar esse problema foi criada a Object Constraint Language (OCL), uma linguagem formal, fácil de ler e escrever.

A OCL é uma linguagem composta de expressões para especificar restrições sobre modelos orientados a objetos ou outros artefatos da linguagem UML (CABOT; GOGOLLA, 2012). A OCL é uma linguagem precisa, textual e formal. Expressões em OCL não possuem efeitos colaterais: apenas retornam um valor, não alterando nenhuma informação do modelo. Sua avaliação é instantânea (os estados dos objetos não podem ser alterados durante essa avaliação) (CABOT; GOGOLLA, 2012). Além disso, utilizando expressões OCL, é possível a navegação entre as classes de um modelo UML, a partir de uma classe-base, o que fornece o contexto da expressão.

A conjunção entre UML (funcional) e a OCL (importante para a especificação de restrições e requisitos não-funcionais) supre o necessário para a fase inicial de eliciação de requisitos sobre o sistema a ser desenvolvido, bem como o seu domínio de entorno (CABOT; GOGOLLA, 2012); (SALOMON, 2017). Entretanto, como já foi assinalado, o sistema SG é na verdade um serviço, que deve ser acoplado ao usuário (final), agora na condição de *prosumer*. Portanto a abordagem UML/OCL traz um problema: não ter nenhum mecanismo para inserir este *prosumer* no processo de design, modelando, por exemplo o acoplamento entre ele e o sistema. Portanto, a despeito do que propõe a Intelligrid, procuramos uma alternativa para a modelagem de requisitos que pudesse superar esta dificuldade.

3.4.0.1.2 Engenharia de Requisitos Orientada a Objetivos

O Goal Oriented Requirements Engeneering (GORE) é uma abordagem diferente para a modelagem, análise de requisitos (HORKOFF *et al.*, 2016). Trata-se de uma abordagem que busca eliminar a dicotomia entre os requisitos funcionais e não-funcionais, pelo menos na fase inicial de eliciação. Ao invés de funcionalidades, o processo e eliciação se inicia pele identificação dos objetivo que o sistema deve alcançar (LAMSWEERDE, 2009). Estes objetivos representam as informações mais estáveis do sistema (invariantes) que devem orientar o *social*

goal, alvo da contribuição de componentes. Estes objetivos podem ainda ser sub-divididos em sub-goals, que por sua vez poderão ser objetivos de sub-sistemas em um modelo SoS.

Por outo lado os objetivos são orientados ao problema (ou *project statement*), enquanto outras técnicas clássicas são orientadas para a solução. Assim, os objetivos tornam-se uma excelente ferramenta de comunicação entre as partes interessadas na solução de um determinado problema. Portanto, um objetivo é uma declaração de intenções que o sistema devera satisfazer através da cooperação entre agentes, um componente ativo do sistema e responsável pelas operações e pelos requisitos (LAMSWEERDE, 2003).

Objetivos devem ser formulados em termos de fenômenos (ocorrências, eventos, fatos, situações, circunstancias, casos, etc.) e compartilhados entre os agentes (passivos ou ativos, homens ou máquinas). Estes fenômenos são controlados por alguns agentes e monitorados por outros, configurando um quadro onde cada agente tem um conjunto de responsabilidades. O conjunto de todos os agentes do sistema e suas responsabilidades define o escopo do sistema. Por outro lado, as propriedades do domínio são consideradas asserções sobre o mundo dos problemas, tais como as leis físicas (ou restrições).

O método GORE tende a facilitar a fase de modelagem dos requisitos, pois funciona como um mecanismo de validação dos mesmos que resulta em um melhor entendimento, e ajuda na eliciação e elaboração dos requisitos. De acordo com Lamsweerde (LAMSWEERDE, 2003), uma ER eficiente deve determinar os objetivos, sua operacionalização em especificações de serviços e restrições, a atribuição de responsabilidades aos agentes que se relacionam com os requisitos, reforçando a rastreabilidade, que associa os requisitos aos agentes que os geraram.

Um modelo GORE é representado em linguagens orientada a objetivos (HORKOFF *et al.*, 2016). Essas linguagens são geralmente gráficas e oferecem uma sintaxe visual. Por exemplo, i * (YU, 1997), KAOS (DARDENNE; LAMSWEERDE; FICKAS, 1993) são linguagens gráficas, mas existem também linguagens textuais, como por exemplo o *Goal Based Requirements Analysis Method* (GBRAM).

Para Laplante (LAPLANTE, 2017) os objetivos têm um nível de abstração superior para o negócio, organização ou sistemas, enquanto um requisito especifica como esse objetivo será atingido. Essa abordagem é particularmente adequada para representar a interação entre *hardware, software*, pessoas, instalações e procedimentos, organizados para atingir objetivos comuns (*social goals*) (BUEDE; MILLER, 2016).

Na última década, a popularidade da abordagem e utilização do método GORE aumentou drasticamente (PARVEEN; IMAM, 2017). A principal razão para isso é que, estes métodos facilitam a decisão entre uma solução qualquer e uma solução otimizada, onde os requisitos "implementam" os objetivos da mesma forma que os programas implementam especificações do *design* do sistema. Portanto, a eliciação dos requisitos é considerada a base para a solução do problema alvo, e deve ser baseada em objetivos, e deve ser clara, para que possa ser facilmente

entendida pelos *stakeholders* (LAPOUCHNIAN, 2005).

3.4.0.1.3 Método KAOS

O KAOS (*Knowledge Acquisition in Automated Specification*) é um método de modelagem de requisitos que abrange a identificação dos requisitos de negócios, de uma rede de objetivos, operações e agentes responsáveis para satisfazer estes objetivos. O KAOS, gera um diagrama visual causal, e tem uma forte disciplina para gerar especificações formais em LTL (*Linear Temporal Logic*). Foi desenvolvido pela Universidade de Oregon (USA) em conjunto com a Universidade de Louvain (Bélgica) em 1990 (PONSARD; DARIMONT; MICHOT, 2015).

O ObjectivER foi a primeira ferramenta comercial projetada especificamente para suportar a modelagem usando KAOS, e desenvolvida pela *Respect-IT*, uma empresa belga com experiência prática em projetos industriais de grande porte aplicados em diversos setores (RESPECT, 2007). Portanto, a opção seguida neste trabalho tem, além de um apelo teórico e estratégico, uma base prática para a sua aplicação, tornando-a uma opção real, tanto para a academia como para o mercado.

O método KAOS visa a construção um modelo formalizável, para adquirir e estruturar requisitos, com uma linguagem de aquisição associada e um método de modelagem de requisitos. Define-se os objetivos em diferentes níveis de abstração, desde objetivos de alto nível relacionados com os objetivos organizacionais, estratégicos, especificados difusamente *loosely coupled*, até objetivos de mais baixo nível com especificações mais técnicas, mais detalhadas, e relacionadas com o desenho do sistema (LAMSWEERDE, 2009). Eventualmente objetivos de baixo nível poderão ser satisfeitos reutilizando sub-sistemas.

Este método consiste em dois níveis de representação: primeiro uma descrição do problema a ser resolvido usando modelagem visual (gráfica), capturando as restrições a serem cumpridas e consolidando os requisitos funcionais e não-funcionais. Em seguida, esta representação será transformada em uma notação formal em Logica temporal linear (LTL) (BAIER; KATOEN, 2008), ou, através de um algoritmo desenvolvido pelo D-Lab da USP, em Redes de Petri.

O método KAOS objetiva:

- Esclarecer as descrições dos problemas, entendendo conceitos relevantes para a descrição do problema;
- Melhorar o processo de análise de problemas, fornecendo uma abordagem sistêmica para estruturar requisitos;
- Definir as responsabilidades de todas os agentes envolvidas no projeto;

- Permitir que as partes interessadas no projeto (usuários, *stakeholders*, etc.) se comuniquem de forma fácil e eficiente sobre os requisitos.

Observa-se quer na maioria de documentos de requisitos produzidos utilizando métodos tradicionais como UML, apenas descrevem soluções, relacionadas a funções, processos e estruturas de dados, etc. Entre tanto, uma análise de requisitos com o KAOS é muito mais do que uma descrição limitada da solução, também fornece:

- Rastreabilidade: O KAOS fornece uma relação direta entre a descrição do problema e a descrição do que se espera do sistema a ser desenvolvido. Existe uma correspondência bidirecional entre o problema e os espaços de solução, fundamental não apenas para quem analisa requisitos esteja seguro sobre o sistema a ser construído, mas também para entender a relação entre o contexto e os objetivos que permita fazer um design correto;

- Completude: A documentação produzida pelo KAOS atende a cinco critérios de completude e é portanto adaptável a um desenvolvimento *model-based*. Um modelo KAOS completo não deixa espaço para subjetividade: requisitos e operações têm um responsável conhecido, incorporando o conceito de *traceability*. Nesse contexto, uma documentação online seria muito importante, o que a ferramenta ObjetivER ainda não fornece.

O método KAOS oferece um conjunto de diagramas para a análises de requisitos, composto por quatro submodelos, mostrados na Figura 15 (CAILLIAU; LAMSWEERDE, 2013):

1. O modelo central é o modelo de objetivos que captura e estrutura e as propriedades invariantes do sistema e os requisitos que devem instanciar estes objetivos;

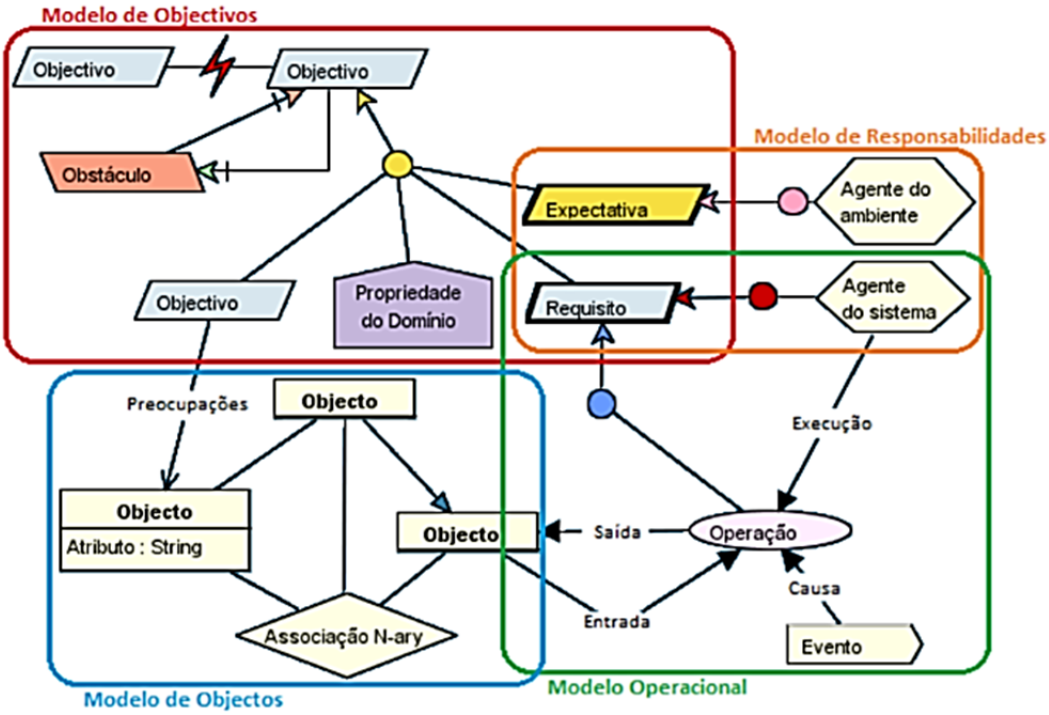
2. O modelo de objetos define os agentes (humanos e máquinas), ativos ou passivos, que contribuem para o funcionamento do sistema;

3. O modelo de operações resume todos os processos e comportamentos que os agentes precisam ter para cumprir os requisitos, (responde o “quando?”). Processos são expressos em termos de operações realizadas por agentes.

4. O modelo de responsabilidades, associam requisitos ou restrições a agentes instanciando a relação de rastreabilidade. Assim, caso seja necessário rever ou alterar um requisito é possível determinar de imediato quais agentes seriam afetados e até as operações associadas a estes.

No processo de ER, técnicas de análise formais ou semi-formais complementam-se em aplicações reais para fornecer a garantia de correção e integridade dos requisitos. O KAOS permite a representação dos elementos desses diagramas com o formalismo da lógica temporal (LTL, Linear Tree Logic).

Figura 15 – Modelo KAOS



Fonte: baseado em (RESPECT, 2007)

3.4.0.1.4 Lógica Linear Temporal-LTL

A lógica temporal é uma extensão da lógica proposicional, possuindo uma notação matemática intuitiva permite expressar propriedades de um sistema de transição (*transition system* em execução, incluindo novas modalidades temporais (BAIER; KATOEN, 2008).

A abordagem básica do tempo na lógica pode ser tratada como linear ou ramificada. Na abordagem linear, a noção de tempo é qualitativa e definida sobre um caminho, ou seja, sobre uma sequência de estados em que para cada estado existe apenas um possível sucessor em um dado instante de tempo, enquanto na ramificada existe uma ramificação definida por uma estrutura de árvore, em que para um instante de tempo a execução do sistema se divide em sucessivos caminhos alternativos.

Dentre as lógicas utilizadas para formalização, destacam-se LTL e a Computation Tree Logics (CTL) (BAIER; KATOEN, 2008). A lógica LTL é um formalismo que visa lidar com aspectos da implementação do sistema e questões de imparcialidade.

O domínio do tempo na LTL é discreto e linear, ou seja, há apenas um sucessor para cada estado. O CTL , por sua vez, possui uma estrutura em árvore, ou seja, possui um nó principal e vários ramos, e é baseado no tempo ramificado, (BAIER; KATOEN, 2008).

Uma transição pode ser representada formalmente em termos de sentenças LTL, como:

$$C \Rightarrow \Theta\, T$$

Figura 16 – Elementos KAOS

Elemento	Descrição
Objetivo	Objetivos são propriedades desejadas do sistema, expressas pelos interessados.
Requisito	Um requisito é um tipo de objetivo de baixo nível é atribuído a um agente do sistema a ser implementado.
Expectativa	A expectativa é um tipo de objetivo a ser atingido por um agente do ambiente do sistema.
Obstáculo	Obstáculos, são eventos indesejados ou restrições que impedem que os objetivos sejam satisfeitos.
Agente	Os agentes são seres humanos ou componentes automatizados, responsáveis pela realização de requisitos e expectativas.
Domínio	Domínio é uma descrição sobre contexto ou meio ambiente e sujeita a mudanças. (lei física, regulamento ou uma restrição, etc.)
	Conectam requisitos ou subjetivos com os objetivos, que representa uma atribuição de refinamento.
	Conectam um agente do contexto com um requisito.
	Conectam um agente do ambiente com uma expectativa.
	Indica que um obstáculo está impedindo o objetivo de ser satisfeito.

Fonte: Adaptado de (RESPECT, 2007)

onde:

- C é uma condição atual;
- T é uma condição alvo;
- Θ representa um dos operadores LTL, mostrados na Tabela 2.

Tabela 2 – Temporal Logic Operators

Operador	Descrição
$\bigcirc\ \phi$	manter no próximo estado
$\Diamond\ \phi$	Eventualmente no futuro
$\Box\ \phi$	Sempre no futuro
$\forall\ \phi$	Para todo.
$\exists\ \phi$	Existe.

A representação formal prescrita pelo método KAOS é baseada em LTL, mas também pode ser representada por uma representação formal estado-transição baseada em grafos. As

rede de Petri desempenham muito bem este papel. Como trataremos de sistemas automatizados, preferimos neste trabalho usar as redes de Petri para formalizar e validar requisitos.

3.4.0.1.5 Exemplo de aplicação

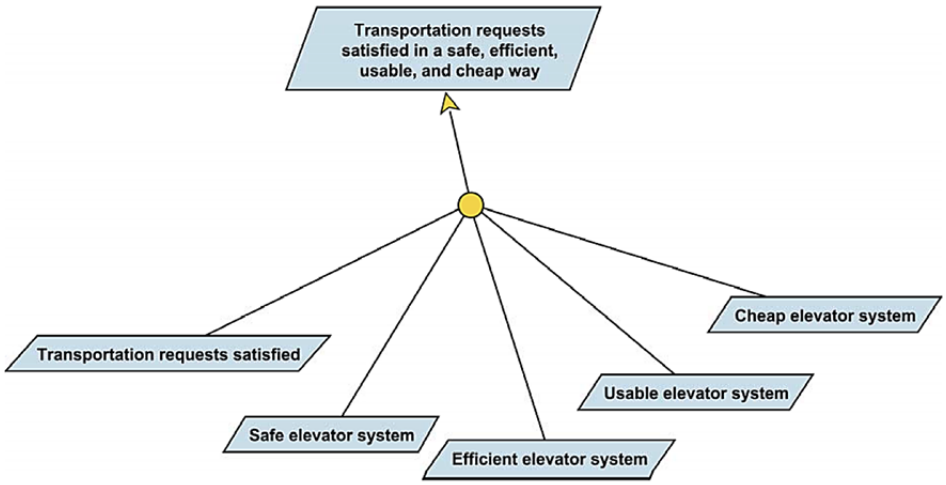
Para melhorar o entendimento com relação ao método, apresenta-se a seguir um exemplo, retirado do próprio tutorial KAOS (RESPECT, 2007). Neste exemplo a ideia é elaborar para uma determinada empresa um sistema de automação para o controle de um elevador. O sistema deve incluir melhorias de desempenho e qualidade. O sistema de automação usado pela empresa foi desenvolvido sem nenhuma metodologia para a eliciação dos requisitos. A proposta para o desenvolvimento do novo sistema de automação será utilizada a metodologia KAOS.

1. Modelo de Objetivos:

Neste primeiro modelo de objetivos mostra-se a visão sistêmica geralista do que o sistema deve atender. Assim, o objetivo é construir um sistema que atenda às necessidades das partes interessadas, isto é, objetivos funcionais e não funcionais. Os objetivos não funcionais são: o sistema deve ser seguro, barato, utilizável e eficiente; o sistema deve ser sustentável; o sistema deve atender as leis vigentes.

Assim o objetivo principal, "Garantir a solicitação de transporte segura, ser eficiente, usável e barato", pode ser refinado em outros cinco objetivos que são: solicitação de transporte satisfeita, sistema do elevador seguro, sistema do elevador eficiente, sistema do elevador usável e, sistema de elevador barato. A Figura 17, mostra o modelo de objetivos para o Elevador.

Figura 17 – Objetivos para o Elevador

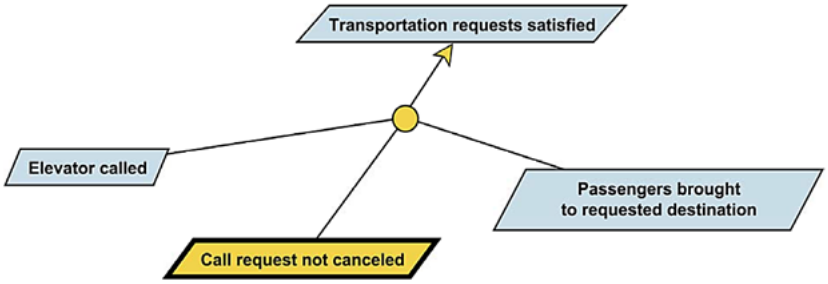


Fonte: baseado em (RESPECT, 2007)

O refinamento do modelo *top-down*, possibilita que todos os objetivos citados sejam satisfeitos. Para exemplificar o refinamento dos objetivos, iremos refinar somente o objetivo

"Transportation request satisfied" (Solicitação de transporte satisfeita). A Figura 18, apresenta o refinamento deste objetivo, mostrando as propriedades de domínio relevantes para a completude dos objetivos, e que são usadas para o design da automação do elevador.

Figura 18 – Objetivo solicitação de transporte



Fonte: baseado em (RESPECT, 2007)

Caso o objetivo *"Call request not canceled"* (Ligação não cancelada) da Figura 18, não for satisfeito, então o objetivo superior *"Transportation request satisfied"* consequentemente não será satisfeito. Refinando o objetivo *"Elevator called"* (Elevador ligado), existe uma interface do sistema com seus respetivos agentes através da botoeira do elevador. Os paralelogramos de bordas grosas e fundo amarelo representam as expectativas vinculados a seus respectivos agentes. Por exemplo, o agente *"Passenger"* (Passageiro) é responsável pela expectativa *"Button depressed"* (Botão acionado), ou seja, pelo acionamento da botoeira. A Figura 19, mostra exatamente isso.

Na Figura 20, se apresenta o refinamento da expectativa *"Button depressed"*, que é refinado nas expectativas. *"Up or down floor button pushed"* (Botão subir ou descer da porta pressionado) e *"Destination floor Button pushed inside cage"* (Botão andar de destino pressionado), ambos vinculados ao agente *"Passanger"* que é o responsável pelas mencionadas ações.

2. Modelo de Responsabilidades:

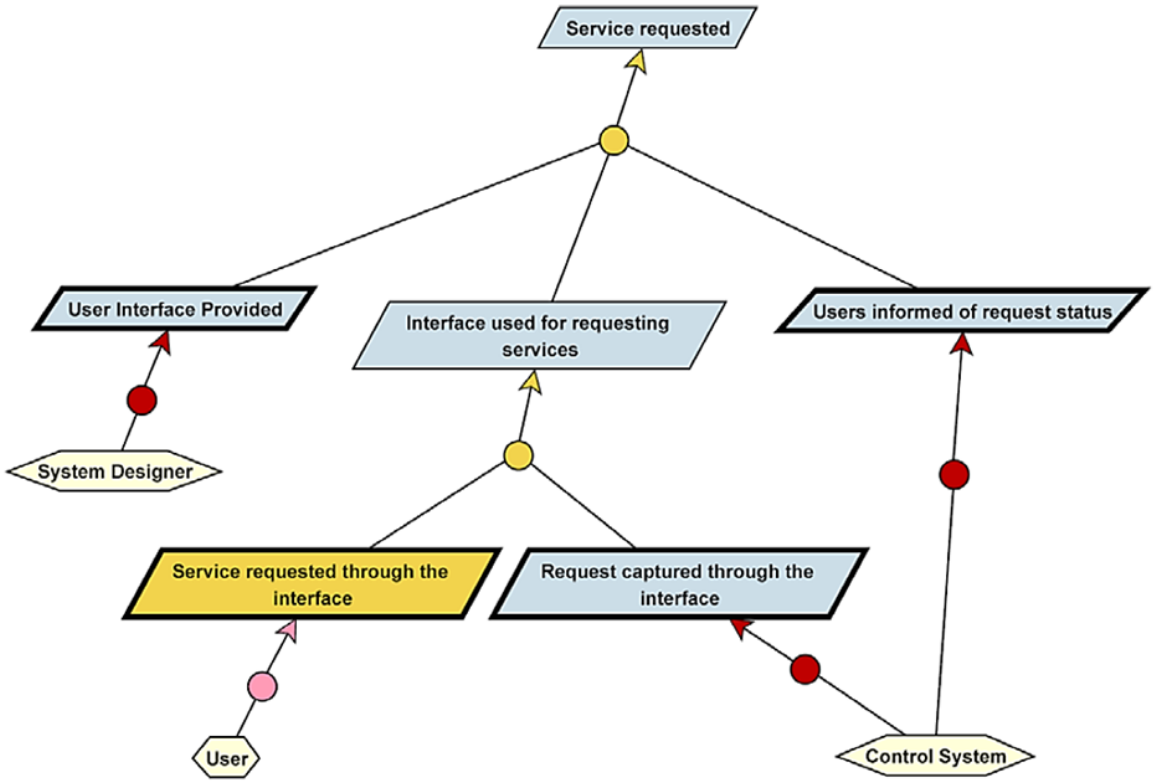
Para construir o diagrama de responsabilidade, analisa-se tanto os requisitos quanto as expectativas no modelo de objetivos e se atribui um agente a cada um deles. Por exemplo, o diagrama da figura 21, *"System protected against fire"* mostra os agentes responsáveis pelos requisitos de reação ao uso de elevadores:

Assim, o controlador do elevador é responsável por abrir as portas quando um alarme de incêndio é acionado e parar todos os elevadores que estão em funcionamento assim que chegarem ao próximo andar;

A empresa de montagem do Elevador que instalará a automação do sistema de elevador é responsável por fornecer dispositivos e equipamentos à prova de fogo.

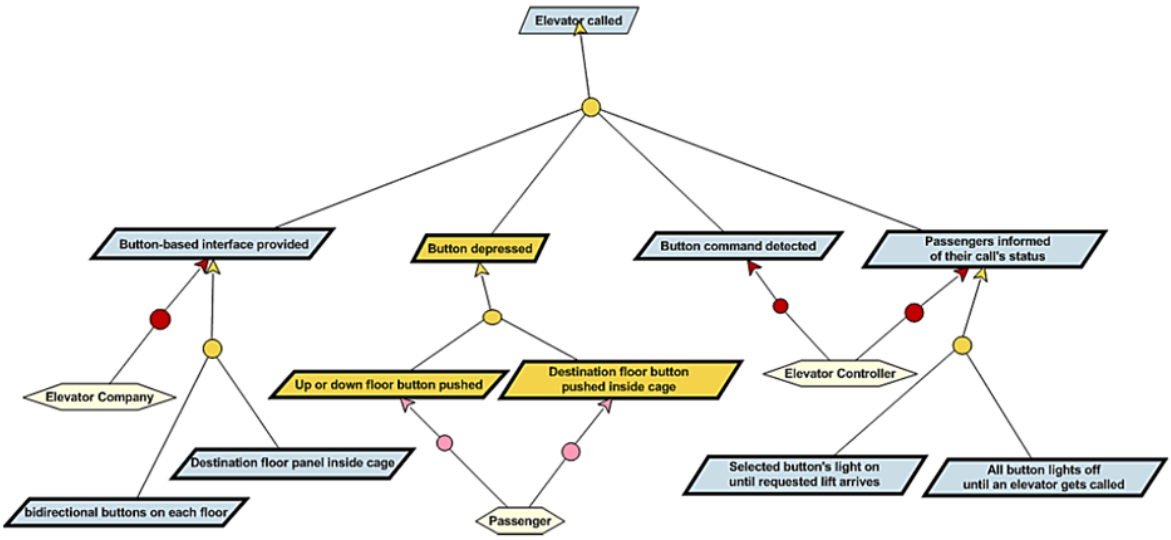
Após da atribuição dos requisitos aos agentes responsáveis, é possível gerar o modelo de responsabilidade por agentes, pois esses podem assumir outras responsabilidades em diferentes

Figura 19 – Objetivo requisição de serviço



Fonte: baseado em (RESPECT, 2007)

Figura 20 – Objetivo chamar ao elevador

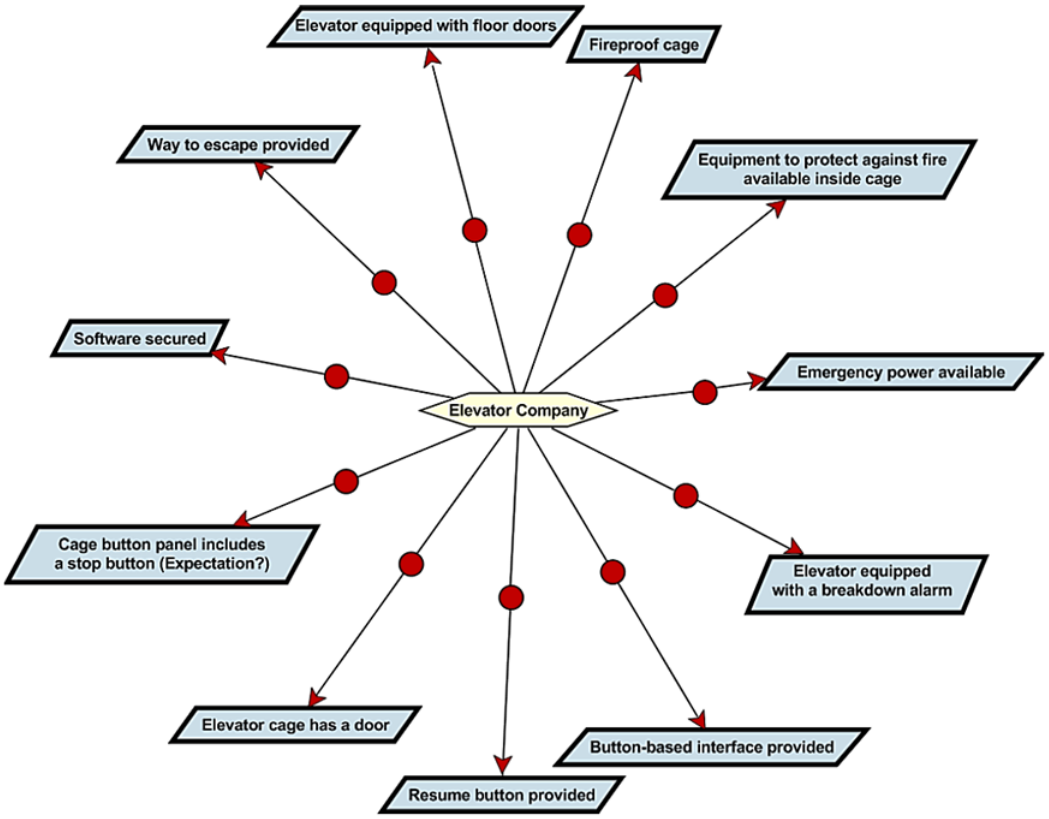


Fonte: baseado em (RESPECT, 2007)

partes do sistema, essa analise permite uma visão geral de todas as suas responsabilidades, conforme mostrado na figura 21.

3. Modelo de Objetos:

Figura 21 – Modelo de responsabilidade



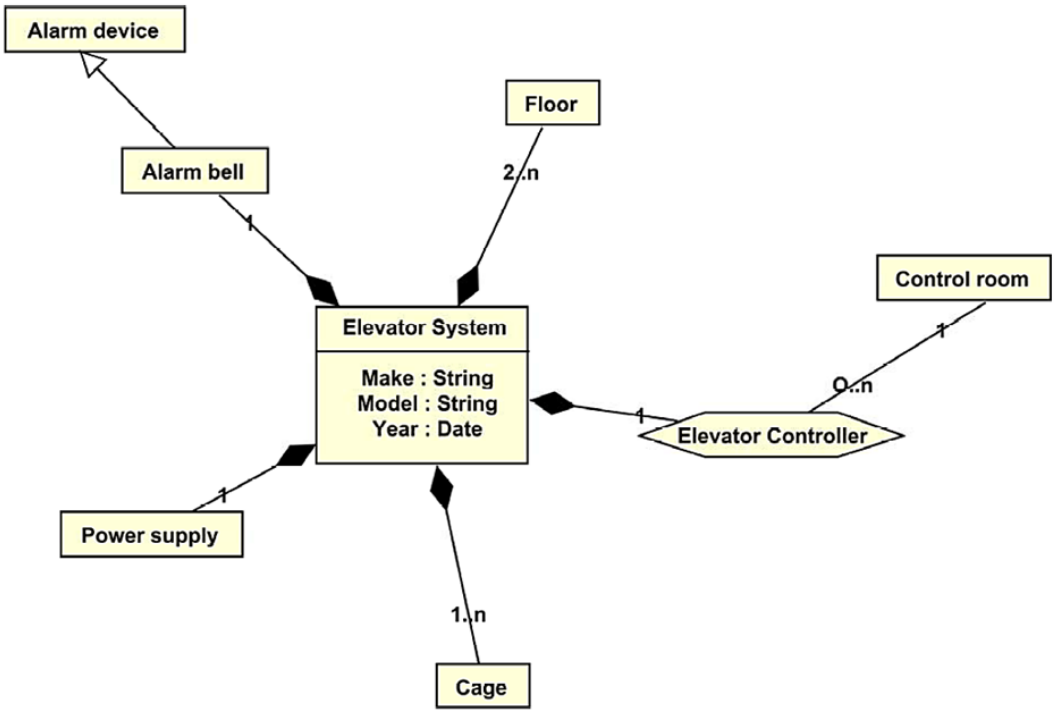
Fonte: baseado em (RESPECT, 2007)

O intuito do modelo de objetos é definir e documentar os conceitos do domínio da aplicação, que são relevantes aos dos requisitos e fornecer restrições estáticas sobre o sistema operacional para satisfação dos requisitos. Parte do modelo de objetos encontra-se relacionado ao domínio das partes interessadas, *stakeholders's*, a outra parte apresenta os requisitos funcionais e requisitos ou restrições do sistema operacional.

A Figura 22 mostra os componentes do Sistema Elevador *"Elevator System"* composto seguintes componentes:

- Uma ou mais *"Cage"* (Gaiolas);
- No mínimo dois *"Floors"* (Andares);
- Uma campainha;
- Um controlador de elevador, situado na sala de controle;
- Fonte de alimentação.

Figura 22 – Modelo de objetos



Fonte: baseado em (RESPECT, 2007)

O modelo de objetos KAOS é compatível com o diagrama de classes UML. As entidades em KAOS correspondem às classes em UML e as associações em KAOS correspondem às associações em UML.

4. Modelo de Operação:

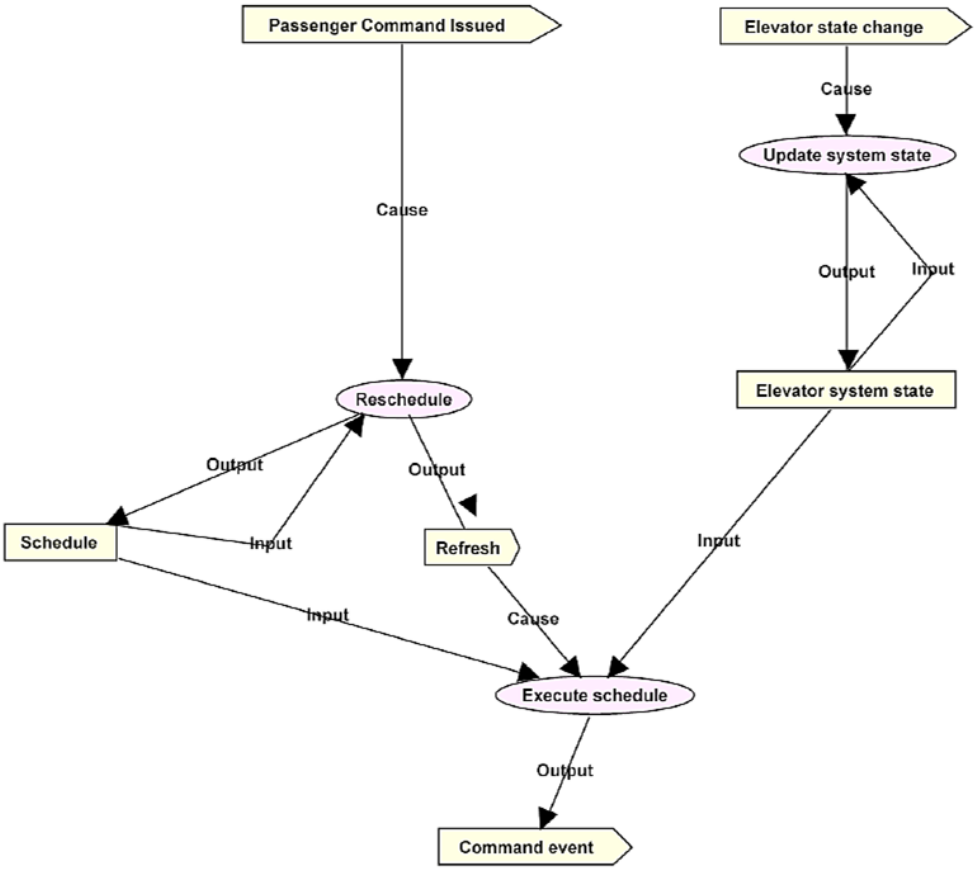
O Modelo de operação KAOS descreve todos os comportamentos dos agentes para cumprir suas exigências. Estes comportamentos são expressos em termos de operações realizadas pelos agentes. No modelo de operações da Figura 23, apresenta-se o evento *"Refresh"* é produzido pela operação *"Reschedule"* que é executada pela operação *"Execute schedule"*.

3.5 As Redes de Petri no Design de Sistemas

A operação de sistemas de automação pode ser conceitualmente descrita por um modelo dinâmico estado-transição. Esta representação pode também ser mapeada em uma estrutura de grafos orientados em que os nós representam os estados e as arestas transições. Nesse contexto, a pesquisa mostra a existência de uma ampla gama de formalismos usados para a modelagem de sistemas de transição (ou modelos estado-transição), tais como: álgebras de processos, Redes de Petri, diagrama de estados, e outros.

A Rede de Petri (RdP) foi criada por Carl Adam Petri, em sua tese de doutorado intitulada Comunicação entre Autômatos, que tinha por objetivo a modelagem de processos

Figura 23 – Modelo de Operações



Fonte: baseado em (RESPECT, 2007)

automatizados com troca de mensagens (SILVA, 2013). A base formal é a teoria de Grafos, e a base conceitual a representação de relações entre conjunto de elementos associados a uma representação estado/transição, característica dos sistemas dinâmicos a eventos discretos (SILVA, 2013).

Portanto, as RdP possuem uma sólida base matemática, o que propicia a análise e a verificação de um número grande de propriedades existentes nos sistemas dinâmicos discretos. A análise de propriedades permite também a verificação dos requisitos destes mesmos sistemas, e a detecção de possíveis erros antes de iniciar a fase de implementação, aderindo às técnicas de design orientadas a modelos (DAVID; ALLA, 2010).

Uma RdP pode ser considerada também um formalismo visual que permite explicar o comportamento físico de um sistema e seus componentes, no que se refere à evolução dos estados. A rede pode ser usada para modelar e verificar sistemas, em diferentes níveis de abstração - usando o conceito de redes hierárquicas. Fazendo uma analogia com outros formalismos, a RdP é mais adequada para ser uma *"running specification"* na medida que o mesmo formalismo pode ser usado tanto para modelagem como para simulação e validação (SILVA, 2013).

A RdP fornece suporte também para a comunicação visual, similar aos diagramas de

fluxos. Uma vantagem das RdP é que esta possibilita o uso de tokens para emular atividades dinâmicas e concorrentes dos sistemas. Além disso, matematicamente, possui uma representação algébrica aderente à representação gráfica, o que permite uma alternância entre métodos formais e a captura de representações semi-formais (DAVID; ALLA, 2010) - como a UML e os diagramas KAOS. Esta característica favorece sua utilização no processo de análise de requisitos.

3.5.1 Definição formal

A definição formal de uma rede de Petri clássica, Place /Tranistion (P/T) (MURATA, 1989) é definida como uma quíntupla $PN = (P,T,F,W,M_0)$, onde:

$P = \{p_1, p_2, ..., p_m\}$ *é um conjunto finito de lugares (elementos passivos);*

$T = \{t_1, t_2, ..., t_n\}$ *é um conjunto finito de transição (elementos ativos);*

$F \subseteq (P \times T) \cup (T \times P)$ *é um conjunto de relacionamentos entre elementos passivos e ativos, chamados de arcos orientados;*

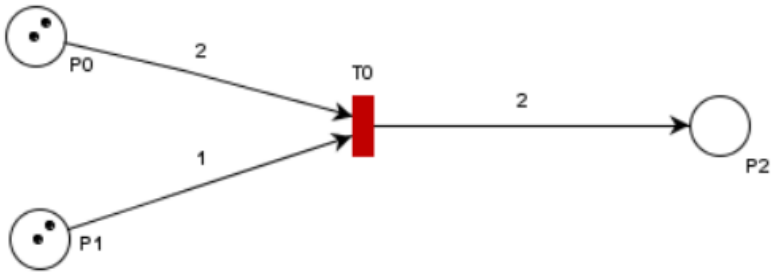
$W : F \rightarrow \{1,2,3,...\}$ *é o peso dos arcos;*
$W(x,y) \rightarrow \mathbb{N}^+ \quad | \quad (x \in P, y \in T) \cup (x \in T, y \in P)$

$M_0 : P \rightarrow \{0,1,2,3,...\}$ *é o vetor de marcação inicial;*

$P \cap T = \emptyset \quad \wedge \quad P \cup T \neq \emptyset$

Na Figura 24, a transição T_0 estará habilitada, se o lugar P_0 tiver pelo menos duas marcas, e o lugar $P1$ tiver pelo menos uma marca. Nesse caso fluirão duas marcas para o lugar P_2.

Figura 24 – RdP transição habilitada



Fonte: Própia

Uma marcação é vetor de dimensão igual ao número de lugares, representado por uma matriz M. Um elemento deste vetor $M(p)$ representa o número de marcas em um determinado lugar p (por exemplo, na Figura 24, $M(P_0) = 2$. É chamada matriz de incidência, à matriz que relaciona lugares e transições que pertencem à mesma localidade.

A dinâmica de uma rede de Petri é representada por uma equação de estados,

$$M_{i+1} = M_0 + \sum_{j=0}^{j=i} A^T \sigma_i$$

onde M_{i+1} é a marcação no instante discreto $i+1$, M_0 é a marcação inicial, A é a matriz de incidência, e σ é o vetor de habilitação, isto é, um vetor indicando quais transições estariam habilitadas em cada estado i. Esta equação pode ser usada para calcular propriedades das redes e consequentemente pode ser aplicada na análise e verificação de requisitos derivados dos diagramas KAOS.

3.5.2 Propriedades das Redes de Petri

As RdP, são analisadas através de suas propriedades (MURATA, 1989) e são classificadas em: dependentes da marcação ou propriedades funcionais, e propriedades estruturais. A análise comportamental de sistemas concorrentes normalmente não admite de um tratamento analítico, pelo que, se faz necessário o conhecimento de propriedades nestes sistemas.

3.5.2.1 Propriedades funcionais

São consideradas propriedades funcionais as que dependem de marcação inicial (MURATA, 1989). Algumas destas propriedades são descritas a continuação:

- Alcançabilidade:

  Pode ser definida como: A marcação Mn é alcançável se existe uma sequência de disparos que leva de M_0 a Mn. A alcançabilidade é uma propriedade fundamental para a análise dinâmica de qualquer sistema. A alcançabilidade objetiva determinar se uma marcação Mn pertence ou não ao conjunto de todas as marcações que podem ser alcançadas desde a marcação inicial $R(M_0)$. Portanto, para que uma marcação M_d seja atingível a partir de M_0 é necessário que exista uma solução inteira não negativa para a equação: $A^T x = M_d - M_0$.

- Limitação

  Uma RdP é limitada ou k-limitada se o número de marcas em cada lugar não excede um número finito k para qualquer marcação alcançável a partir de M_0, ou seja, que $M(p) \leq k$ para todo lugar p e para toda marcação $Mn \in R(M_0)$. Assim, se a RdP é 1-limitada, então se diz que a rede é segura (*safe*).

  Para aplicações que lidam com recursos, os lugares na RdP são usados para representar buffers e registros que armazenam dados intermediários. Nesse caso a exigência de uma rede limitada garante a não existência de *overflows* nos *buffers* ou registros, qualquer que seja a sequência de disparos executada. Se existir a possibilidade de *overflow* as transições que antecedem no lugar são bloqueadas. Para detectar a tendência de *overflow*

em processos, é importante monitorar a marcação e liberar a capacidade dos lugares, utilizando uma condição de disparo mais branda, sem a verificação das pós-condições.

- Vivacidade

  A vivacidade está relacionada à ausência de bloqueios (*deadloks*). Assim uma RdP é dita viva, se para toda marcação atingível a partir de M_0 for possível disparar pelo menos uma transição. A vivacidade é considerada uma propriedade desejável para qualquer sistema, porém, a verificação direta do seu reverso, o deadlock, pode ser combinatoriamente explosiva. Existem entretanto alguns algoritmos dedicados à verificação de *deadlock* que lidam com condições necessárias e que têm desempenho computacional bastante aceitável (SALMON; SILVA, 2012).

- Reversibilidade

  Uma RdP é considerada reversível, se para cada marcação M em $R(M_0)$, M_0 é alcançável desde M. Ou seja, uma rede é reversível, sempre é seja possível voltar ao estado inicial.

- Persistência

  Uma RdP é persistente, se para qualquer par de transições, o disparo de uma delas não desabilita a outra. Toda rede considerada persistente, toda vez que uma transição se habilita, permanece habilitada até ser disparada. A persistência é uma propriedade que está relacionada com as redes livres de conflitos.

3.5.2.2 Propiedades estruturais

As propriedades estruturais não dependem da marcação inicial, mas sim da estrutura da rede (MURATA, 1989). São caracterizadas em termos da matriz de incidência, e associadas a um sistema de equações homogêneo.

- Limitação Estrutural

  Uma RdP é limitada estruturalmente se é limitada para qualquer marcação inicial finita M_0. Isto é: A RdP é estruturalmente limitada se existe um m-vetor x de inteiros positivos tal que: $A^T x \leq 0$, sendo A é a matriz de incidência (MURATA, 1989).

- Conservatividade

  Uma RdP é conservativa se existe um vetor de ponderação de marcas x, onde $x(p)$ é um número inteiro positivo para todo lugar p, de maneira que para cada marcação o produto escalar do vetor de marcação pelo vetor de ponderação é constante, isto é, $\forall M, M^T x = M_0^T x$.

  Assim, uma RdP é conservativa, se e somente se, existe um m-vetor x de inteiros positivos tal que: $A^T x = 0$, onde A é a matriz de incidência e $x \neq 0$.

- Repetitividade

  Uma RdP é repetitiva se existe uma marcação M_0 e uma sequência de disparos s a partir de M_0, de modo que, cada transição acontece infinitas vezes em s.

  Portanto, uma RdP é repetitiva, se e somente se, existir um n-vetor y de inteiros positivos, tal que: $Ay \geq 0$, onde A é a matriz de incidência e $y \neq 0$.

- Consistência

  Uma RdP é considerada consistente se existe uma marcação M_0, e uma sequência de disparos s de M_0 a Md, tal que cada transição acontece pelo menos uma vez em s.

  Assim, a RdP x é consistente, se e somente se, existe um n-vetor y de inteiros positivos tal que se cumpra que: $Ay = 0$, onde A é a matriz de incidência e $y \neq 0$.

- Vivacidade estrutural

  Uma RdP é viva estruturalmente se existe uma marcação inicial viva para a rede. Não se tem conhecimento de uma caracterização geral de vivacidade estrutural envolvendo outros estados de marcas.

- Invariantes

  Outra propriedades das RdP são as invariantes, que representam os componentes conservativos e repetitivos da rede. Existem dois tipos de invariantes: Invariantes de lugar (representam os componentes conservativos) e Invariantes de transição (representam os componentes repetitivos).

  Através, da análise de invariantes podemos analisar propriedades da RdP de forma análoga à verificação de programas. A análise de invariantes de uma rede clássica (*unfolding*) é transformada em algoritmo baseado em álgebra linear (SALOMON, 2017).

  Neste trabalho usaremos a análise de propriedades em RdP baseado em algoritmos já elaborados e testados, e consolidados em ferramentas disponíveis no *Perti Nests World* (PETRINETS, 2018).

3.6 Síntese do Capítulo

As primeiras seções deste capítulo descrevem conceitualmente as proposta de *design* para domínios flexíveis de sistemas de energia, necessários para a concepção de um método de design formal aplicado ao ciclo de vida de sistemas de energia. Foram apresentadas pesquisadas recentes, iniciativas tecnológicas, modelos de referência consolidados tanto na academia quanto no âmbito de organizações governamentais e provedores de energia.

A nossa proposta para o design de sistemas de microgrids teve seus pressupostos mapeados nestes modelos, e deve possuir as seguintes características:

- Visão sistêmica, ou seja, uma abordagem holística onde a SG é componente do sistema de serviço, e, portanto, deve ser pensada desde a fase inicial como um sistema SoS;
- A abordagem deve ser baseada em objetivos, especialmente na fase de requisitos;
- A abordagem segue a Engenharia Orientada a Serviços, o que significa que o acoplamento adequado entre sistema e usuário faz parte do projeto;
- A abordagem é recursiva e distribuída, permitindo utilizar subsistemas já desenvolvidos ou especificar componentes que deverão ser desenvolvidos posteriormente;
- A formalização deve estar presente desde a fase de requisitos utilizando LTL ou Redes de Petri.

CAPÍTULO

4

MÉTODO PROPOSTO

Neste capitulo, apresentaremos a proposta de um ciclo de requisitos para o projeto de sistemas elétricos, em particular de Smart Grids (SGs). Uma abordagem sistêmica para um ciclo de requisitos é apresentada e instanciada para sustentar especificamente sistemas SG's, com um modelo distribuído, baseado em sistemas de sistemas (SoS), e baseado em serviços (POSTIGO; SILVA; SILVA, 2020). Com isso será possível ter SGs preparados para se integrar a um ecossistema de geração/consumo distribuída e se adequar aos perfis e preferências dos usuários. O ciclo pretende ainda prover mecanismos para lidar com os requisitos emergentes, embora isso ainda dependa da habilidade do designer, auxiliado pelos processos automatizados.

4.1 Abordagem proposta para o Design de sistemas SG

O projeto SG será tratado conceitualmente como um sistema (e como um SoS) e no escopo da Engenharia de Sistemas Baseada em Modelos (MBSE). O International Council on Systems Engineering define MBSE como uma aplicação formal da modelagem para apoiar as atividades de requisitos, projeto, análise, verificação e validação, começando na fase de projeto conceitual e continuando ao longo do desenvolvimento e fases posteriores do ciclo de vida (INCOSE, 2015) (INCOSE, 2020). Esse pressuposto insere a demanda por "requisitos dinâmicos", isto é, requisitos associados a invariantes, que persistem durante toda a fase de projeto. Além disso, o MBSE tem várias estruturas para examinar diferentes visões do sistema corporativo: visão de negócios, sistema, tecnologia, operações e serviço (JAMSHIDI, 2008).

Partimos do pressuposto que o *design* -ideal- busca correção e integridade, o que só pode ser alcançado integralmente com a formalização (CRETU; DUMITRIU, 2014). Portanto, uma abordagem formal deve ser inserida para esclarecer aumentar a eficiência das diferentes fases do projeto:

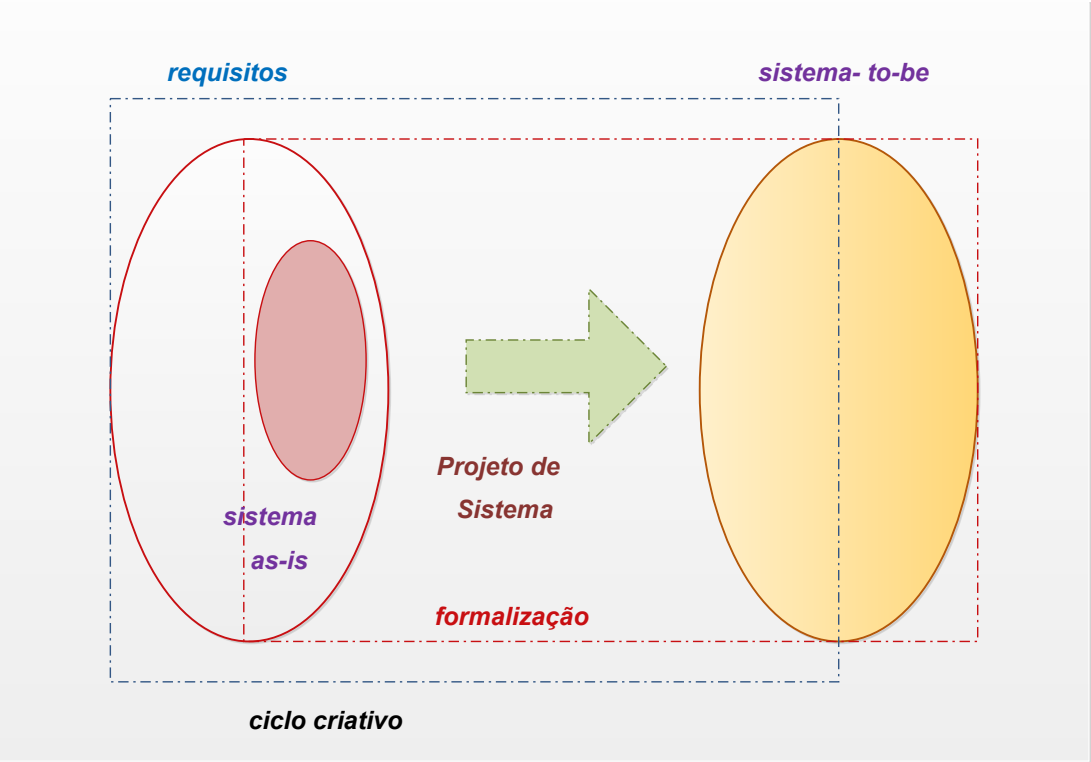
- Requisitos (o que deve ser feito e por quê);

- Design (como atingir os objetivos do sistema forma direta e com menor custo e maior eficiência possível);
- Implementação (como implantar, interagir efetivamente com o usuário final e eliminar o sistema ao finalizar seu ciclo de vida);
- Deployment (como obter feedback do sistema funcionando no ambiente de entorno).
- Manutenção (como fazer uso da documentação e da orientação a serviços para tratar da evolução do sistema).

Certamente a contribuição deste trabalho não vai se estender por todos as fases, mas é importante ressaltar que tem como meta ser aderente e integrada a todas elas, embora trate especificamente da fase de requisitos. A conexão fica mais clara, por exemplo, com a fase de manutenção ou em uma futura evolução do projeto, como mostraremos mais adiante.

Frequentemente, um sistema existente precisa evoluir, ou, um novo sistema é necessário para "substituir"o existente que se tornou obsoleto. Portanto, o processo deve começar modelando o sistema legado, ou o "sistema As-Is"(ou um modelo já existente deve ser reutilizado). Os requisitos do novo sistema começam a ser elaborados demarcando as diferenças entre o "*sistem-as-is*"e o "*sistem- to-be*", ilustrado na Figura 25.

Figura 25 – Ciclo Criativo



Fonte: adaptado (SILVA; NOF, 2015)

Outra característica importante está relacionada à mudança de paradigma, de sistemas orientados a “produto” para sistemas orientados a "serviço".

Estes novos conceitos devem se integrar com os conceitos clássicos de estruturação e refinamentos, utilizando modelagem hierárquica. Essa mudança leva à revisão da linguagem usada para a captura dos requisitos e do processo de modelagem inicial e análise, substituindo a forma clássica, baseada no RUP (*Rational Unified Process*) e em UML, pelo *Goal-oriented Requirements Engineering* e KAOS.

Associando os conceitos de orientação a serviços e de Sistema de Sistemas (SoS), temos, no caso dos SGs, uma rede de microgrids, alguns com capacidade de geração - além de consumo - considerados como sub-sistemas, que se integram para atingir objetivos gerais (*social goals*, tais como o provimento harmônico e sustentável de energia, a a redução de interrupções, otimizando o balanço de carga na distribuição (GOMES; JR.; J.R., 2019). Esse arranjo distribuído tem respaldo em iniciativas de instituições especializadas no design de sistemas elétricos, como mostrado a seguir.

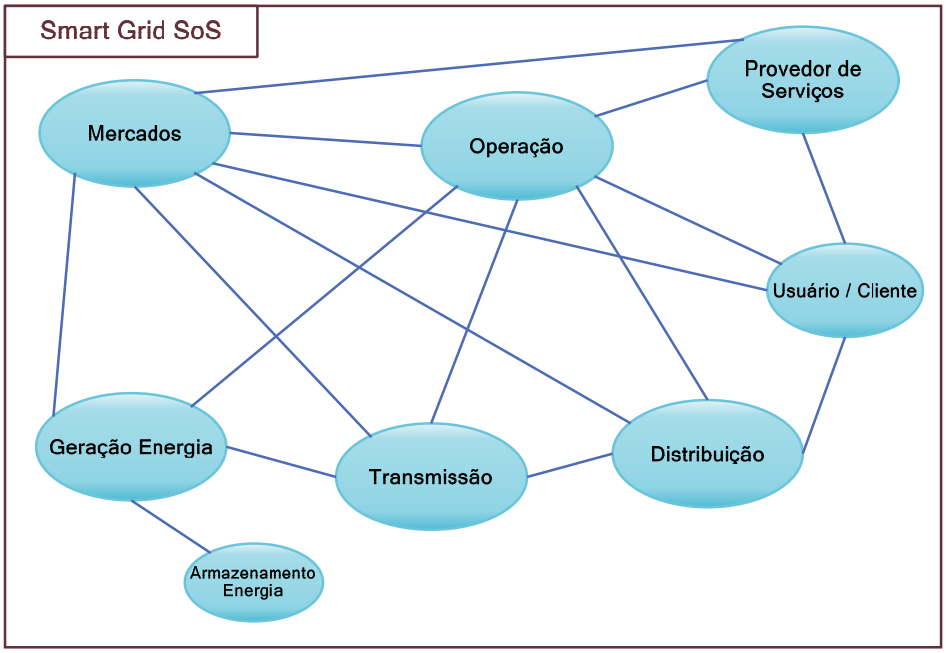
4.1.1 Smart Grid como um Serviço de Serviços (SoS+)

O NIST *Framework e Roadmap para Smart Grid Interoperability Standards*, versão 1.0, foi adaptado para incluir uma abordagem SoS para SG. A figura 26 mostra um esquema conectando diferentes sistemas (e serviços), o que implica em alguma responsabilidade no consumo ou fornecimento de energia. O diagrama também inclui o usuário final - considerado de forma genérica no processo de design de sistemas elétricos. O diagrama é uma representação funcional do domínio relacionando à geração, transmissão, distribuição, a operação, e da responsabilidade implícita do agente. O primeiro indício de que seria interessante ter um "diagrama de responsabilidade"na modelagem de sistemas elétricos.

A análise de sistemas SG como um SoS (LOPES; LEZAMA; PINEDA, 2011), leva a um modelo onde o SG pode ser pensado como um provedor de serviços de energia. Portanto, a modelagem desse tipo de sistema dificilmente é concebível sem planejamento estratégico e modelagem de *workflow* para maximizar o desempenho e otimizar seu funcionamento. Os aspectos tecnológicos, ambientais e socioeconômicos devem ser considerados na mesma perspectiva, capturados como restrição ou interação com o domínio que envolve o sistema. Isso necessariamente aponta para um modelo de requisitos que transcende o paradigma clássico da funcionalidade.

Uma SG - definida como um SoS - pode ser dividida em uma rede de subsistemas que compartilham recursos e possuem objetivos comuns (*social goals*). Nesse contexto, o usuário, também pode ser considerado como um sub-sistema que possui várias funcionalidades (eventualmente realizadas com outros sub-sistemas), como mostrado na Figura 27. Um desses

Figura 26 – Smart Grid SoS



Fonte: Adaptado de (ARNOLD *et al.*, 2010)

sistemas constituintes é o *Smart Meter* (SM)[1].

Projetar um SG como um arranjo distribuído de serviços, ou como um *Service of Services* (SoS+), requer uma abordagem baseada em modelos (MBSE) - como já foi dito anteriormente - e um ciclo de desenvolvimento de requisitos que antecipe formalização. Tal abordagem deve ser mais abrangente do que uma abordagem puramente funcional, o que leva à busca de abordagens alternativas para a Engenharia de Requisitos que relacionam agentes, mapas de responsabilidade e modelem o acoplamento o usuário (ou sistema produtor/consumidor).

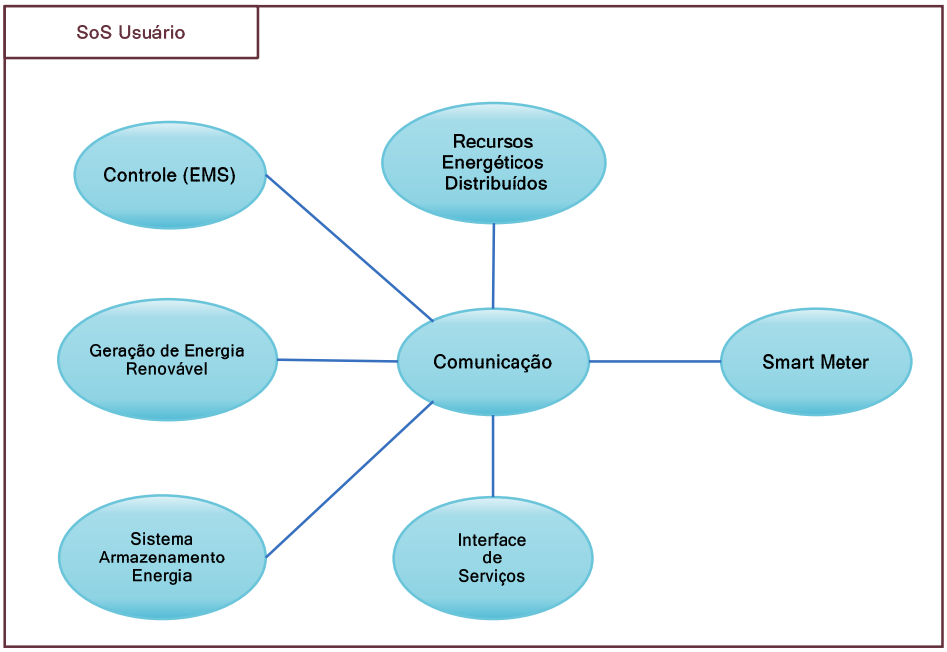
Considerando a importância da fase de requisitos no *Design* da SG, propomos estender a abordagem estritamente funcional para uma Engenharia de Requisitos Orientada a Objetivos (GORE), para embasar um ciclo de requisitos baseado em modelos.

4.2 Ambiente de Design proposto

O ambiente de design proposto aborda o ciclo de requisitos do sistema SG com uma visão sistêmica, portanto seguindo o modelo dos sete samurais do design de sistemas (MARTIN, 2004), o sistema alvo é identificado com P na Figura 28 que tem um domínio de entorno d, inserido no domínio de aplicação D. O ciclo de desenvolvimento dos requisitos começa pela eliciação dos requisitos e das propriedades do entorno d (a parte do domínio onde as interações

[1] Note-se que o usuário representado é na verdade um sub-sistema produtor/consumidor, e não apenas o indivíduo beneficiado com o fornecimento de energia

Figura 27 – Domínio Usuário



Fonte: próprio autor

com o sistema alvo são importantes o suficiente para integrar o modelo). Os requisitos eliciados (eventualmente em linguagem natural) e modelado usando uma linguagem semi-formal (no caso deste trabalho usaremos KAOS, que pertence ao domínio do *Goal Oriented Requirements Engineering*).

A formalização é feita - mesmo em uma fase inicial, onde os requisitos formam um modelo mas ainda incompleto - usando um algorítimo desenvolvido no D-Lab para o sistema ReKPlan (SILVA, 2016a), que transfere automaticamente diagramas de objetivo para redes de Petri. A conversão dos demais diagramas foi feito manualmente neste trabalho.

O modelo até então desenvolvido é analisado e verificado formalmente, especialmente para checar as propriedades listadas no Cap. 3. O modelo em Redes de Petri pode gerar um arquivo XML que pode ser contrastado com propriedades de comunicação das normas, em especial as que geram diagrama XML. No caso deste trabalho usamos mais especificamente a IEC 61850 que pode dar um *feedback* para a modelagem quando o ciclo foi repetido, ampliando R e também o modelo dos requisitos.

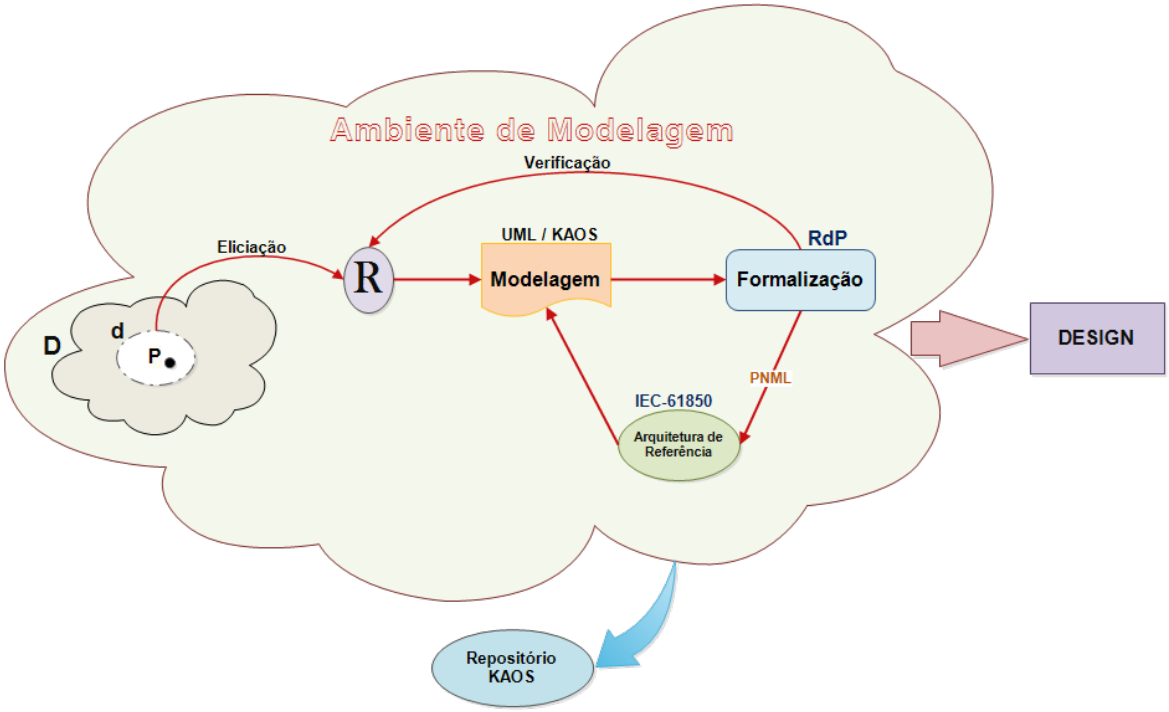
A cada ciclo de desenvolvimento o resultado da especificação de requisitos pode ser enviada para um repositório destinado à reutilização (que não foi explorada neste trabalho) [2]. A

[2] A reutilização de designs não é trivial e estenderia muito o escopo do presente trabalho. Existem trabalhos focados na reutilização baseados em Redes de Petri, como a tese de doutorado de José Reinaldo Silva, Uma formalização do processo de design baseado em metáforas:Sua aplicação na automatização de Sistemas de Eventos Discretos, 1992

geração de um repositório também é recomendada pelo modelo Intelligrid e SGAM, só que feita em UML. Depois de vários ciclos é possível chegar a um ponto em que a evolução do processo não gera nenhum avanço no conjunto dos requisitos e este é o momento apropriado para gerar a documentação que vai para a fase de Design.

O mesmo processo pode ser usado tanto na geração do *sistem-as-is* como do *sistem-to-be*, com a diferença que neste último o *sistem-as-is* e a identificação do que deve ser melhorado são parte do problema P.

Figura 28 – Ambiente de Modelagem



Fonte:próprio autor

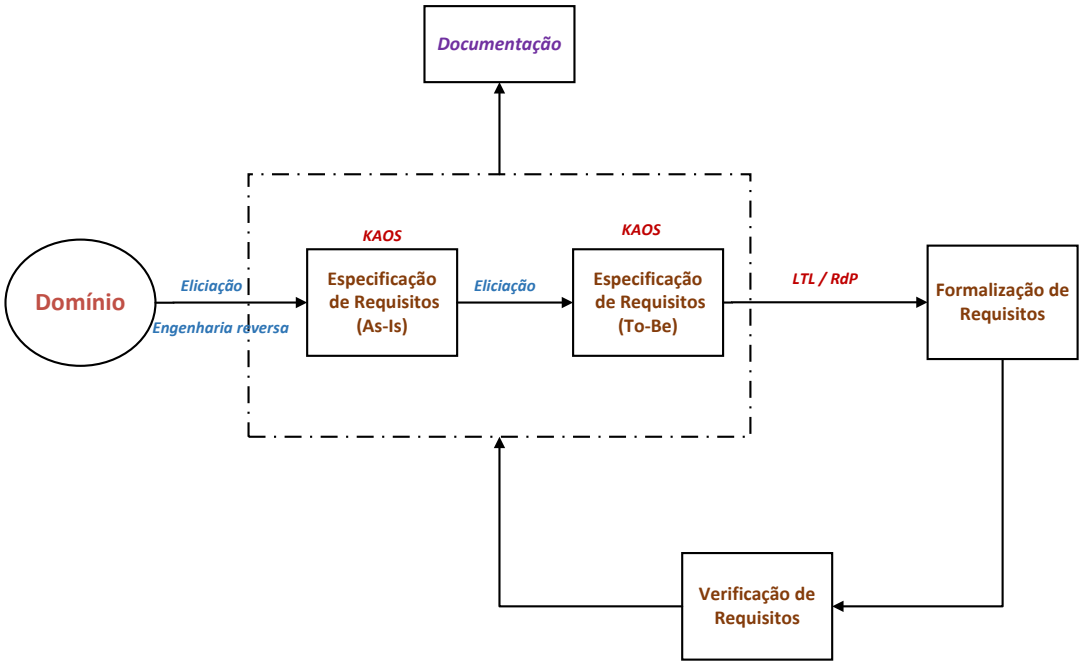
Para o domínio de aplicação D de sistemas elétricos distribuídos integrados a um sistema legado centralizado, o problema P representa uma microgrid específica, o domínio "d" representa a o entorno desta SG, - outros SG's, sistema legado, fontes externas sustentáveis, etc.). O ciclo de requisitos é apresentado com mais detalhes na seção seguinte.

4.3 Ciclo de requisitos para design de sistemas SG

Mais especificamente, ciclo de desenvolvimento começa com a captura de um modelo preliminar requisitos composto pelo problema (o microgrid) e o domínio $d \in D$ que deve ser modelado logo em seguida usando os diagramas KAOS (de objetivos, objeto, operação, e responsabilidade). Utilizamos a plataforma Objetiver. Estes requisitos semiformais serão sintetizados em RdP, (usando o software RekPlan do D-lab para os diagramas de objetivo, e desenvolvendo manualmente a conversão dos demais diagramas. A análise de propriedades e

workflow na RdP será feita usando a plataforma PIPE, disponível do banco de dados do Petri Nets World (PETRINETS, 2018)

Figura 29 – Ciclo de desenvolvimento de requisitos.



Fonte: próprio autor

A Figura 29, ilustra este ciclo de desenvolvimento de requisitos baseado em uma abordagem sistêmica. O processo busca representar e estruturar, o tanto sistema legado (*sistem-as-is*) quanto, o novo sistema (*sistem-to-be*) em um único arranjo, que possibilite modelar formalizar, validar e documentar os requisitos do sistema SG, através de uma estruturação do tipo *top-down refinement*.

Portanto, a modelagem comerá pelo *sistem-as-is*, evoluindo para um *sistem-to-be*, como resultado da otimização e engenharia após alguma eventual negociação. Caso não existir o *sistem-as-is*, então os requisitos serão extraídos através um processo de eliciação com o *Stakeholder*. A verificação dos requisitos da SG, garante a documentação desses requisitos, importante para futuras manutenções do sistema.

A seguir apresentamos mais detalhes sobre cada uma das atividades do ciclo.

4.3.1 Etapa 1: Coleta de dados do Domínio

O passo inicial do processo implica na modelagem (informal) do domínio $d \in D$ - em linguagem natural - e a colocação do problema P, também em linguagem natural (MARTIN, 2004). A especificação do domínio interagente "d"é um trabalho feito pelos designers com o conhecimento da região e das restrições impostas pelo entorno onde o sistema microgrid deve funcionar. Tanto a interação direta como indiretas e relacionadas com o comportamento do usuário final ou possíveis demandas surgidas após a instalação do sistema devem ser consideradas. Já a definição do problema é um trabalho de definição dos objetivos coletivos *social goals* retirados do levantamento de informações do sistema legado, das demandas do usuário, e dos *stakeholders*. Não há como saber se os requisitos eliciados formam um modelo, ou se são consistentes até passar pela etapa de modelagem (semi-formal) seguinte. Esta etapa começa justamente por modelar o *sistem-as-is*.

4.3.2 Etapa 2: Especificação de Requisitos do Sistem-as-is)

Uma vez coletadas as informações do domínio, classificadas e organizadas, modela-se o sistema legado *"sistem-as-is"*, utilizando o método KAOS (RESPECT, 2007).

A identificação dos objetivos é realizada através de uma análise interativa de palavras chaves. O Objetivo desta etapa é obter uma visão geral que possibilite entender e analisar o sistema legado e o domínio onde será aplicado, e compreender como se dá o acoplamento com o usuário e a correspondente *co-criação* de valor (no sistema legado).

O *sistem-as-is* deve ser modelado em KAOS (diretamente ou reutilizando um modelo do repositório, caso exista) e em seguida analisado em redes de Petri. Neste ciclo é incomum detectar algum problema nesta verificação, dado que o *sistem-as-is* existe e já funciona com os requisitos inicialmente especificados. O ciclo se fecha agora com a identificação das mudanças que devem ser inseridas no *sistem-to-be*.

4.3.3 Etapa 3: Especificação de Requisitos Sistem-to-be

Após o entendimento e análise do sistema legado, identifica-se os requisitos futuros que atenderão pertinentes ao *"sistem-to-be"* (as novas demandas dos *stekeholders*, dos usuários final, e do ambiente de entorno). Ou seja, as melhorias que se pretende introduzir no sistema.

É importante salientar que o gerenciamento de requisitos é fundamental no design de qualquer sistema de automação. No caso de sistema potencialmente complexos, como é o caso dos SG's, os processos de elicitação e analise de requisitos tornam-se ainda mais cruciais, e o devido sequenciamento, análise e documentação, mesmo durante os ciclos deve ser observado com o devido rigor (WALDEN *et al.*, 2015) (QUEIROZ; SILVA *et al.*, 2009).

Tanto no caso geral como no caso de serviços (lembrando que estamos tratando os

sistemas SG's como serviço ou pelo menos como produto-serviço) é preciso que em cada ciclo estejamos tratando com um modelo fechado. A utilização de modelos ou arquiteturas de referência (mostrado na Figura 28, é também muito importante, uma vez que possibilitem a aproximação entre os ambientes de desenvolvimento e a aplicação (OLIVEIRA; SILVA, 2015).

Na definição dos novos objetivos a ideia é adequar os futuros requisitos a uma arquitetura de referência consolidada na área de sistemas elétricos, no intuito de projetar uma solução de engenharia otimizada para SG's, e novas restrições ao sistema, surgidas da revisão do domínio d. Usando modelos de referência é possível ter mais confiança de que os requisitos em cada ciclo convergem e que temos um modelo, além de acelerar o desenvolvimento.

Portanto, a arquitetura de referência selecionada influenciará diretamente na definição dos objetivos do *sistem-to-be*. A utilização deste método, visa a identificação dos objetivos, obstáculos, ameaças e conflitos, assim como, a identificação dos riscos envolvidos para posterior gerenciamento. Os modelos gerados serão devidamente armazenados no repositório KAOS (depois do último ciclo) e servirão como *sistem-as-is* em futuras modificações do *Design*.

4.3.4 Etapa 5: Formalização dos Requisitos

Uns risco na etapa de eliciação de requisitos é privilegiar a funcionalidade como alvo central. Os requisitos devem refletir o que o sistema faz, mas nunca "o como fazer" (que seria alvo de etapas posteriores). Portanto, a estratégia adotada neste trabalho consiste em basear a busca pelos requisitos em conceitos que integrem o funcional e o não-funcional, evitando o dilema da escolha ou da prioridade (e o risco de omitir algum requisito não-funcional). De fato, é difícil distinguir quando os requisitos correspondentes a algum modelo parcial do sistema foram plenamente eliciados. Se além disso for acrescentada a preocupação com a possível falta de requisito não-funcionais, o impacto no processo de desenvolvimento seria ainda maior. Por isso a preocupação com a fase inicial, pois a primeira visão sobre o sistema normalmente se propaga para todo o processo de desenvolvimento (FRAPPIER; HABRIAS, 2012). A proposta é então usar métodos baseados em objetivos para a modelagem semi-formal e formalizar os modelos parciais em ciclos.

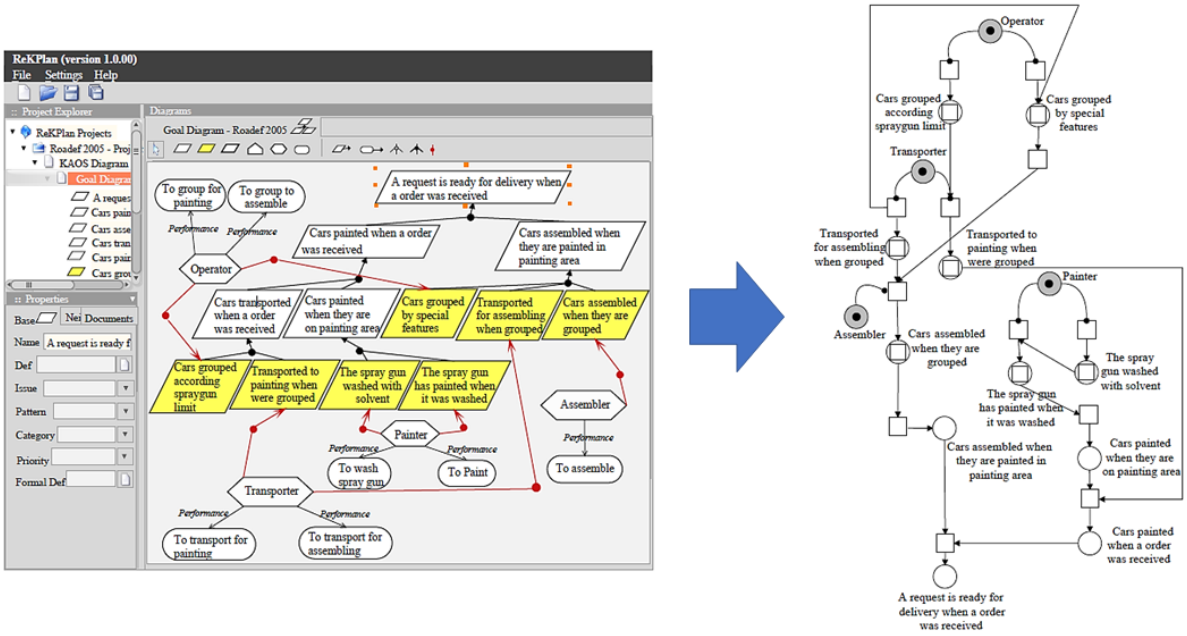
A utilização de métodos formais de representação é maneira efetiva para garantir qualidade na fase de especificação dos requisitos, introduzindo unicidade e precisão no processo (BAIER; KATOEN, 2008). Os métodos formais representam um grande potencial para a representação de estados com conhecimento incompleto (modelos abstratos), detectando ambiguidades ou inconsistências, mesmo na face inicial do *Design*.

Uma proposta para a formalização dos requisitos é utilização da UML como base semi-formal para uma posterior conversão dos requisitos em uma representação formal, baseado em RdP (BARESI; PEZZÈ, 2001), usando algoritmos de transferência baseados em semântica (FOYO; SALMON; SILVA, 2011).

Uma proposta alternativa para formalizar os requisitos consiste em utilizar como base as técnicas GORE, que têm como objetivo aumentar o volume de informação especialmente nesta primeira fase, retirando muito os riscos de uma modelagem exclusivamente funcional. O método KAOS gera um formalismo baseado em LTL (*Linear Temporal Logic*), bastante usado em projetos de desenvolvimento de software (LAMSWEERDE, 2009). Entretanto, a LTL possui algumas limitações, por exemplo, ela não pode representar sistemas distribuídos de forma compacta: como uma representação de processos e LTL seria equivalente a uma representação por autômatos finitos, sistemas distribuídos seriam representados por produtos de autômatos. Esta opção traria problemas para os designers, especialmente os que não pertencem à área de computação, e também para o armazenamento do repositório de casos e sua reutilização.

Neste trabalho propomos a utilização das Redes de Petri (RdP) como representação formal. Assim, partimos de um modelo KAOS com o modelo dos requisitos capturados e, após uma análise prévia usando esta representação, traduzimos o modelo para uma RdP por meio de um algoritmo de transferência desenvolvido no D-Lab (SILVA; SILVA, 2015) denominado RekPlan conforme mostrado na Figura 30.

Figura 30 – Algoritmo de transferência ReKPlan



Fonte: Adaptado de (SILVA; SILVA, 2015)

Através do uso das RdP como ferramenta gráfica e formal, é possível representar atividades dinâmicas e concorrentes dos sistemas (SILVA, 2013), oferecendo um grande potencial para a comunicação entre os diferentes componentes (lembrando que a RdP foi criada justamente para isso), agora usando também a referência da norma IEC 61850. Especificamente para o diagrama de operações, serão definidos os estados cada objetivo atingirá com a execução das operações, dando suporte formal para a verificação dos processos (sequências de ações são disparadas).

Note-se ainda que as redes de Petri começaram a ser usadas na modelagem e análise de requisitos de sistemas, por ser uma ferramenta adequada para reduzir o ciclo de manutenção e evolução, verificando e validando os novos requisitos a serem acrescentados a um sistema já existente, pois permite obter uma representação global dos sistemas associando a estrutura à dinâmica das ações. (FOYO; SALMON; SILVA, 2011). Por isso a escolha desta representação para a verificação formal dos modelos parciais (e final) do ciclo de requisitos.

4.3.5 Etapa 6: Verificação formal dos Requisitos

O Objetivo da verificação formal é confirmar se o modelo parcial é consistente e completo (SOBEIH; VISWANATHAN; HOU, 2004). Com a análise da dinâmica do processo, pretende-se identificar problemas relacionados a possíveis estados incompletos, contraditórios, ambiguidades, inconsistências e condições indesejadas do sistema (como a existência de gargalos e *deadlocks*) em cada ciclo do desenvolvimento dos requisitos. Como este processo pode ser tedioso, pressupomos uma automação parcial deste processo usando ferramentas de suporte. No caso deste trabalho usamos o sistema PIPE versão 4.3 disponível no Petri Nets World (PETRINETS, 2018).

A verificação formal de sistemas pode seguir duas linhas distintas: a prova de teoremas e a verificação de modelos. A prova de teoremas utiliza uma técnica formal para confirmar se o modelo é fechado e completo e se valida propriedades representadas pelo paradigma estado-transição (SOBEIH; VISWANATHAN; HOU, 2004).

Para este tipo de verificação é necessário fornecer o espaço de estados de um sistema e uma propriedade a ser verificada. Para provar a validade desta propriedade é preciso, em princípio, todos os caminhos do espaço de estados (FERRAREZI, 2014) mostrando que a propriedade se verifica em um ou alguns (possibilidade $\Diamond$) deles ou em todos (necessidade $\Box$). Segundo Baier (BAIER; KATOEN, 2008), a verificação de um sistema é utilizada para provar se esse sistema possui ou não determinadas propriedades. A prova de "teorema"(ligados à validade das propriedades) seria a maneira mais indicada para o design de sistemas automatizados, especialmente os que possuem um tempo ΔT de funcionamento autônomo. É preciso ter certeza sobre o comportamento do sistema durante o tempo de autonomia, o que só pode ser dado pelos métodos formais.

Os modelos baseados em RdP podem ser usados para avaliação qualitativa e quantitativa. Assim uma vez definida a rede, pode verificar-se a existência de conflitos, gargalos, e a presença de *deadlocks*, mas é possível também fazer a análise semântica das propriedades comportamentais e estruturais e avaliações de desempenho. Temos ainda a análise de invariantes, o que pode dar suporte formal para os requisitos dinâmicos, isto é, requisitos que persistem e se mantém durante todo o processo de desenvolvimento. Como este trabalho se restringe à fase de desenvolvimento de requisitos, não entraremos em detalhes sobre estas propriedades, mas justificamos assim a inclusão da análise de invariantes no ciclo de desenvolvimento de requisitos.

Neste trabalho, propomos utilizar à análise das propriedades estruturais e comportamentais e avaliação de desempenho da RdP. Serão analisados os invariantes de lugar, que é uma das propriedades que representam o comportamentos dos componentes conservativos e repetitivos da rede, e permitem a identificação de conjuntos de lugares que preservam de forma ponderada o número de marcas.

A verificação das propriedades das RdP deve permitir a análise e verificação da dinâmica do processo de provimento consumo das SGs,e nos permitirá identificar possíveis estados contraditórios e condições não desejáveis, deixando o modelo parcial pronto para um novo ciclo de desenvolvimento dos requisitos.

4.4 Síntese do Capítulo

Neste capítulo foi apresentado o método proposto, que consiste em um ciclo fechado de requisitos Figura 29. Este ciclo é realimentado, ou seja, percebendo que a especificação não está completa podemos voltar à fase inicial de captura de requisitos. O método começa representando o sistema legado (*sistem-as-is*), que, após definir as mudanças ou adequações dará origem a um novo sistema (*sistem-to-be*). Nos casos em que não exista um *sistem-as-is* (um sistema legado documentado ou um sistema reutilizável de onde os requisitos possam ser extraídos), um modelo será derivado da elicitação de requisitos com os usuários finais e *stakeholders*.

O ciclo de análise consiste em fazer a modelagem nos diagramas KAOS (objetivo, objeto operação, e responsabilidade) utilizando uma ferramenta de suporte, como o ObjetivER, e depois transferir para uma linguagem formal em RdP, utilizando a ferramenta RekPan (desenvolvida pelo DLab). As RdP possibilitarão a modelagem e verificação formal dos requisitos fechando cada ciclo.

O método também aborda, a especificação dos requisitos, através da utilização de arquiteturas de referência, o que possibilitará à aproximação como métodos já praticados no design de sistemas elétricos e o tratamento de demandas multidisciplinares, caracterizadas por domínios (de discurso) disjuntos (OLIVEIRA; SILVA, 2015), quando se trata de usuários finais, concessionárias, sistemas de regulação ambiental, autoridades regionais, e outros agentes, distintos da equipe de projeto.

CAPÍTULO

5

ESTUDO DE CASO

Neste capitulo ilustraremos a aplicação do métodos proposto e de potenciais resultados através de um estudo de caso baseado no Projeto P&D "Microgrid com fontes intermitentes para atender áreas isoladas". Este projeto foi desenvolvido e implementado pelo GEPEA (Grupo de Energia do Departamento de Engenharia de Energia e Automação Elétricas da USP) em uma comunidade isolada no interior da Amazônia em 2012 (MARTINEZ; SAIDEL; FADIGAS, 2016). O objetivo principal do projeto era o desenvolvimento e implantação de uma microgrid baseada em um modelo (bateria-solar-diesel). O domínio de implementação foi uma pequena comunidade no município de Laranjal do Jarí, localizado no estado do Amapá. Trata-se de uma realidade semelhante a várias outros pequenos povoados da região, que estão isoladas energeticamente e longe de grandes provedores ou linhas de transmissão. Ademais, estando na região de floresta, tem a demanda de prover energia com sustentabilidade, observando a preservação ambiental.

A escolha deste estudo de caso, justifica-se devido a ter sido um projeto tecnicamente bem desenvolvido e que atendeu com satisfação os objetivos específicos no âmbito do projeto base P&D (MARTINEZ; SAIDEL; FADIGAS, 2016). O intuito do processo de revisão do projeto foi desenvolver novos objetivos formalizados a partir da modelagem e análise de requisitos, seguindo o método proposto, que utiliza GORE.

O projeto - já implementado - serviu como *sistem-as-is* para que pudéssemos investigar possíveis melhorias e/ou manutenção. Para o *sistem-to-be*, a meta era incluir possíveis requisitos emergentes, e, em especial verificar se, sendo o projeto original puramente funcional, analisar a diferença da aplicação de um método orientado a serviços e a objetivo no capítulo anterior. O relacionamento com os usuários deveria estar na base desta diferença.

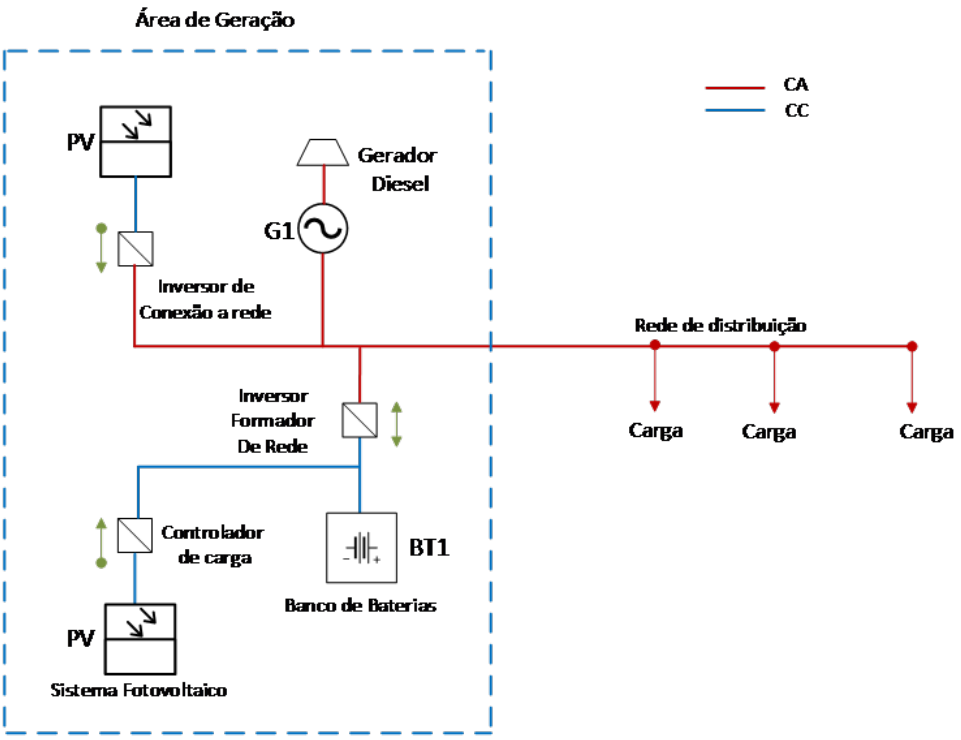
Detectamos alguns requisitos emergentes, que derivaram justamente da mudança no domínio D para um domínio D', que é comum na modelagem de sistemas (MARTIN, 2004). Estas mudanças ocorrem justamente porque o relacionamento entre usuário e sistema muda

assim que este é instalado, especialmente quando se trata de uma inovação que muda a vida destes usuários e o seu padrão de comportamento. Pretendíamos ainda analisar o legado do sistema-as-is para melhorar o nível de automação e otimizar a microgrid, visando o aumento de eficiência, sustentabilidade, qualidade energética e satisfação do usuário. O projeto já tem muitos desses recursos, porém, a satisfação do usuário não é um fator importante em métodos orientados a funcionalidade.

Verificamos que a microgrid utiliza uma arquitetura híbrida (fotovoltaico-diesel-bateria) acoplada em CA e CC, mostrada na Figura 31. Este sistema resulta da combinação das fontes intermitentes solar e bateria, com unidades geradoras a diesel, em um único sistema de geração de energia, que aproveita o recurso solar, operando de maneira combinada com um gerador diesel, para atender uma determinada carga previamente estipulada. A unidade geradora a diesel atua como unidade de reserva (*backup*), para garantir o funcionamento da microgrid durante possíveis períodos de indisponibilidade da fonte solar.

Um Sistema de Aquisição de Dados (SAD) para controle, monitoramento e armazenamento é embutido nos inversores formadores de rede, e utilizado para fornecer informações sobre o comportamento do sistema - como a quantidade de energia gerada e principalmente o consumo das cargas.

Figura 31 – Microgrid Legada - Arquitetura AC-DC



Fonte: próprio autor

A microgrid possui medidores de energia que monitoram o consumo de energia de cada unidade consumidora, e são acionados para bloquear o consumo toda vez que ultrapassarem

o valor pré-estabelecido. O consumo de energia mensal das residências é monitorado pelos medidores de energia mas não é cobrado, pois o proposito não era o faturamento da energia.

5.1 Modelando o sistem-as-is

Seguindo o ciclo de requisitos da Figura 29 usamos o sistema legado para classificar e organizar as informações mostradas na Tabela 3. Essas informações foram usadas para identificar os objetivos.

Tabela 3 – Dados do Projeto

Dados	Descrição
Dados Técnicos	Dados do Projeto; Estudo da demanda; Disponibilidade da energia diária a ser concedida; Disponibilidade da matriz energética; Dimensionamento do sistema foto voltaico; Dimensionamento do Banco de Baterias; Dimensionamento do Sistema de Geração Auxiliar (Gerador Diesel); Dimensionamento das Ligações Elétricas (ANEEL 493); Característica da Rede Elétrica existente; Sistema de Gerenciamento de Energia.
Dados Operacionais	Operação e controle da Microgrid; Monitoramento de variáveis operacionais; Programa de manutenção dos sistemas da microgrid; Sistema de armazenamento de dados e comunicação da microgrid; Procedimentos de Segurança do sistema.
Dados Financeiros	Análise Econômica Financeira do projeto; Planilhas orçamentarias; Aquisição de materiais e mão-de-obra; Custo da Logística de transportes.
Dados legais / Sociais	Licenças Ambientais; Norma Regulamentadoras; Levantamento Socioeconômico; Perspectiva de aumento populacional.
Dados Ambientais	Planilha de levantamento de impactos ambientais; Levantamento de restrições ambientais.

Analisando as informações coletadas, constatamos que, além de pequenas diferenças na implementação o sistema implementado atende aos requisitos originais do projeto, conforme era esperado. No entanto, havia alguns pontos no sistema-as-is que mereceriam ser revisados:

Muitos desses pontos são consequência dos requisitos emergentes, derivados da própria implementação do sistema. Portanto, o intuito do processo de revisão do projeto passou a ser desenvolver novos objetivos formalizados a partir da análise do sistema legado seguindo o ciclo

Tabela 4 – Possíveis novos objetivos para o projeto

Lista de possíveis novos Objetivos
• Consumo excessivo de energia aliado à presença de ligações clandestinas e instalação de cargas não previstas no projeto básico. Aumentando em 240%, aproximadamente, o consumo de energia elétrica; • Aumento no consumo de combustível diesel, além da cota fornecida pelo governo do estado, devido ao incremento na demanda de energia; • Condições anormais de funcionamento dos bancos de bateria (descarregados além do SOC mínimo 50%; • O aumento de consumo de energia, também, provocou sobrecarga nos transformadores e cabos de energia; • Não existe controle geral nas medições referentes consumo de energia; • Não existe penalidades devido ao furto de energia, devido à falta de contratos de energia; • Verificou-se que existe uma cultura do desperdício de energia.

de desenvolvimento mostrado no capítulo anterior, que usa requisitos orientados a objetivos (GORE).

Os modelos de objetivos do system-as-is) é mostrados na Figura 32 baseado na documentação original do projeto. Entretanto, a partir da análise do sistema surgem outras informações que são apresentadas na Tabela 4 e representam expectativas, restrições (obstáculos) ou conflitos.

O *sistem-as-is* foi modelado como primeiro passo do projeto, como previsto no ciclo de desenvolvimento proposto neste trabalho, e seu objetivo principal (*social goal*) foi identificado como "Projetar uma microgrid em baixa tensão na região Amazônica". Estes objetivos foram refinados nos subobjetivos "Levantamento de Dados Técnicos", "Atender a aspectos financeiros", "Atender às Demandas Legais / Sociais", "Atendimento a quesitos Ambientais"e "Atendimento a aspectos Operacionais".

Todos os subobjetivos identificados são refinados de forma recursiva. Assim, por exemplo, "Atendimento as aspectos operacionais"é refinado nos subobjetivos "Controlar a produção de energia", "Medir o consumo"e a expectativa "Manutenção do sistema". Para atingir o subobjetivo “Medir consumo” é necessário “Realizar medição de energia” e “Registrar informações de consumo". As expectativas e requisitos estão vinculadas aos respectivos agentes: a Concessionária - responsável pela manutenção -, o Usuário, o Medidor de Energia, o Controlador do sistema de aquisição de dados (SAD), e o provedor financeiro, o Governo. Porém, no projeto original, o agente identificado como “Usuário” é responsável apenas por “Monitorar medição", ou seja,

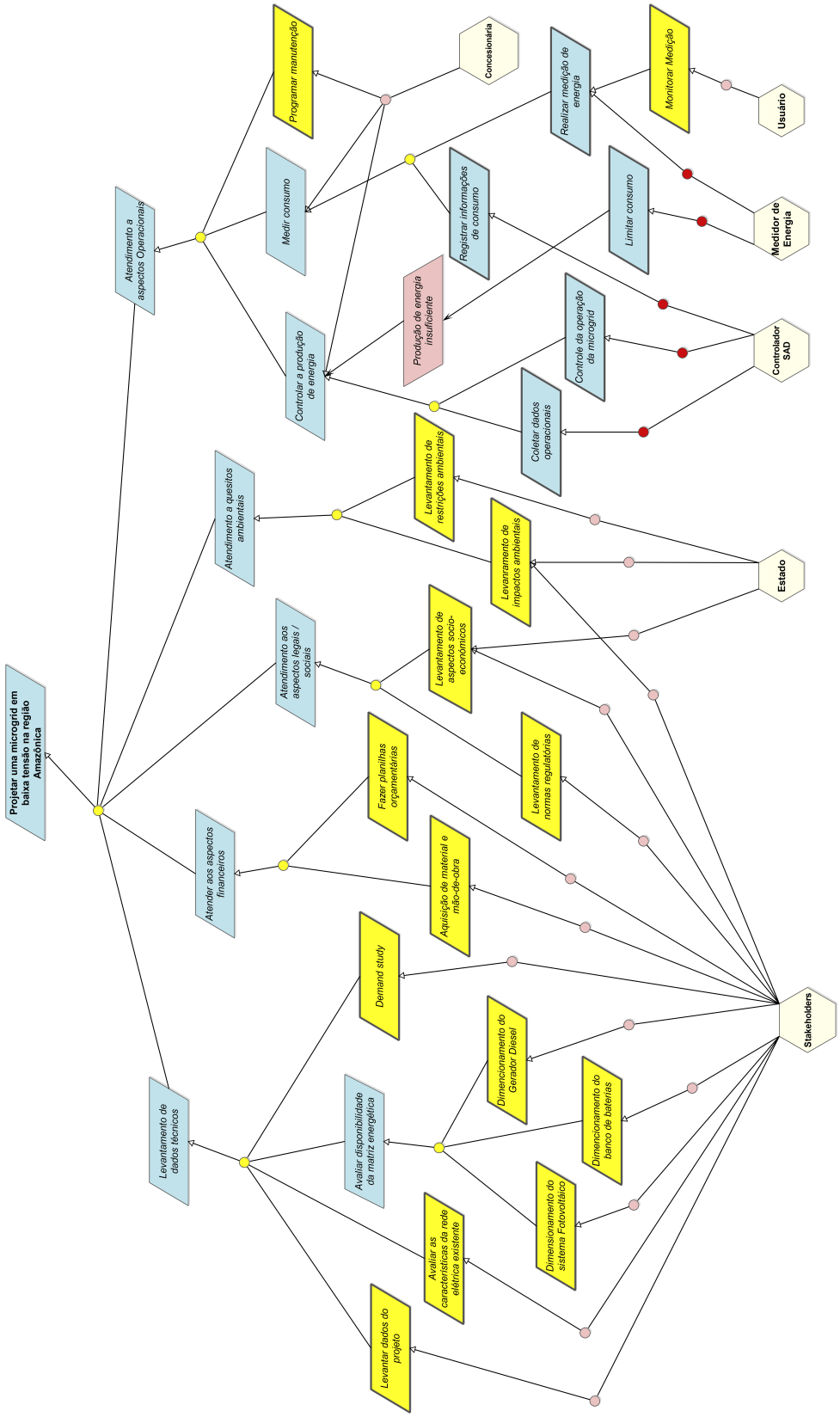


Figura 32 – Sistema-as-is.

este agente não representa de maneira completa o usuário final como receptor de um serviço. Consequentemente, o projeto não identifica o sistema e seus componentes como um serviço.

O modelo de objetivos do *sistem-as-is* mostrou que o do fornecimento de energia é bloqueado sempre que o consumo do usuário ultrapassar um limite pré-estabelecido de 100 kWH. Portanto, não há consideração sobre o histórico de uso ou o consumidor.

5.2 Modelando o sistem-to-be

Para modelar o *sistem-to-be* primeiramente precisamos identificar o problema, e como este se relaciona com o ambiente de execução (o domínio). Assim, após analisar o *sistem-as-is*, verificou-se que o problema principal está relacionado ao consumo excessivo de energia devido as vantagens que oferece eletricidade, aliada ao furto de energia, e ao alheamento do usuário final.

Começamos por identificar o Domínio (ou a mudança em relação ao domínio identificado no *sistem-as-is*), ou seja, o ambiente onde o sistema irá a funcionar, e nesse caso, o contexto é a microgrid legada (*sistem-as-is*), isto é, uma microgrid isolada (*off-grid*) do tipo diesel-solar-bateria, com um nível de automação limitado, onde o usuário não participa do sistema.

Para modelar o novo sistema propomos mudar o domínio, de uma microgrid isolada para uma microgrid distribuída, conectada à rede pública (*on-grid*) e com maior grau de automação, e introduzimos as mudanças pertinentes no domínio. Entretanto, alguns problemas podem surgir neste novo cenário, especialmente relacionados à interface entre a empresa prestadora de serviços e o usuário, pois, neste caso trataremos do "serviço oferecido" e não do produto (Energia Elétrica). Portanto, será necessário priorizar os requisitos do dispositivo que realiza esta função.

Para atingir os objetivos do *sistem-to-be*, propomos a introdução de uma arquitetura de referência no contexto dos SoS (e também SoS+) para suportar um novo arranjo de serviços baseados subsistemas (de serviço) onde os agentes são elementos dinâmicos.

A arquitetura de referência selecionada é a IEC 61850, cuja escolha se justifica por ser um padrão que possibilita a automação da microgrid através de ferramentas orientadas a objetos que permitem a normatização da comunicação, e atende a requisitos de interoperabilidade, extensibilidade, principalmente entre equipamentos de diferentes fabricantes.

Conforme mostrado no capítulo 3 a arquitetura IEC 61850 é composta principalmente por duas partes: um sistema de informação e um sistema de potência. O sistema de potência é a base física para realizar a função da microgrid, composto por unidades de GD, unidade de carga, sistemas de armazenamento de energia (SAE), e unidades auxiliares. A escolha dos GD dependerá das características de cada localidade. Já o sistema de informação da microgrid é a base para o sistema de controle e comunicação, e é dividido em três níveis da piramide de automação.

5.2.1 Modelo de Objetivos

O modelo de objetivos do *sistem-as-is* (Figura 32) é usado como ponto de partida para construir o *sistem-to-be*, colapsando no início objetivos funcionais e não funcionais. Já na elaboração deste modelo será utilizada a arquitetura de referência. Portanto, o objetivo principal do *sistem-to-be* será "Melhorar o design da microgrid legada de baixa tensão na região amazônica". Para cumprir esse objetivo será necessário aumentar o nível de automação da microgrid. Além disso, os requisitos emergentes, previstos na modelagem do *sistem-as-is*, aparecem nas expectativas, obstáculos ou restrições e podem levar a novos requisitos.

A Figura 33, apresenta o refinamento do modelo de objetivos do system-to-be, com o objetivo principal “Melhorar o projeto da microgrid legada em baixa tensão na região amazônica”, refinado nos subobjetivos "Melhorar o planejamento estratégico", "Melhorar a produção/consumo da energia"e "Melhorar o controle operacional da microgrid". Por sua vez, esses subobjetivos podem ser refinados sucessivamente.

Para refinar o subobjetivo "Melhorar o controle operacional da microgrid” seguiremos as orientações da arquitetura de referência IEC 61850. Sendo assim, o subobjetivo principal será refinado nos subobjetivos “Melhorar o sistema de informação da microgrid” e “Melhorar a operação do sistema de força da microgrid”, que visam aumentar o nível de automação da microgrid.

5.2.1.1 Modelando o sistema de Informação

O sistema de informação segundo a IEC 61850 envolve as atividades relacionadas ao gerenciamento da comunicação e da informação. A Figura 34, mostra o subobjetivo "Melhorar o sistema de informação da microgrid"refinado nos subobjetivos "Melhorar o monitoramento e controle da informação”, e "Aumentar a participação do Usuário” que insere o acoplamento com o usuário final - importante para a caracterização do novo sistema como um sistema de serviços (SoS+). O acoplamento com o usuário final é fundamental também para melhorar sua qualidade de vida, dar mais conforto e a segurança sem elevar os custos operacionais. Estes fatores devem portanto melhorar a satisfação do usuário e ajuda a torná-lo mais participativo.

Na modelagem o usuário, agora denominado (prosumer), será inserida a capacidade de produzir sua própria energia a partir de fontes renováveis limpas, reduzindo a dependência da rede comunitária. Portanto, a atuação deste usuário é mais ativa e espera-se um empenho e incentivo na redução do consumo. Neste novo cenário o prosumidor precisará interagir com a microgrid através de uma Interface Homem-Máquina (IHM), verificando, planejando e otimizando seu consumo, ou até mesmo avaliando a qualidade dos serviços oferecidos pela rede comunitária.

Obstáculos são inseridos nos diagramas de objetivos para denotar não-conformidades com os objetivos. Por exemplo, alguns dos obstáculos para o não cumprimento dos objetivos traçados são "Produção insuficiente de energia"e "Medidor inteligente com defeito", ou mau

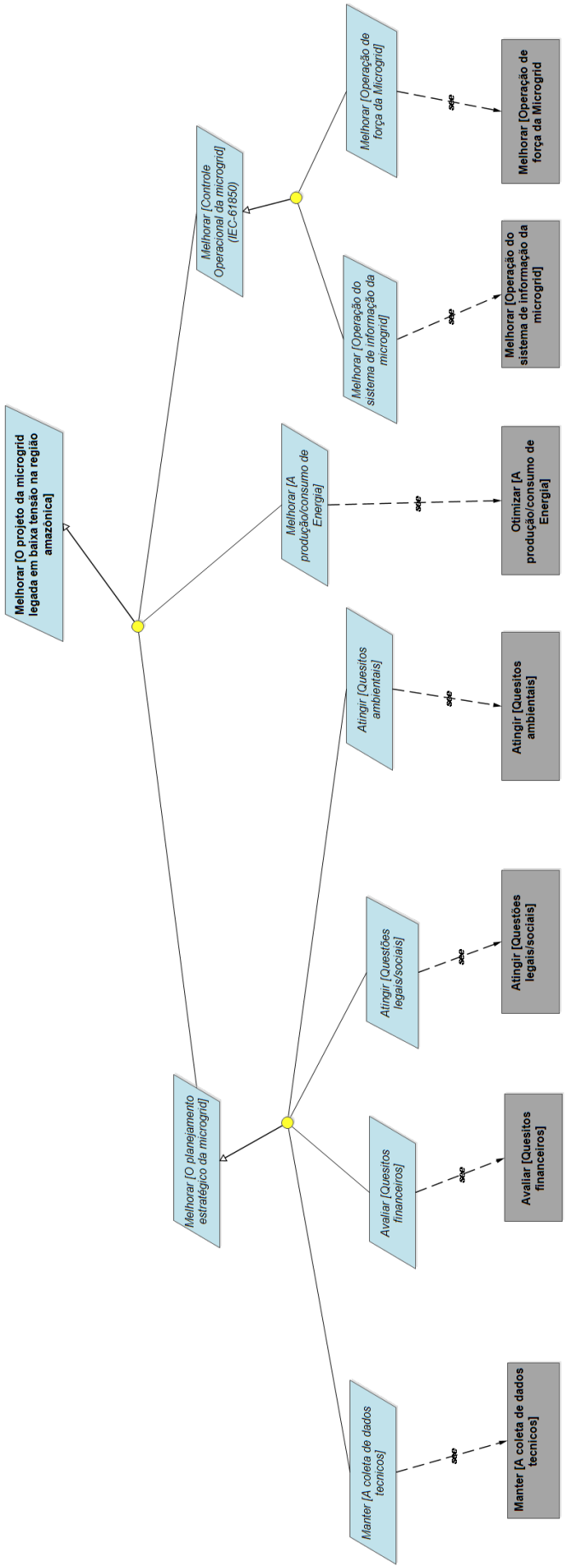


Figura 33 – Sistema-to-be (Principal)

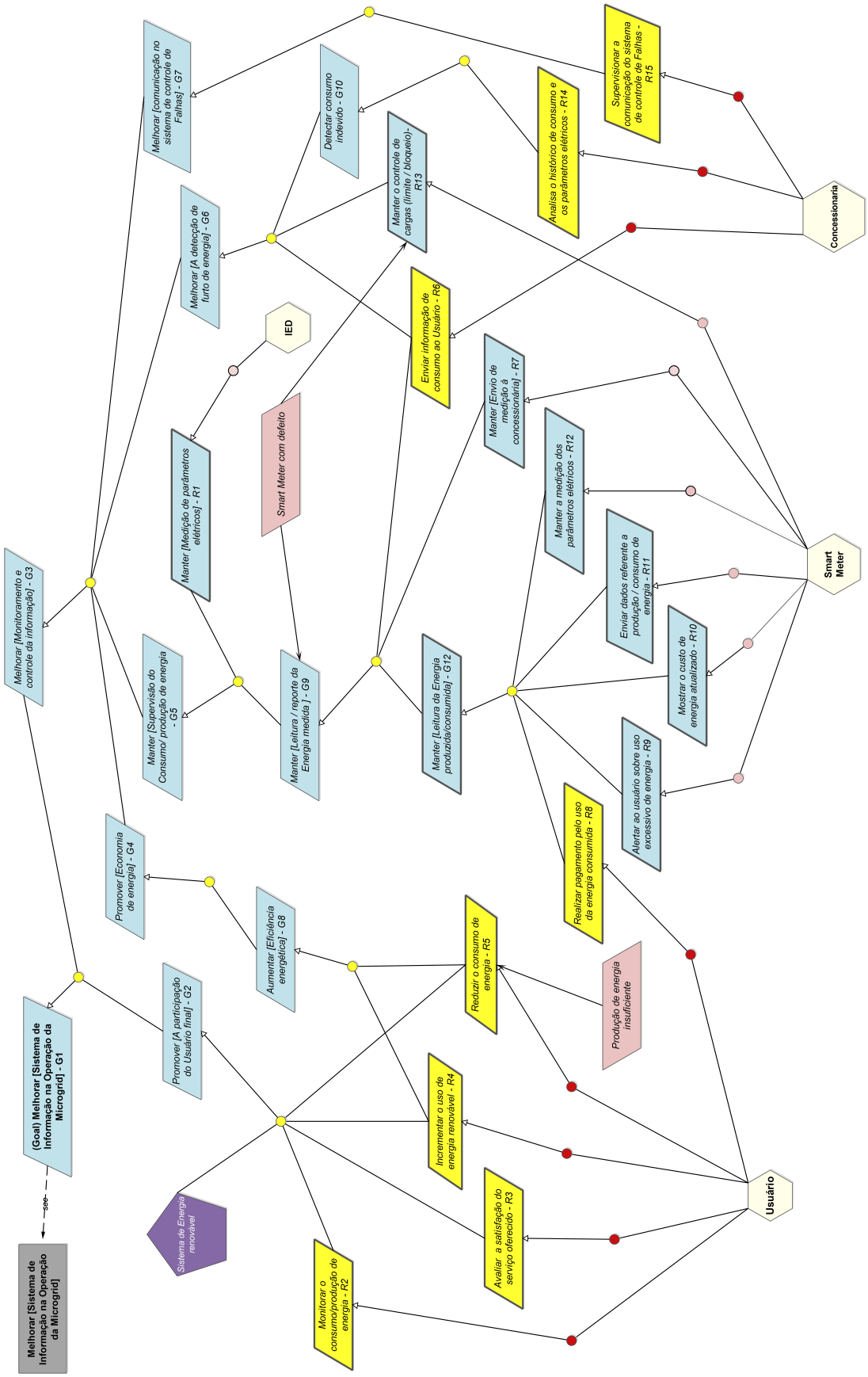


Figura 34 – Refinamento Objetivos Informação

funcionamento. Obstáculos chamam a atenção para as causas do não cumprimento dos objetivos e bloqueio da automação: no nosso exemplo, para garantir a coleta de dados, deve-se verificar o funcionamento do SM.

O objetivo de "Reduzir o consumo de energia"falhará se o usuário não realizar as ações necessárias. Essas ações devem vaiar desde atos de economia de energia até a produção de energia como Prosumidor. Dessa forma, o acoplamento com o usuário final também será monitorado, preenchendo o acoplamento sistema-usuário. O Apêndice-A mostra a formalização destes requisitos em LTL.

Os requisitos emergentes são outro problema de design, derivados das novas perspectivas de uso do sistema, modificando os objetivos. Estes requisitos aparecem devido às mudanças causadas no domínio pelo novo sistema automatizado. Logo, o sistema automatizado deve ser revisado para detectar, em tempo de design, esses requisitos emergentes e não ocasionar um impacto negativo no projeto - especialmente depois da implementação. Cabe ressaltar que as abordagens funcionais não facilitam a detecção de requisitos emergentes, e por isso, justifica-se a necessidade de uma abordagem orientada a objetivos.

Nesse contexto, a Tabela 5 apresenta uma lista de metas emergentes que foram eliciadas.

Tabela 5 – Objetivos Emergentes

Metas do Usuário
• necessidade de melhorar as ações de economia de energia; • incluir geração de energia local e sustentável; • equilibrar os custos para lidar com o uso crescente de eletricidade; • monitorar conexões ilegais para desviar energia ; • diversificar as opções de pagamento para lidar com o crescimento legal do consumo; • eliminar faturas imprecisas ou estimadas.

5.2.1.2 Modelando o sistema de Força

Para modelar o sistema de força da microgrid, deveremos levar em consideração a disponibilidade das RED's, as unidades de carga, as unidades de armazenamento de energia, e as unidades auxiliares. Também, devem ser consideradas as mudanças imprevisíveis das cargas, e a estratégia para a reconfiguração topológica da microgrid.

A IEC 61850 indica como cada RED acessa o barramento da microgrid, ou seja, os modos de operação. A Microgrid se conecta com a rede comunitária pelo disjuntor de conexão

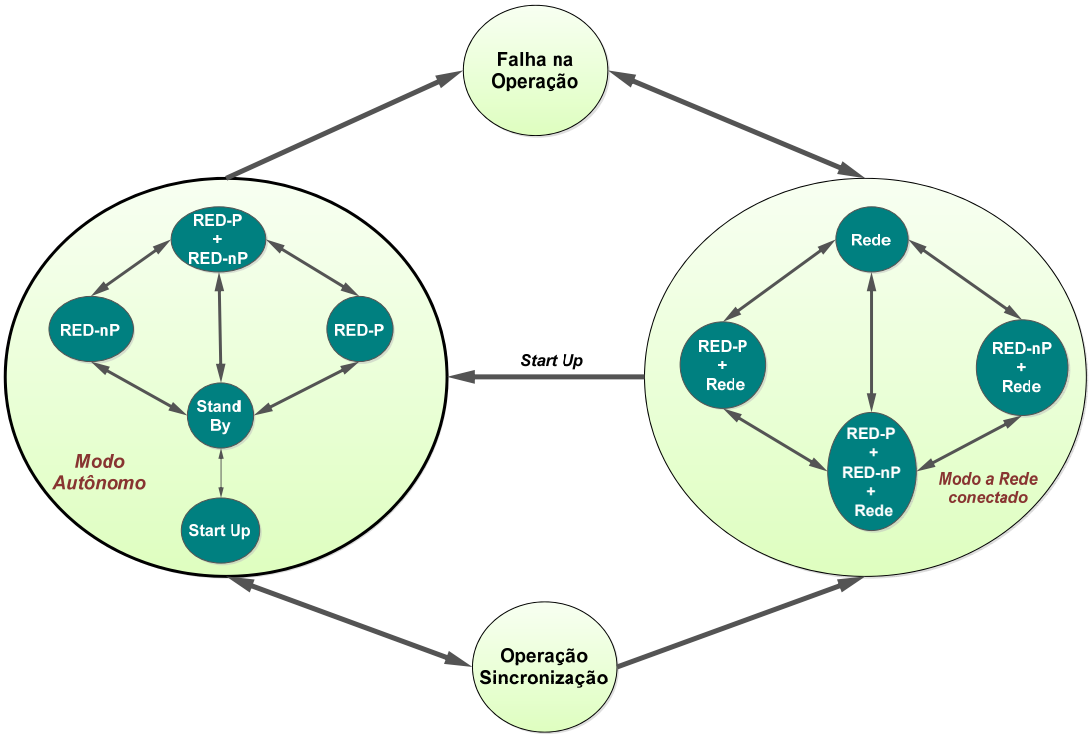
PCC, ou eventualmente pode estar conectada a outra microgrid ou SG. Quando o PCC fecha a microgrid acessa a rede pública e comuta para o modo rede conectada *(on-grid)*. Quando o disjuntor está aberto, a microgrid é separada da rede comunitária, e comuta para o modo autônomo *(off-grid)*.

O modo de operação ou modo de aplicação, regula a estratégia de controle para cada RED, como mostrado na figura 35. Por conseguinte, se a microgrid for conectada à rede de serviços, mas o sistema estiver funcionando de forma independente, então o sistema funcionará no modo de operação ilhada.

Se o sistema perceber ausência de energia na rede comunitária, então o este deverá funcionar no modo de operação isolado *(off-grid)*. Neste modo a microgrid deve fornecer energia suficiente para alimentar a carga total com autos-suprimento ou compartilhamento de energia com outras microgrids.

Portanto, para projetar a automação da microgrid, devem ser considerados os requisitos de cada um desse modos, e a escolha do modo de operação dependerá do estado da rede de energia elétrica e do comportamento da microgrid.

Figura 35 – Operação IEC 61850



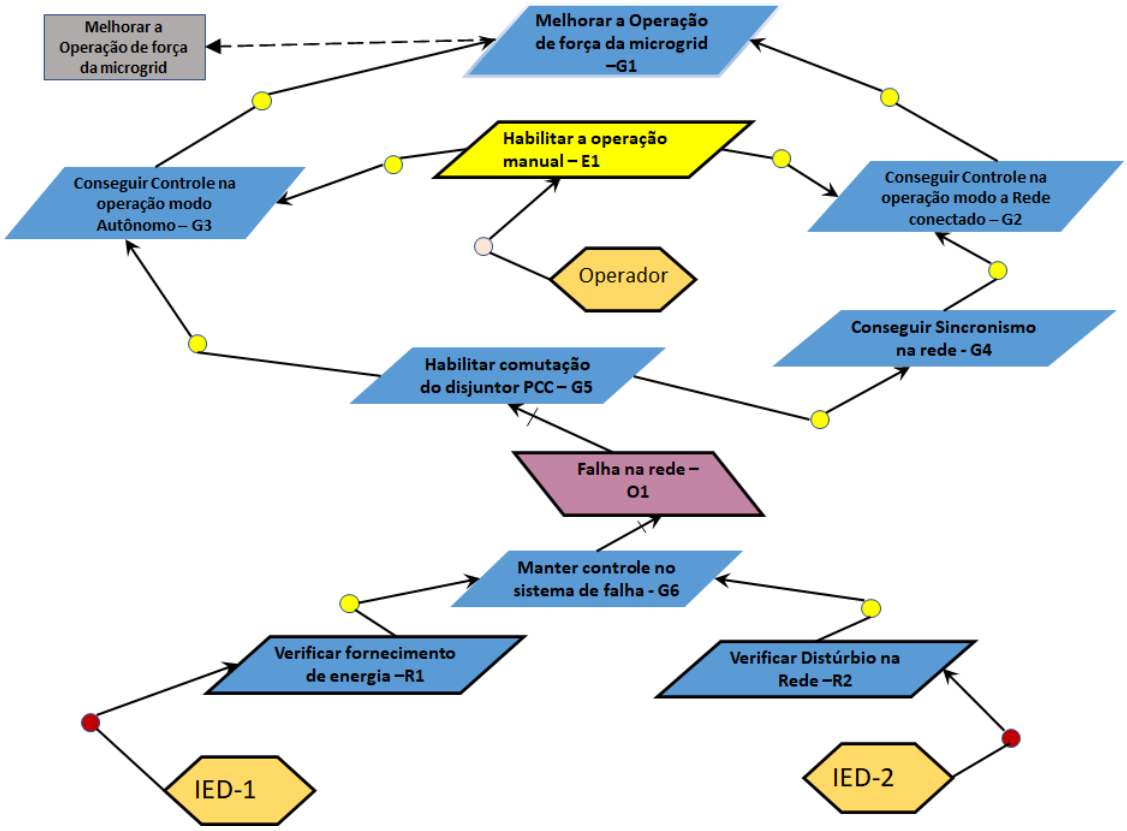
Fonte: próprio autor

A Figura 36, mostra o refinamento do subjetivo “Melhorar a operação do sistema de força da microgrid” refinado nos subobjetivos “Conseguir o controle na operacional no modo

Autônomo” e “Conseguir controle na operação a rede conectado. Entretanto, para que estes subobjetivos sejam satisfeitos é necessário o cumprimento do subobjetivo “Manter o controle do sincronismo da rede”. Este subobjetivo será demandado toda vez que o subobjetivo “Habilitar a comutação disjuntor PCC” tenha sido demandado, mas, o cumprimento deste objetivo passa pela solução (ou ausência) ao obstáculo “Falha na rede”, que acontece quando o subobjetivo “Manter o controle no sistema de falhas” for satisfeito, através do cumprimento dos requisitos “Verificar fornecimento de energia” e do requisito "Verificar distúrbio na rede- que é acionados pelos agentes do contexto “IED-1” e “IED-2” respetivamente, ou pelo agente de ambiente “Operador”.

A Tabela 6, descreve os principais objetivos e subobjetivos relacionados à dinâmica operacional do sistema de força da microgrid baseado na arquitetura de referência.

Figura 36 – Modelo de Objetivos da operação baseado na IEC-61850



Fonte: próprio autor

5.2.2 Modelo de Objetos

O modelo de objetos do system-to-be enfatiza os objetos relacionados à modelagem de objetivos da operação do sistema de força da microgrid, baseado na arquitetura de referência. Ele mostra objetos pertencentes ao domínio da aplicação que suportam os requisitos e as restrições - esses objetos modelam também as interações com o domínio.

Tabela 6 – Requisitos da Operação de força baseado na IEC 61850.

Tipo	Objetivos	Descrição
G-1	Melhorar a operação de força da microgrid	Este é o objetivo principal da operação baseada na IEC-61850.
G-2	Conseguir controle na operação a rede conectada	Este objetivo será alcançado quando a microgrid está conectada à rede principal e aos RED's.
G-3	Conseguir controle na operação no modo autônomo	Este objetivo será atingido sempre que uma falha ou circunstância anormal ocorre na rede principal e a microgrid, alimenta as cargas (pelo menos as cargas mais sensíveis) unicamente com os RED's.
G-4	Conseguir sincronismo na rede	Este objetivo é cumprido quando a microgrid tem apenas energia renovável, modos de transição adicionais são necessários para sincronizar a geração distribuída (GD) com e sem a rede principal. Além disso, quando os programas de resposta à demanda estão ativos, é considerado como outro modo de operação que faz parte do modo de rede conectada.
G-5	Habilitar comutação do disjuntor PCC	Este objetivo acontece cada vez que a chave principal de interconexão (PCC) fecha a microgrid, ele acessa a rede e muda para o modo de rede conectada e, quando a chave do PCC é aberta, a microgrid é separada da rede de serviço e muda para o modo autônomo.
G-6	Manter o controle no sistema de falhas	Este objetivo visa solucionar o Obstaculo "Falha na rede".

A Figura 37, apresenta a troca de dados entre os principais dispositivos. Os IED's recebem os valores medidos da operação e status de informações dos equipamentos da microgrid através do barramento de processo, e atualiza os valores correspondentes medidos e, em seguida, aciona o registro, com base no relatório opções do bloco de controle, para transmitir esta informação para o sistema de controle da microgrid.

Cabe destacar que estes objetos podem ser de diferentes tipos: entidades (sistema de medição, sistema de geração, sistema de armazenamento de dados, sistema de comunicação) - representando objetos independentes e passivos; agentes (Operador, e IED's) representando objetos independentes e ativos; e associações, representando objetos dependentes e passivos.

5.2.3 Modelo de Responsabilidades

Através do modelo de responsabilidade iremos mapear a relação entre os agentes e seu papel na satisfação e restrições de requisitos. Construir o modelo de responsabilidade significa analisar diferentes requisitos e expectativas e atribuir um agente a cada um deles.

A Figura 38, permite mapeamos a relação entre os agentes e sua responsabilidade

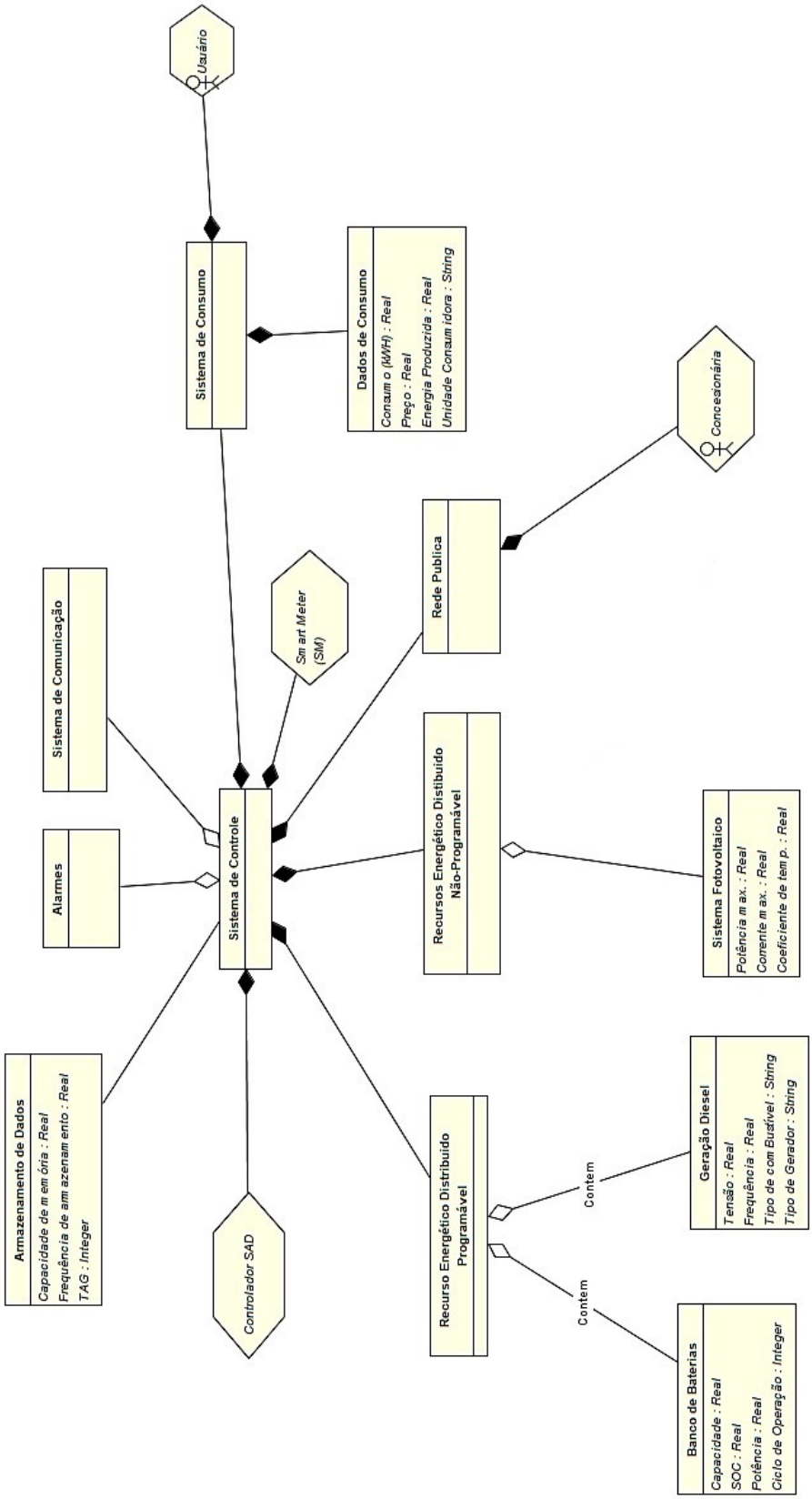
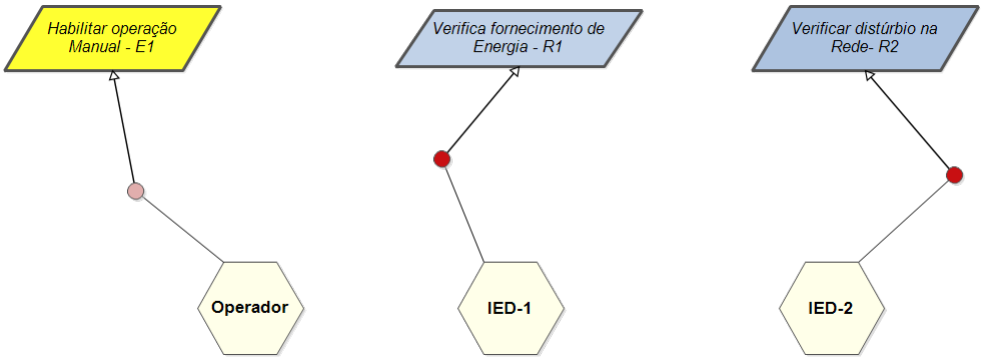


Figura 37 – Modelo de Objetos

Figura 38 – Diagrama de responsabilidades



Fonte: próprio autor

ou demandas para satisfação dos requisitos no intuito de aumentar o nível de automação da microgrid, assim como, descreve os novos requisitos e expectativas atribuídos ao "Operador" e as IED's.

Este diagrama também permite fazer rastreabilidade e calcular o impacto na hora de fazer futuras manutenções do sistema.

5.2.4 Diagrama de Operações

O Diagrama de operações permite modelar os processos correspondentes à dinâmica da operação de força baseado na IEC 61850.Assim como, descreve o comportamento dos agentes para atender aos requisitos. Assim, na figura 39 mostra-se o comportamento dos agentes "IED-1" e "IED-2" encarregados de "verificar o fornecimento" e "verificar se tem distúrbio respectivamente na rede" realizando operações que afetam "na operação do modo de falhas" e provoca a ativação da operação de "Sincronização da rede". Por sua vez, ativa a operação "Modo de rede conectado". Em manutenções posteriores, qualquer alteração na comunicação com o sistema de controle levaria a uma investigação, reprogramação ou ajuste destes agentes.

5.3 Formalização dos Requisitos

Como mencionado anteriormente, o ciclo de projeto exige uma formalização do modelo de requisitos para realizar uma análise adequada visando sua correção e completude. Nesta etapa, a representação semiformal KAOS é transferida para outra representação (formal), neste caso, para RdP, na qual a verificação poderá ser realizada.

A Figura 40 apresenta a modelo formal dos requisitos em RdP, sintetizada a partir do modelo KAOS, utilizando do algoritmo de transferência ReKplan (SILVA; SILVA, 2019).

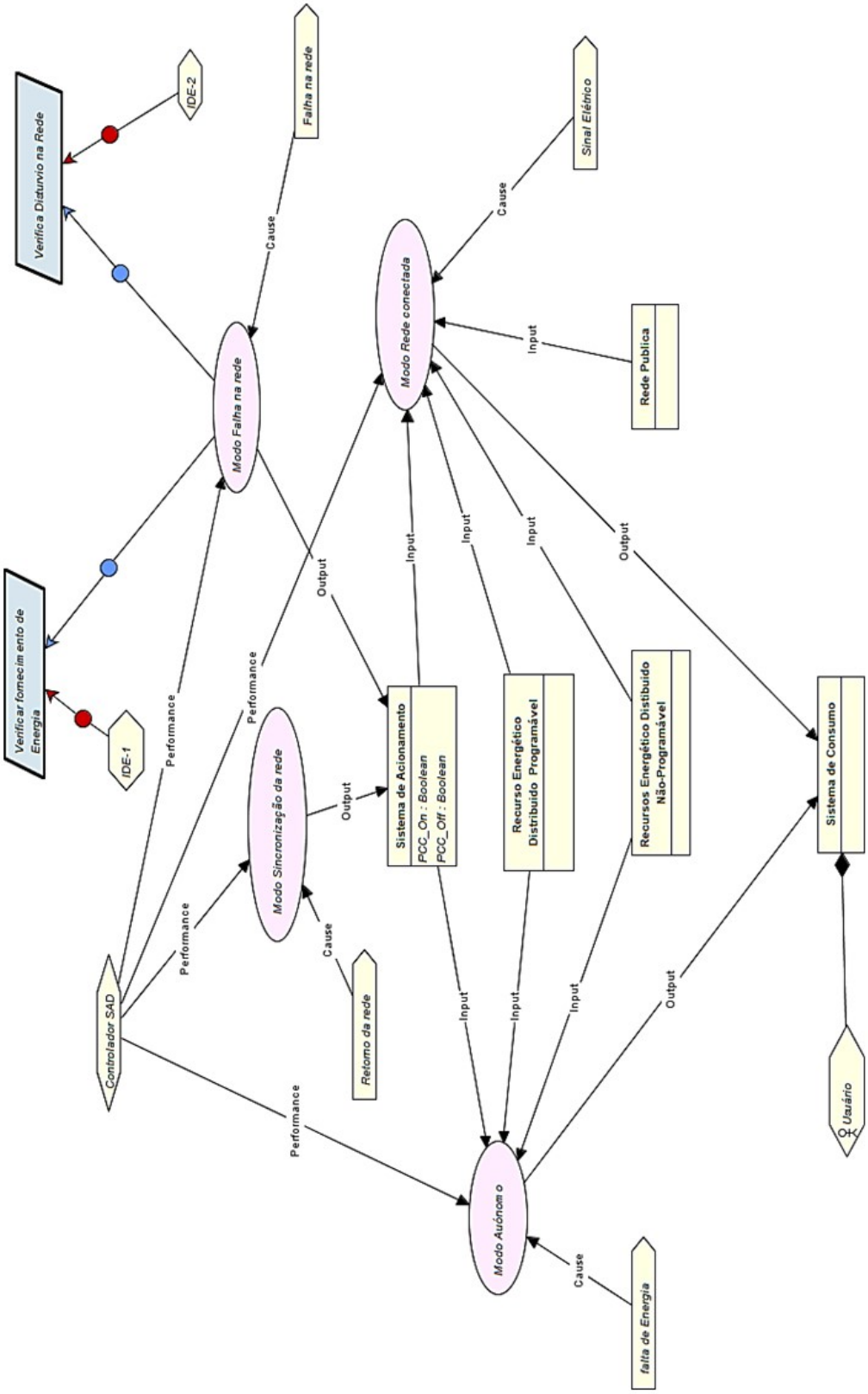


Figura 39 – Modelo de Operações

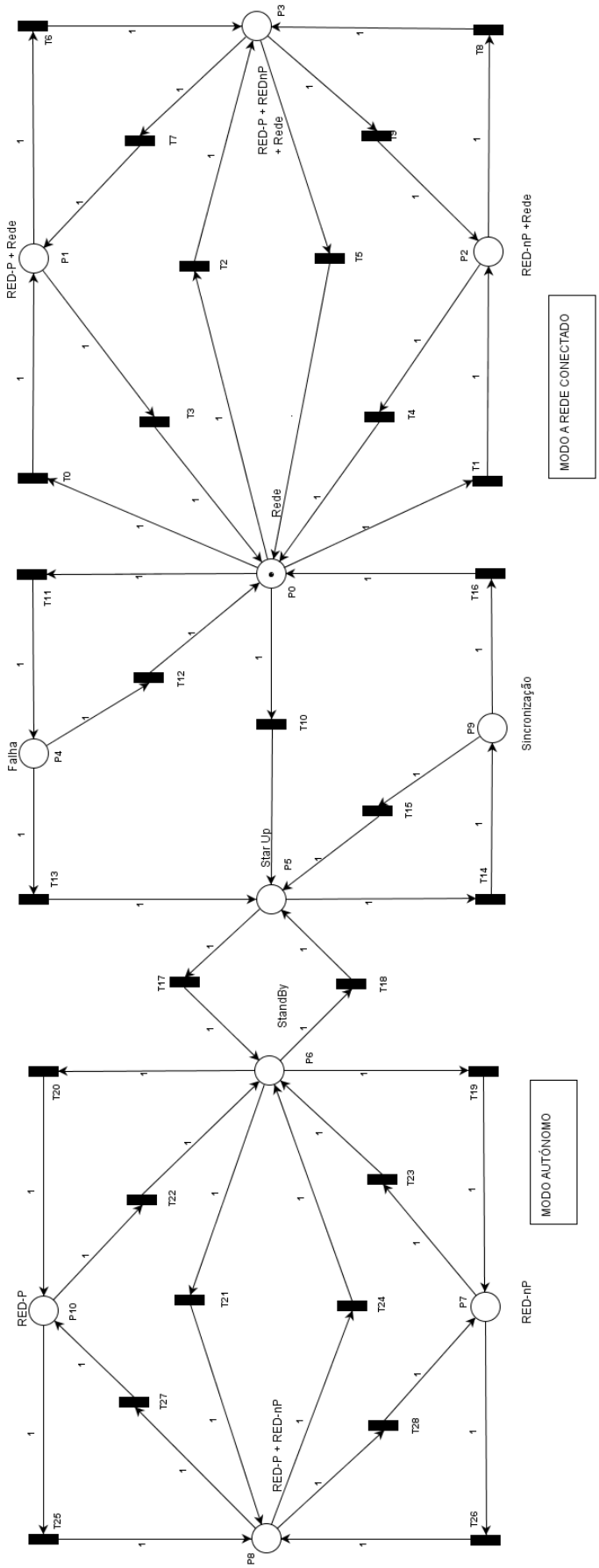


Figura 40 – Rede de Petri (Operação IEC-61850)

5.3.1 Verificação formal dos Requisitos

A verificação formal de requisitos é realizada através da análise da dinâmica da RdP da Figura 40. Foram analisadas as propriedades estruturais e comportamentais e avaliação de desempenho da rede, e também foram analisados os componentes conservativos e repetitivos da rede, isto é, as invariantes da matriz de incidência e, equação de estados.

Após fazer a análise de invariantes, verificou-se que a rede é coberta por P-Invariantes positivos, portanto, é limitada.

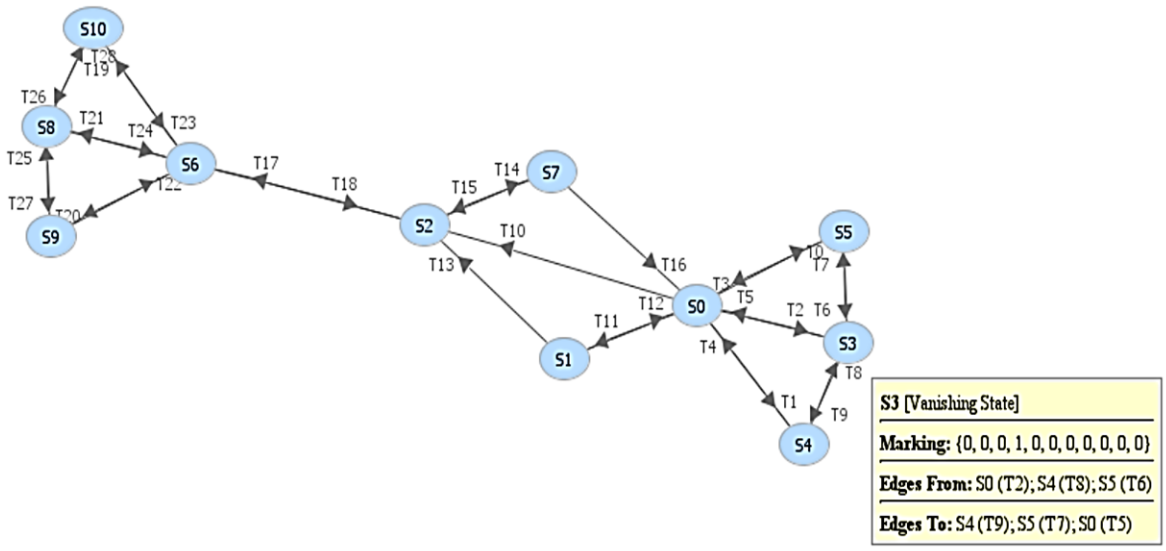
P-Invariantes										
P0	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
1	1	1	1	1	1	1	1	1	1	1

Em seguida é mostrado o resultado das Equações de Invariantes de Lugar (P), que comprova que a rede é limitada (a soma dos *tokens* dos lugares é igual a um):

M(P0) + M(P1) + M(P2) + M(P3) + M(P4) +M(P5) + M(P6) + M(P7) + M(P8) + M(P9) + M(P10) = 1

Ou seja, a soma total das marcas permanece constante durante toda sua execução.

Figura 41 – Arvore de alcançabilidade

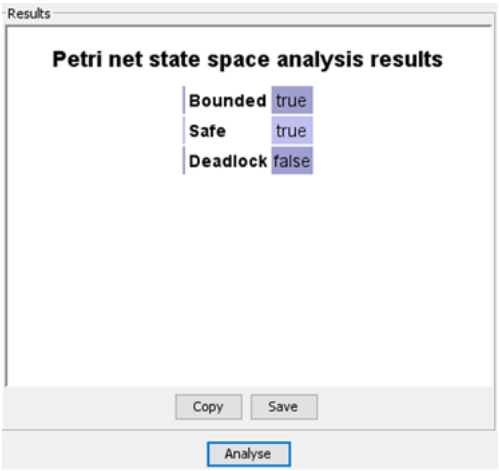


Fonte: próprio autor

Na pratica, à análise de invariantes permite aumentar a capacidade de análise dos requisitos através da antecipação da formalização, que é a base para potencializar os requisitos.

Portanto, através este processo de verificação comprova se os objetivos especificados (final e intermediários) foram atingidos.

Figura 42 – Analise de resultados do espaço de estados da RdP



Fonte: próprio autor

Portanto, considerando as técnicas de análise relativas à árvore de alcançabilidade gerada pela RdP mostrada na figura 41, e da análise de resultados correspondente ao espaço de estados, apresentado na figura 42, pode-se concluir que:

- Todas as marcações podem ser atingidas desde a marcação inicial, portanto, a rede é viva e está livre *deadlock*, portanto, os objetivos representados pelos lugares da RdP, serão alcançados;
- A RdP representa adequadamente a operação da microgrid, isto é, a ausência de *deadlock* implica que não haverá interrupção no fornecimento de energia;
- A rede é limitada porque a árvore alcançabilidade também é limitada, o que implica que a sistema é fechado e, portanto, a microgrid é controlável;
- A análise de invariantes que mostra que o sistema é conservativo, isso significa, que as condições lógicas e eventos correspondentes à automação da microgrid permanecerão estáveis durante toda sua execução.

5.4 Síntese do Capítulo

Aplicamos o método proposto a um estudo de caso, uma microgrid em uma comunidade isolada da floresta Amazônica. Uma abordagem sistêmica foi apresentada baseada em SoS (e SoS+), e começa modelando o *sistem-as-is* (no caso existe um sistema legado) para entender e analisar a situação atual do projeto. O *sistem-as-is* é examinado para encontrar pontos de melhorias serem inseridas no *sistem-to-be*. Nesta parte do método, aplica-se a modelagem usando diagramas KAOS, que também foi proposta em vários trabalhos na literatura (HORKOFF *et al.*, 2019) (NWOKEKI; CLARK; BARNS, 2013) (RESPECT, 2007).

No entanto, diferentemente dos métodos clássicos, caracterizados por serem estritamente funcionais, usamos o método orientados a objetivos para modelar e analisar serviços, seguindo o ciclo de desenvolvimento de requisitos proposto no capítulo anterior. Portanto, além de modelar à automação da microgrid e as operações de serviço, também é necessária a modelagem do acoplamento entre o serviço oferecido e o usuário final.

Conforme mostrado no estudo de caso, tal análise revela mais claramente requisitos emergentes (surgidos justamente devido a mudanças de comportamento do usuário) e melhorias que podem ser representadas desde o início do processo de design ou adicionadas em um processo de manutenção ou inovação. Logo, esses recursos apareceram no processo de design de serviço, quando se aplica uma abordagem orientada a objetivos.

Devemos ainda destacar que aplicamos a abordagem orientada a objetivos visando retardar a discriminação expressa entre os aspectos funcionais e não funcionais, sempre pensando nos objetivos. Evitamos assim a pressão para balancear requisitos funcionais e não funcionais no início do processo, quando ainda estamos lidando com um conhecimento incompleto. Posteriormente, uma vez identificadas as necessidades para atingir os objetivos, podemos considerar os requisitos não funcionais necessários, e as restrições de domínio.

O design do serviço foi implementado heuristicamente, embora tenha se mostrado muito útil, ainda precisa ser formalizado. Por exemplo, o modelo atual admite modelagem formal direta para classes de usuários, mas o acoplamento entre o serviço prestado e os agentes usuários não é formal. Em outras palavras, é fundamental destacar a co-criação de valor.

Para fazer a validação do método proposto, considerando o tipo de sistema que estamos lidando (sistemas SG) - sistemas complexos e com alto nível de automação - utilizamos a modelagem como eventos discretos, baseados no paradigma estado/transição. O modelo formal utilizado para validação foi transferido para Redes de Petri, onde foi feita a análise de propriedades.

O método também abordou a especificação dos requisitos, através da utilização de arquiteturas de referência, mais especificamente a IEC 61850 que possibilitará um *feedback* para a modelagem quando o ciclo for realimentado.

CAPÍTULO

6

CONCLUSÕES E TRABALHO FUTURO

Neste capítulo apresenta-se as conclusões do trabalho, e as recomendações visando dar continuidade à pesquisa neste assunto.

6.1 Conclusões

Sistemas tradicionais de energia são caracterizados por serem centralizados por empresas de energia, geralmente concessionarias de serviços que pode também ser a responsável pela geração, transmissão e distribuição de energia elétrica. Em raros casos, em se tratando de sistemas sustentáveis, temos um sistema (centralizado) comunitário local, baseado na geração de fontes alternativas. Seja qual for o caso, novas adequações e inovações no projeto elétricos (do todo ou das unidades constituintes) devem ser compatíveis com sistema legado, ou seja, resultando em pouca flexibilidade no processo de *design*.

Entretanto, este modelo tradicional centralizado vem sendo modificado, devido ao surgimento dos modernos sistemas de energia denominados SG. Estes sistemas possuem um tipo de gestão de geração / consumo não centralizada, onde o usuário consumidor pode também ser produtor (prosumer). Neste novo cenário, os novos sistemas SG podem se adaptar ao sistema legado como podem funcionar de forma isolada. Por esse motivo, o domínio de aplicação de design deve ser mais flexível, e não apenas os aspectos técnicos (equipamentos, entre outros) devem ser considerando, mas também as diferentes formas de cooperação entre unidades, o acoplamento com o usuário final, além do já referido acoplamento com o sistema legado.

Esta complexidade dos modernos sistemas de energia nos leva à necessidade de novos métodos, ferramentas e conceitos para melhor capturar os requisitos e projetar, implementar e integrar novos arranjos de microgrids, especialmente em regiões mais isoladas, ou em arranjos urbanos mais complicados.

Portanto, para atender a essa demanda, defendemos que:

- os SGs devem ser modeladas e projetadas como sistemas, ou sistemas de sistemas (e não como um arranjo de equipamentos, diretamente certificados pelo modelo legado);
- os SGs devem ser projetadas como serviço, visando inclusive uma integração mais participativa com o usuário final;
- o design de sistemas SG deve ser em um ciclo de modelagem formalizado, permitindo a verificação do provimento (tanto como gerador como como consumidor) do serviço e seu acoplamento com o sistema legado (nesse trabalho não tratamos da formalização da interação com o usuário, ou *value co-creation*).
- o *design* de SG deve utilizar arquiteturas de referência já consagradas na área de desenvolvimento de sistemas elétricos, o que possibilite uma aceleração da fase inicial e também uma aceitação prática maior.

Para tratar esses desafios, propusemos um ciclo de desenvolvimento de requisitos aplicável a um domínio flexível e aos modernos sistemas elétricos (SG) baseado em uma abordagem sistêmica, e que antecipa a formalização para a fase de requisitos. Este método propõe a combinação de arquiteturas de referência com métodos modernos de requisitos orientados a objetivos, modelagem e verificação formal em RdP, através de uma representação orientada para o processo, baseada na análise de propriedades.

Metodologicamente, proposta inclui um ciclo de vida de requisitos orientado por modelos *Model Based System Engineering* (MBSE), *Goal Oriented Requirements Engineering* (GORE) e, *Service Oriented Engineering Design* (SOED), enfatizando a comunicação e o acoplamento com o usuário final - que é uma questão crítica na abordagem de serviço. Portanto a contribuição deste trabalho foi propor um método que reúne de forma harmônica MBSE, GORE e SOED em um mesmo ciclo, que se fecha com a formalização utilizando a Teoria de Sistemas Discretos e as Redes de Petri.

Assumiu-se que os requisitos sintetizados são dinâmicos, e devem persistir por toda fase de projeto, extrapolando o desenvolvimento de requisitos. Assim, é essencial que estes possam ser verificados em qualquer fase, preferencialmente pelo mesmo formalismo. As redes de Petri e a análise de invariantes podem fornecer a base formal para atingir esta meta e assim foram utilizadas neste trabalho. Note-se que não houve contribuição teórica ou prática deste trabalho para os algorítimos de análise de invariantes, e simplesmente acoplamos este processo (feito em ferramentas disponíveis e consagradas) no ciclo de desenvolvimento de requisitos proposto.

Para mostrar a viabilidade de resultados e o esforço necessário na prática escolhemos um estudo de caso real (na floresta amazônica), e com restrições substanciais de sustentabilidade. Na verdade foi usado um projeto já desenvolvido (de forma funcional) como *"sistem-as-is"* e proposta uma melhoraria e evolução do projeto original para uma microgrid distribuída, e que mudasse significativamente o relacionamento com o usuário. Ressaltando que as novas demandas

e requisitos emergentes deste mesmo usuário poderiam (e de fato estavam) causando certos problemas para o funcionamento efetivo do sistema.

Propomos assim um *sistem-to-be* com maior grau de automação potencialmente distribuído (depende é claro da capacidade de geração dos "prosumers", que, ainda que não produzam, são mais participativos, aumentando o *"value co-creation"*. Na especificação dos novos requisitos, usamos a arquitetura de referência IEC 61850, o que possibilitou um *feedback* na realimentação do ciclo. Os novos requisitos da automação da microgrid foram modelados usando os diagramas semi-formais KAOS, analisados, traduzidos para uma RdP por meio de um algoritmo de transferência RekPlan (que não é resultado deste trabalho), e verificados. Este ciclo poderia ser feito várias vezes refinando por ciclos o modelo de requisitos. Por razões óbvias - para não ser repetitivo e tedioso - foi colocado no estudo de caso apenas um grande ciclo.

Assim, feita a análise usando as Redes de Petri como representação formal, particularmente uma abordagem hierárquica, foi derivado um modelo e *workflow* de controle para as unidades prosumer e seu acoplamento com o sistema legado (SILVA; SILVA, 2019). Mesmo usando um único ciclo mostramos como um modelo parcial dos requisitos deve ser provado consistente e completo, sem inconsistências ou condições indesejadas, ou seja, sem gargalos e livre de *deadlock*.

O método proposto é substancialmente diferente da prática corrente, e mesmo dos métodos recomendados pela academia para o modelo hierárquico estrito, sempre fortemente dependente de um provedor/gerador. Explora métodos ainda em discussão ou ainda em aberto na literatura acadêmica e propõe um novo ciclo integrador. A introdução de métodos formais se ajusta à necessidade de modelar sistemas automatizados, com maior demanda de confiabilidade. Ao mesmo tempo confere maior flexibilidade e adaptabilidade, possibilitando a crescente adequação de sistemas existentes para receber novas unidades SG ou para mudar substancialmente o arranjo produtor/consumidor. Prevê ainda a reutilização de designs, como esquematizado na Figura 28. A documentação e organização dos resultados do design não foi alvo deste trabalho.

6.2 Trabalhos futuros

Um plano para trabalhos futuros inclui propostas para a documentação do Projeto, que pode ser obtida diretamente do modelo KAOS, ou de mudanças no software Objectiver [1], que estão sendo propostas e discutidas pelo D-Lab. Esta documentação possibilitará a elaboração da matriz de rastreabilidade dos requisitos e a devida classificação dos requisitos dinâmicos, ligados aos invariantes. Esta matriz servirá ainda como referência para a manutenção dos sistemas, e análise de impacto de mudanças.

[1] Esta possibilidade depende de entendimentos entre FAPESP e CORNET, o consórcio que dá suporte a projetos que beneficiam pequena e média empresa, que é o caso da RESPECT-IT, fabricante do Objectiver. A RESPECT-IT foi criada como *spin off* da Universidade de Leuven.

Outro desafio para o futuro consiste em resolver problemas de comunicação detectados no projeto, utilizando de arquiteturas de referência, e buscando uma comunicação ativa e em tempo real, como a que se propõe no momento para os sistemas de manufatura I4.0. Nesse caso um sistema unificado de mensagens online dá cobertura para a comunicação entre todos os agentes (humanos e máquinas) de modo a permitir a avaliação do funcionamento do sistema em qualquer instante. No caso de comunidades isoladas este *"feature"* tem custo/benefício questionável, mas nos arranjos urbanos é exatamente ao contrário.

A proposta do ciclo de desenvolvimento de requisitos apresenta características inovadoras referente ao ciclo fechado de requisitos no contexto de SoS e SoS+, e pode ser aplicada em sistemas de controle e supervisão da Indústria 4.0, onde também se encaixa o sistema de comunicação online descrito no parágrafo anterior. A aplicação o mesmo ciclo de requisitos para aplicações industriais é um desafio que pode ser encarado no futuro.

No que se refere aos objetivos mais específicos, temos como plano para o futuro ampliar o algoritmo de tradução automática do RekPlan (que transforma diagramas de objetivo KAOS para RdP) de modo a incluir os outros três tipos de diagramas: objetos, operação e responsabilidade. Este trabalho deve ser feito em paralelo com a validação da linguagem KML (KAOS *Markup Language*, proposta pelo D-Lab) que deve ser transformada em PNML (*Petri Net Markup Language*), dando um escopo mais formal para o algorítimo de transferência.

Finalmente, um outro objetivo para o futuro seria aprofundar a possibilidade de utilização das RdP hierárquicas de alto nível para dar suporte à análise de *workflow* em redes de SG no contexto de SoS/SoS+.

REFERÊNCIAS

ANDRÉN, F.; STRASSER, T.; KASTNER, W. Engineering smart grids: Applying model-driven development from use case design to deployment. **Energies**, Multidisciplinary Digital Publishing Institute, v. 10, n. 3, p. 374, 2017. Citado 2 vezes nas páginas 37 e 40.

ARNOLD, G. W.; WOLLMAN, D. A.; FITZPATRICK, G. J.; PROCHASKA, D.; HOLMBERG, D. G.; SU, D. H.; JR, A. R. H.; GOLMIE, N. T.; BREWER, T. L.; BELLO, M. *et al.* Nist framework and roadmap for smart grid interoperability standards, release 1.0. 2010. Citado 2 vezes nas páginas 36 e 78.

ASSAAD, M.; TALJ, R.; CHARARA, A. A view on systems of systems (sos). In: IFAC. **Proc. of the 20th World Congress of the International Federation of Automatic Control (IFAC WC 2017)**. [S.l.], 2017. Citado na página 46.

BAIER, C.; KATOEN, J.-P. **Principles of model checking**. [S.l.]: MIT press, 2008. Citado 4 vezes nas páginas 59, 61, 83 e 85.

BARESI, L.; PEZZÈ, M. Improving uml with petri nets. **Electronic Notes in Theoretical Computer Science**, Elsevier Science, v. 44, n. 4, p. 1–13, 2001. Citado na página 83.

BERGER, L. T.; INIEWSKI, K. **Smart grid: applications, communications, and security**. [S.l.]: Wiley, 2012. Citado 2 vezes nas páginas 52 e 53.

BERNAL-AGUSTÍN, J. L.; DUFO-LÓPEZ, R.; RIVAS-ASCASO, D. M. Design of isolated hybrid systems minimizing costs and pollutant emissions. **Renewable Energy**, Elsevier, v. 31, n. 14, p. 2227–2244, 2006. Citado na página 51.

BLAABJERG, F.; TEODORESCU, R.; LISERRE, M.; TIMBUS, A. V. Overview of control and grid synchronization for distributed power generation systems. **IEEE Transactions on industrial electronics**, Citeseer, v. 53, n. 5, p. 1398–1409, 2006. Citado 2 vezes nas páginas 50 e 51.

BRUNNER, C. Iec 61850 for power system communication. In: IEEE. **Transmission and Distribution Conference and Exposition, 2008. T&D. IEEE/PES**. [S.l.], 2008. p. 1–6. Citado na página 53.

BUEDE, D. M.; MILLER, W. D. **The engineering design of systems: models and methods**. [S.l.]: John Wiley & Sons, 2016. Citado 2 vezes nas páginas 24 e 58.

CABOT, J.; GOGOLLA, M. Object constraint language (ocl): a definitive guide. In: **Formal methods for model-driven engineering**. [S.l.]: Springer, 2012. p. 58–90. Citado na página 57.

CAILLIAU, A.; LAMSWEERDE, A. van. Assessing requirements-related risks through probabilistic goals and obstacles. **Requirements Engineering**, Springer, v. 18, n. 2, p. 129–146, 2013. Citado na página 60.

CHUNG, L.; NIXON, B. A.; YU, E.; MYLOPOULOS, J. **Non-functional requirements in software engineering**. [S.l.]: Springer Science & Business Media, 2012. v. 5. Citado na página 56.

CLAVEL, F.; SAVARY, E.; ANGAYS, P.; VIEUX-MELCHIOR, A. Integration of a new standard: A network simulator of iec 61850 architectures for electrical substations. **IEEE Industry Applications Magazine**, IEEE, v. 21, n. 1, p. 41–48, 2015. Citado na página 52.

COMMISSION, I. E. *et al.* **Intelligrid Methodology for Developing Requirements for Energy Systems**. [S.l.]: IEC, 2008. Citado 6 vezes nas páginas 26, 37, 38, 40, 41 e 48.

CRETU, L. G.; DUMITRIU, F. **Model-Driven Engineering of Information Systems: Principles, Techniques, and Practice**. [S.l.]: CRC Press, 2014. Citado 3 vezes nas páginas 25, 27 e 75.

DARDENNE, A.; LAMSWEERDE, A. V.; FICKAS, S. Goal-directed requirements acquisition. **Science of computer programming**, Elsevier, v. 20, n. 1-2, p. 3–50, 1993. Citado na página 58.

DAVID, R.; ALLA, H. Timed continuous petri nets. In: **Discrete, continuous, and hybrid Petri nets**. [S.l.]: Springer, 2010. p. 159–229. Citado 2 vezes nas páginas 68 e 69.

DENG, W.; PEI, W.; SHEN, Z.; ZHAO, Z.; QU, H. Adaptive micro-grid operation based on iec 61850. **Energies**, Multidisciplinary Digital Publishing Institute, v. 8, n. 5, p. 4455–4475, 2015. Citado na página 53.

DOBING, B.; PARSONS, J. Dimensions of uml diagram use: a survey of practitioners. **Journal of Database Management (JDM)**, IGI Global, v. 19, n. 1, p. 1–18, 2008. Citado 3 vezes nas páginas 40, 56 e 57.

DORFMAN, M.; THAYER, R. H. Standards, guidelines and examples on system and software requirements engineering. **IEEE Computer Society Press Tutorial, Los Alamitos: IEEE Computer Society Press, 1990, edited by Dorfman, Merlin; Thayer, Richard H.**, 1990. Citado na página 55.

EIJKELHOF, D.; RAPP, S.; FASEL, U.; GUANAA, M.; SCHMEHL, R. Reference design and simulation framework of a multi-megawatt airbone wind energy system. **Journal of Physics: Conference Series**, v. 1618, 2020. Citado na página 34.

ELÉTRICA, E. A. E. de E. Ministério de minas e energia. **Postado em**, 2018. Citado 4 vezes nas páginas 23, 24, 31 e 50.

ELGARGOURI, A.; ELFITURI, M. M.; ELMUSRATI, M. Iec 61850 and smart grids. In: IEEE. **Electric Power and Energy Conversion Systems (EPECS), 2013 3rd International Conference on**. [S.l.], 2013. p. 1–6. Citado na página 54.

FALCÃO, D. M. Integração de tecnologias para viabilização da smart grid. **III Simpósio Brasileiro de Sistemas Elétricos**, p. 1–5, 2010. Citado 4 vezes nas páginas 23, 48, 49 e 51.

FERRAREZI, R. C. **Framework para modelagem e verificação formal de programas de controle de sistemas instrumentados de segurança.** Tese (Doutorado) — Universidade de São Paulo, 2014. Citado na página 85.

FILHO, J. M.; MAMEDE, D. R. **Proteção de sistemas elétricos de potência**. [S.l.]: Grupo Gen-LTC, 2000. Citado 2 vezes nas páginas 49 e 50.

FITZPATRICK, G. J.; WOLLMAN, D. A. Nist interoperability framework and action plans. In: IEEE. **IEEE PES General Meeting**. [S.l.], 2010. p. 1–4. Citado 2 vezes nas páginas 36 e 37.

FOYO, P. M. G. del; SALMON, A.; SILVA, J. R. Requirements analysis of automated projects using uml/petri nets. In: **Proc. of COBEM**. [S.l.: s.n.], 2011. Citado 2 vezes nas páginas 83 e 85.

FRANGOPOULOS, C. A.; SPAKOVSKY, M. R. V.; SCIUBBA, E. A brief review of methods for the design and synthesis optimization of energy systems. **International Journal of Thermodynamics**, v. 5, n. 4, p. 151–160, 2002. Citado 2 vezes nas páginas 33 e 34.

FRAPPIER, M.; HABRIAS, H. **Software specification methods: an overview using a case study**. [S.l.]: Springer Science & Business Media, 2012. Citado na página 83.

GERBER, L. **Designing renewable energy systems: a life cycle assessment approach**. [S.l.]: EPFL Press, 2014. Citado 3 vezes nas páginas 25, 32 e 33.

GOMES, R.; JR., C. C.; J.R., S. Smartlvgrid platform—convergence of legacy low-voltage circuits toward the smart grid paradigm. **Energies**, v. 12, n. 13, 2019. Citado na página 77.

GOTTSCHALK, M.; USLAR, M.; DELFS, C. **The Use Case and Smart Grid Architecture Model Approach: The IEC 62559-2 Use Case Template and the SGAM Applied in Various Domains**. [S.l.]: Springer, 2017. Citado 2 vezes nas páginas 37 e 40.

HEFFNER, G. Smart grid–smart customer policy needs. In: **Workshop Report for the IEA Energy Efficiency Working Party, April, Paris**. [S.l.: s.n.], 2011. Citado na página 48.

HORKOFF, J.; AYDEMIR, F. B.; CARDOSO, E.; LI, T.; MATÉ, A.; PAJA, E.; SALNITRI, M.; MYLOPOULOS, J.; GIORGINI, P. Goal-oriented requirements engineering: a systematic literature map. In: IEEE. **Requirements Engineering Conference (RE), 2016 IEEE 24th International**. [S.l.], 2016. p. 106–115. Citado 2 vezes nas páginas 57 e 58.

HORKOFF, J.; AYDEMIR, F. B.; CARDOSO, E.; LI, T.; MATÉ, A.; PAJA, E.; SALNITRI, M.; PIRAS, L.; MYLOPOULOS, J.; GIORGINI, P. Goal-oriented requirements engineering: an extended systematic mapping study. **Requirements engineering**, Springer, v. 24, n. 2, p. 133–160, 2019. Citado na página 105.

IBGE. 2018. <https://ww2.ibge.gov.br/home/>. Accessed: 2018-02-26. Citado na página 23.

INCOSE. Sebok v.2.2. **SEBOK**, INCOSE San Diego, CA (US), 2020. Citado 2 vezes nas páginas 44 e 75.

INCOSE, R. Guide for writing requirements. **Version 2. Prepared by: Requirements Working Group**, INCOSE San Diego, CA (US), 2015. Citado 5 vezes nas páginas 24, 43, 44, 45 e 75.

JAMSHIDI, M. **Systems of systems engineering: principles and applications**. [S.l.]: CRC press, 2008. Citado na página 75.

KENDAL, S. L.; CREEN, M. **An introduction to knowledge engineering**. [S.l.]: Springer, 2007. Citado na página 24.

LAMSWEERDE, A. V. **Requirements engineering: From system goals to UML models to software**. [S.l.]: Chichester, UK: John Wiley & Sons, 2009. v. 10. Citado 4 vezes nas páginas 56, 57, 59 e 84.

LAMSWEERDE, A. van. Goal-oriented requirements engineering: From system objectives to uml models to precise software specifications. In: IEEE COMPUTER SOCIETY. **Proceedings of the 25th International Conference on Software Engineering**. [S.l.], 2003. p. 744–745. Citado na página 58.

____. Goal-oriented requirements enginering: a roundtrip from research to practice [enginering read engineering]. In: IEEE. **Requirements Engineering Conference, 2004. Proceedings. 12th IEEE International**. [S.l.], 2004. p. 4–7. Citado na página 55.

LAPLANTE, P. A. **Requirements engineering for software and systems**. [S.l.]: Auerbach Publications, 2017. Citado na página 58.

LAPOUCHNIAN, A. Goal-oriented requirements engineering: An overview of the current research. **University of Toronto**, v. 32, 2005. Citado 2 vezes nas páginas 55 e 59.

LOPES, A. J.; LEZAMA, R.; PINEDA, R. Model based systems engineering for smart grids as systems of systems. **Procedia Computer Science**, Elsevier, v. 6, p. 441–450, 2011. Citado na página 77.

MACKIEWICZ, R. E. Overview of iec 61850 and benefits. In: IEEE. **Power Systems Conference and Exposition, 2006. PSCE'06. 2006 IEEE PES**. [S.l.], 2006. p. 623–630. Citado na página 52.

MAIER, M. W. Architecting principles for systems-of-systems. **Systems Engineering: The Journal of the International Council on Systems Engineering**, Wiley Online Library, v. 1, n. 4, p. 267–284, 1998. Citado na página 46.

MARTIN, J. The seven samurais of systems engieering: Dealing with the complexity of 7 interrelated systems. In: **Symposium of the Int. Council on Systems Engineering (INCOSE)**. [S.l.: s.n.], 2004. p. 2–11. Citado 4 vezes nas páginas 44, 78, 82 e 87.

MARTINEZ, J. R.; SAIDEL, M. A.; FADIGAS, E. A. Influence of non-dispatchable energy sources on the dynamic performance of microgrids. **Electric Power Systems Research**, Elsevier, v. 131, p. 96–104, 2016. Citado na página 87.

MARZAL, S.; SALAS, R.; GONZÁLEZ-MEDINA, R.; GARCERÁ, G.; FIGUERES, E. Current challenges and future trends in the field of communication architectures for microgrids. **Renewable and Sustainable Energy Reviews**, Elsevier, 2017. Citado na página 41.

MAZZOLINI, M.; BRUSAFERRI, A.; CARPANZANO, E. An integrated framework for model-based design and verification of discrete automation solutions. In: IEEE. **Industrial Informatics (INDIN), 2011 9th IEEE International Conference on**. [S.l.], 2011. p. 545–550. Citado na página 25.

MURATA, T. Petri nets: Properties, analysis and applications. **Proceedings of the IEEE**, IEEE, v. 77, n. 4, p. 541–580, 1989. Citado 3 vezes nas páginas 69, 70 e 71.

NAFI, N. S.; AHMED, K.; GREGORY, M. A.; DATTA, M. A survey of smart grid architectures, applications, benefits and standardization. **Journal of Network and Computer Applications**, Elsevier, v. 76, p. 23–36, 2016. Citado 3 vezes nas páginas 34, 35 e 41.

NAUMANN, A.; BIELCHEV, I.; VOROPAI, N.; STYCZYNSKI, Z. Smart grid automation using iec 61850 and cim standards. **Control Engineering Practice**, Elsevier, v. 25, p. 102–111, 2014. Citado na página 52.

NIELSEN, C. B.; LARSEN, P. G.; FITZGERALD, J.; WOODCOCK, J.; PELESKA, J. Systems of systems engineering: basic concepts, model-based techniques, and research directions. **ACM Computing Surveys (CSUR)**, ACM New York, NY, USA, v. 48, n. 2, p. 1–41, 2015. Citado 2 vezes nas páginas 45 e 46.

NWOKEKI, J.; CLARK, T.; BARNS, B. Towards a comprehensive meta-model for kaos. In: IEEE. **Proc. of IEEE Model-Driven Requirements Engineering Workshop - MoDRE**. [S.l.], 2013. p. 30–39. Citado na página 105.

NYSO. Distributed energy resources roadmap for new york's wholesale electricity markets: A report by the new york independent system operator. 2017. Citado 3 vezes nas páginas 31, 50 e 51.

OLIVEIRA, V. C. de; SILVA, J. R. A service-oriented framework to the design of information system service. **Journal of Service Science Research**, Springer, v. 7, n. 2, p. 55–96, 2015. Citado 3 vezes nas páginas 23, 83 e 86.

PAPAEFTHYMIOU, S. V.; KARAMANOU, E. G.; PAPATHANASSIOU, S. A.; PAPADOPOULOS, M. P. A wind-hydro-pumped storage station leading to high res penetration in the autonomous island system of ikaria. **IEEE Transactions on Sustainable energy**, IEEE, v. 1, n. 3, p. 163–172, 2010. Citado na página 49.

PARVEEN, S.; IMAM, A. Analysis of different techniques of gore (goal oriented requirement engineering). **Global Sci-Tech**, Al-falah School of Engineering and Technology, v. 9, n. 1, p. 22–36, 2017. Citado na página 58.

PETRINETS. 2018. <www.informatik.uni-hamburg.de/TGI/PetriNets/index.php>. Accessed: 2018-02-26. Citado 3 vezes nas páginas 72, 81 e 85.

PONSARD, C.; DARIMONT, R.; MICHOT, A. Combining models, diagrams and tables for efficient requirements engineering: Lessons learned from the industry. In: **INFORSID**. [S.l.: s.n.], 2015. p. 235–250. Citado na página 59.

POSTIGO, M. A. O.; SILVA, J. M.; SILVA, J. R. Applying goal-oriented requirement engineering and reference architecture for microgrid systems. **Brazilian Applied Science Review**, v. 4, n. 3, p. 913–926, 2020. Citado 2 vezes nas páginas 25 e 75.

POSTIGO, M. A. O.; SILVA, J. R. Modeling in petri nets for micro smart grid operation based on iec 61850 architecture. In: IEEE. **2018 Simposio Brasileiro de Sistemas Eletricos (SBSE)**. [S.l.], 2018. p. 1–6. Citado na página 25.

____. Microgrid system design based on model based systems engineering: the case study in the amazon region. **Simpósio Brasileiro de Sistemas Elétricos-SBSE**, v. 1, n. 1, 2020. Citado na página 50.

QUEIROZ, R. A. d. A.; SILVA, J. R. *et al.* Eliciação e comunicação de requisitos em domínios disjuntos: estudo de caso para automação na área médica. **Sba: Controle & Automação Sociedade Brasileira de Automatica**, SciELO Brasil, v. 20, n. 4, p. 637–651, 2009. Citado na página 82.

RANGANATHAN, P.; NYGARD, K. E.; MAGEL, K. Uml design patters in smart grid. In: **CATA**. [S.l.: s.n.], 2011. p. 114–119. Citado na página 41.

RESPECT, I. **A kaos tutorial**. [S.l.]: Objectiver, 2007. Citado 11 vezes nas páginas 59, 61, 62, 63, 64, 65, 66, 67, 68, 82 e 105.

ROBOAM, X. **Systemic design methodologies for electrical energy systems: analysis, synthesis and management**. [S.l.]: John Wiley & Sons, 2012. Citado 4 vezes nas páginas 25, 31, 32 e 40.

RODRIGUEZ, P.; TIMBUS, A. V.; TEODORESCU, R.; LISERRE, M.; BLAABJERG, F. Flexible active power control of distributed power generation systems during grid faults. **IEEE Transactions on Industrial Electronics**, IEEE, v. 54, n. 5, p. 2583–2592, 2007. Citado 3 vezes nas páginas 24, 26 e 49.

ROHJANS, S.; DÄNEKAS, C.; USLAR, M. Requirements for smart grid ict-architectures. In: IEEE. **Innovative Smart Grid Technologies (ISGT Europe), 2012 3rd IEEE PES International Conference and Exhibition on**. [S.l.], 2012. p. 1–8. Citado 5 vezes nas páginas 25, 36, 37, 38 e 40.

RUMBAUGH, J.; JACOBSON, I.; BOOCH, G. **Unified modeling language reference manual, the**. [S.l.]: Pearson Higher Education, 2004. Citado 2 vezes nas páginas 56 e 57.

RUTA, M.; SCIOSCIA, F.; SCIASCIO, E. D.; LOSETO, G. Semantic-based enhancement of iso/iec 14543-3 eib/knx standard for building automation. **IEEE Transactions on Industrial Informatics**, IEEE, v. 7, n. 4, p. 731–739, 2011. Citado na página 35.

SALMON, A. Z. O.; SILVA, J. R. Usando invariantes na analise de requisitos. In: **CBA Congresso Brasileiro de Automática. Campina Grande/PB-Brasil**. [S.l.: s.n.], 2012. Citado na página 71.

SALOMON, A. Z. O. **Modelagem e análise de requisitos de sistemas automatizados usando UML e Redes de Petri.** Tese (Doutorado) — Universidade de São Paulo, 2017, 2017. Citado 2 vezes nas páginas 57 e 72.

SATO, T.; KAMMEN, D. M.; DUAN, B.; MACUHA, M.; ZHOU, Z.; WU, J.; TARIQ, M.; ASFAW, S. A. **Smart grid standards: specifications, requirements, and technologies**. [S.l.]: John Wiley & Sons, 2015. Citado 3 vezes nas páginas 34, 35 e 36.

SECHILARIU, M.; LOCMENT, F. **Urban DC Microgrid: Intelligent Control and Power Flow Optimization**. [S.l.]: Butterworth-Heinemann, 2016. Citado na página 48.

SEVERINO, A. J. **Metodologia do trabalho científico**. [S.l.]: Cortez editora, 2017. Citado 2 vezes nas páginas 28 e 29.

SILVA, J. M. **Modelagem e análise de conhecimento para planejamento automático: uma abordagem baseada em GORE e redes de Petri.** Tese (Doutorado) — Universidade de São Paulo, 2016. Citado 2 vezes nas páginas 26 e 79.

SILVA, J. M. Using gore method for requirement engineering of planning & scheduling. In: **The 26th International Conference on Automated Planning and Scheduling**. [S.l.: s.n.], 2016. p. 137. Citado na página 25.

SILVA, J. M.; SILVA, J. R. Combining kaos and ghenesys in the requirement and analysis of service manufacturing. **IFAC-PapersOnLine**, Elsevier, v. 48, n. 3, p. 1634–1639, 2015. Citado 2 vezes nas páginas 40 e 84.

____. A new hierarchical approach to requirement analysis of problems in automated planning. **Engineering Applications of Artificial Intelligence**, v. 81, p. 373–386, 2019. Citado 2 vezes nas páginas 101 e 109.

SILVA, J. R.; NOF, S. Y. Manufacturing service: from e-work and service-oriented approach towards a product-service architecture. **IFAC-PapersOnLine**, Elsevier, v. 48, n. 3, p. 1628–1633, 2015. Citado 2 vezes nas páginas 24 e 76.

SILVA, M. Half a century after carl adam petri's ph. d. thesis: A perspective on the field. **Annual reviews in control**, Elsevier, v. 37, n. 2, p. 191–219, 2013. Citado 2 vezes nas páginas 68 e 84.

SILVA, R.; ARAKAKI, J.; JUNQUEIRA, F.; FILHO, D. J. d. S.; MIYAGI, P. E. Modeling of active holonic control systems for intelligent buildings. **Automation in Construction**, Elsevier, v. 25, p. 20–33, 2012. Citado na página 25.

SILVA, R. M. da; SCHETTINO, S.; LIMA, P. M. R. de; ADRIANO, F. C. Smart grid: evaluation and trend in brazil. **Independent Journal of Management & Production**, Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, v. 5, n. 3, p. 752–776, 2014. Citado na página 49.

SOBEIH, A.; VISWANATHAN, M.; HOU, J. C. Check and simulate: A case for incorporating model checking in network simulation. In: IEEE. **Formal Methods and Models for Co-Design, 2004. MEMOCODE'04. Proceedings. Second ACM and IEEE International Conference on**. [S.l.], 2004. p. 27–36. Citado na página 85.

SOMMERVILLE, I. **Engenharia de software. traduçao: Selma shin shimuzu melnikoff, reginaldo arakaki, edilson de andrade barbosa**. [S.l.]: São Paulo: Pearson Addison Wesley, 2007. Citado na página 55.

USLAR, M.; ROHJANS, S. *et al.* Applying the smart grid architecture model for designing and validating system-of-systems in the power and energy domain: A european perspective. **Energies**, Multidisciplinary Digital Publishing Institute, v. 12, n. 2, p. 258, 2019. Citado 2 vezes nas páginas 39 e 40.

USLAR, M.; SPECHT, M.; DÄNEKAS, C.; TREFKE, J.; ROHJANS, S.; GONZÁLEZ, J. M.; ROSINGER, C.; BLEIKER, R. **Standardization in smart grids: introduction to IT-related methodologies, architectures and standards**. [S.l.]: Springer Science & Business Media, 2012. Citado 5 vezes nas páginas 26, 37, 38, 39 e 40.

VASUDEVAN, J.; SEKKAPPAN, K.; YASHWANT, K.; JAY, D.; SWARUP, K. Research issues in smart grid-application to automation, renewable energy sources and demand response. In: IEEE. **India Conference (INDICON), 2012 Annual IEEE**. [S.l.], 2012. p. 1132–1137. Citado na página 52.

VÁZQUEZ, M. J. V.; MÁRQUEZ, J. M. A.; MANZANO, F. S. A methodology for optimizing stand-alone pv-system size using parallel-connected dc/dc converters. **IEEE Transactions on Industrial Electronics**, IEEE, v. 55, n. 7, p. 2664–2673, 2008. Citado na página 51.

WALDEN, D. D.; ROEDLER, G. J.; FORSBERG, K.; HAMELIN, R. D.; SHORTELL, T. M. **Systems engineering handbook: A guide for system life cycle processes and activities**. [S.l.]: John Wiley & Sons, 2015. Citado 3 vezes nas páginas 43, 44 e 82.

WERNECK, V. M. B.; OLIVEIRA, A. d. P. A.; LEITE, J. C. S. do P. Comparing gore frameworks: i-star and kaos. In: **WER**. [S.l.: s.n.], 2009. Citado na página 55.

YU, E. S. Towards modelling and reasoning support for early-phase requirements engineering. In: IEEE. **Requirements Engineering, 1997., Proceedings of the Third IEEE International Symposium on**. [S.l.], 1997. p. 226–235. Citado na página 58.

ZHAO, B.; XUE, M.; ZHANG, X.; WANG, C.; ZHAO, J. An mas based energy management system for a stand-alone microgrid at high altitude. **Applied Energy**, Elsevier, v. 143, p. 251–261, 2015. Citado na página 52.

GLOSSÁRIO

Data-flow: Na análise estruturada refere-se à passagem de informação desde uma entidade para outra. Pode ser a passagem de um grupo de dados ou de um único dado.

Deadlocks: A situação de deadlocks é caraterizada quando a rede de Petri se encontra em um estado que não tem sucessor, isto é, onde nenhuma transição pode ser habilitada. Isto pode ser o resultado de uma má escolha do estado inicial ou da sequencia de estados (das transições separadas).

Model checking: É uma técnica de verificação automática para sistemas concurrentes finitos que tem seu origem no começo dos 80. É possível analisar diversos tipos de sistemas: sistemas em tempo real, sistemas híbridos, entre outros e sobre vários domínios: sistemas industriais, processo de negócios, etc.

Rationales: Dentro da Engenharia de Requisitos responde ao "Por quê" dos requisitos.

Roadmap: Trata-se de uma poderosa ferramenta visual e descritiva, que tem como principal função apontar como será o produto ou projeto em cada período de sua evolução. É um mapa que visa alinhar todos os stakeholders (interessados no projeto), ao redor dos mesmos passos sequenciais rumo à construção integral do produto. Deixando todos os envolvidos cientes do processo de evolução e quais variáveis que envolvem esse caminho.

Stakeholders: Qualquer usuário (pessoa, grupo de pessoas ou organizações) afetado pelo sistema de alguma forma. Isto inclui os usuários diretos do sistema, os desenvolvedores, os clientes, os fornecedores, os órgãos governamentais e mesmo usuários que diretamente não manipulam o sistema (ou mesmo não sabem que este existe), mas cujos efeitos do sistema podem interferir de alguma forma nas atividades destes.

Standish Group: É um grupo que tem a missão de estudar e pesquisar sobre as causas para o sucesso dos projetos ou a desafios na gestão de projetos e assim, sugerir novos métodos no gerenciamento.

Traceability: Referido à rastreabilidade dos requisitos. Documentar a "vida" de um requisito e fornecer rastreabilidade bidirecional. É possível encontrar a origem de cada requisito e acompanhar cada alteração que foi feita.

Viewpoints: Visões como opiniões (pontos de vista) no contexto social, em que cada um dos interessados tem suas próprias premissas, prioridades e experiências e lidam com os problemas de maneiras diferentes. Assim, é necessário conhecer, comparar e negociar as diferentes opiniões ou diferentes perspectivas de enxergar o mesmo artefato, por exemplo: qual a opinião do gerente, do comprador e do vendedor sobre uma compra.

Workflow management: É o gerenciamento e sistematização do processo de negócio, na sua totalidade ou em partes, onde documentos, informações ou tarefas são passadas de um participante para o outro para execução de uma ação, de acordo com um conjunto de regras de procedimentos.

APÊNDICE

A

FORMALIZAÇÃO DOS REQUISITOS EM LTL

Tabela 7 – Sentenças LTL do Objetivo de "Reduzir o consumo de energia"

Goal	**LTL Sentences**
G1 - The goal “Improve Operational Management” was done successesfully.	∀ (G1: Operational_Management, G2: User coupling, G3: microgrid control): □ (G1 ⇒ G2 ∧ G3)
G2 - The goal “Increase User participation” was done successfully.	∀ (R1: measurement of electrical parameters, R2: energy consumption and production, R3: satisfaction of the service offered, R4: use of renewable energy, R5: Reduces energy consumption): ♢ (G2 ⇒ R1 ∧ R2 ∧ R3 ∧ R4 ∧ R5).
G3 - The goal “Improve Monitoring and controly” was achieved successfully.	∀ (G3: microgrid control, G4: Energy Savings, G5: Supervision of energy consumption, G6: Theft Detection Control, G7: failure control): □ (G3 ⇒ G4 ∧ G5 ∧ G6 ∧ G7)
G4 - The goal “Maintain Energy Savings” was achieved successfully.	∀ (G4: Energy Savings, G8:energy efficiency, O1: Unidentified user energy saving) : □ (G4 ⇒ G8 ∨ O1)
G5 - The goal “Supervision of energy consumption and production” was achieved successfully.	∀ (G5: Supervision of energy, G9: reading and reporting energy , R1: measurement of electrical parameters) : □ (G5⇒ G9, ∧ R1)
G6 - The goal “Improve Theft Detection Control” was achieved successfully.	∀ (G6: Theft Detection Control, G10: illicit consumption, R6: send energy measurement to concessionaire, R13: load limitation and blocking control) : ♢ (G6 ⇒ G10 ∧ R6 ∧ R13),
G7 - The goal “Improve failure control” was achieved successfully.	∀ (G7: failure control, R15: Supervise failure control system) : ♢ (G7 ⇒ R15)
G8 - The goal “Increase energy efficiency” was achieved successfully.	∀ (G8: energy efficiency, R4: energy efficiency, R5: energy consumption) : ♢ (G8 ⇒ R4 ∧ R5)
G9 - The goal “Maintain reading and reporting energy measurement” was achieved successfully.	∀ (G11: reading of energy consumed and produced, R7: send energy measurement to concessionaire, R6: receive user consumption information) ∨ O2: Defective smart meter : □ (G9 ⇒ G11 ∧ R7 ∧ R6) ∨ ¬ O2))
G10 - The goal “Improve detect illicit consumption” was achieved successfully.	∀ (G10: detect illicit consumption, R14: Analyzes consumption history): □ (G10 ⇒ R14)
G11 - The goal “Improve Improve reading of energy consumed and produced and produced” was achieved successfully.	∀ (G11: Improve reading of energy consumed and produced, R8:payment of energy consumed, R9:Alerts user to excessive energy usage, R10: Show updated energy cost, R11:Send energy production consumption data, R12: measurement of electrical parameters) : □ (G11 ⇒ R8 ∧ R9 ∧ R10 ∧ R11 ∧ R12)

APÊNDICE

B

PUBLICAÇÕES DECORRENTES DA PESQUISA

Tabela 8 – Joutnal Sustainable Energy, Grids and Networks - Elsevier

Título	**Towards a Model Based Design Method for Microgrids: A Case study in Amazon forest communities**
Abstract	Energy systems design has increased in complexity due to new demands of being sustainable, distributed, heterogeneous, scalable, self-healing while preserving high performance. New approaches, methods, concepts, and tools emerged to characterize, understand, and specify hybrid production/consumer energy systems fully adapted to the contextual environment. This work presents a proposal for a systemic design method applied to Smart Grid (SG) clustering and based on a service approach. It inserts a coupling between the system, user, and the domain where it is supposed to work. The proposal emphasizes the modeling and specification of formal requirements for microgrids systems based on objectives, using Goal-Oriented Requirements Engineering (GORE), which provides a better framework (than conventional functional approaches) to design sustainable energy systems. The proposed method starts by representing the legacy energy system (if it exists) and projecting the enhancements by new requirements that are further analyzed and verified. These steps compose a cycle consisting of elicitation, analysis, validation, formal modeling, and verification. A case study is presented, revising an energy system installed in an isolated Amazon rain forest community. The results showed effectiveness in detecting objectives, hidden ambiguities in the legacy speciation, and anticipating emergent requirements that would appear later.
Palavras Chaves	Microgrids, systems design, model based systems engineering, service systems, goal-oriented requirements engineering
Evento/Periódico	Joutnal Sustainable Energy, Grids and Networks - Elsevier.
Ano	2021
ISSN	2352-4677
Autores	Postigo, M.A; Silva, J.R.; Pellini, E.L; Sicchar, J.R

Tabela 9 – Cap-15 Livro: Eng. Elétrica

Título	**MICROGRID SYSTEM DESIGN BASED ON MODEL BASED SYSTEMS ENGINEERING: THE CASE STUDY IN THE AMAZON REGION**
Abstract	Microgrid is a technically and economically viable opportunity to meet the demands of populations that, for various reasons, do not have access to electricity. The complexity of Smart Grid (SG) systems requires considerable engineering effort in the design process. Designing this type of complex system requires new approaches, methods, concepts and engineering tools. Where, requirements analysis plays a major role in better characterizing, understanding and specifying the domain of application that SG systems should solve. This work presents a systemic proposal based specifically on System Systems (SoS) which anticipates the formalization of requirements, aiming to understand, analyze and design SG within the scope of Model Based Systems Engineering (MBSE). The definition of a microgrid from the SoS perspective is presented in order to provide a complete view of its life cycle. Requirements would be represented in an Objective Oriented Requirements Engineering (GORE) approach, specifically using visual diagrams based on the Keep All Objectives Satisfied (KAOS) method, where network operation and control will be formally represented. A case study for small communities in the equatorial Amazon forest is used as a case study for the proposed method.
Palavras Chaves	Smart Grid; System Design; MBSE, Requeriments Engineering; KAOS.
Evento/Periódico	Capítulo de Livro. Engenharia Elétrica: Desenvolvimento e Inovação Tecnológica
Ano	2021
DOI	10.22533/at.ed.73421220215.
Autores	Postigo, M.A; Silva, J.R.

Tabela 10 – INDUSCOM-2021

Título	**Proposta de método sistêmico baseado em modelos para Smart Grid**
Abstract	O design de sistemas modernos de energia aumentou em complexidade devido à demanda por novas características, como: ser sustentável, distribuído, heterogêneo, escalável, colaborativo, preservando um alto desempenho. Novas abordagens, métodos, conceitos e ferramentas surgiram para caracterizar, compreender e especificar sistemas híbridos de produção/consumo de energia totalmente adaptados ao ambiente de entorno onde devem funcionar. Este trabalho apresenta uma proposta de método sistêmico para o desenvolvimento de requisitos aplicado a Smart Grid (SG), baseado em uma abordagem de serviço. Portanto, privilegia o acoplamento entre o sistema, o usuário e o domínio. A proposta enfatiza a antecipação da formalização, já na especificação de requisitos, para redes distribuídas de SGs, com análise e verificação do workflow em Redes de Petri (RdP). A rede é sintetizada à partir de diagramas semi-formais, baseados na Engenharia de Requisitos Orientada a Objetivos (GORE). É proposto um ciclo iterativo de desenvolvimento de requisitos composto pelas fases de eliciação, modelagem, análise, modelagem formal e verificação. O ciclo proposto foi aplicado a um estudo de caso, para o upgrade de uma microgrid localizada em uma comunidade da floresta amazônica.
Palavras Chaves	smart grid, system design,model-based systems engineering, goal-oriented requirements engineering.
Evento/Periódico	INDUSCOM 2021 - IEEE Transactions on Industry Applications.
Ano	2021
ISBN	978-1-5386-7994-4
Autores	Postigo, M.A; Silva, J.R.; Pellini, E.L

Tabela 11 – CBA-2020

Título	**MICROGRID SYSTEM DESIGN BASED ON MODEL BASED SYSTEMS ENGINEERING AND GOAL-ORIENTED REQUIREMENTS ENGINEERING**
Abstract	Microgrids appear as a practical, clean and reliable solution to meet the demand of populations that, for various reasons, do not have access to electricity. The complexity of microgrid systems, requires considerable engineering effort in the design process. To design this type of complex systems, new approaches, methods, concepts and engineering tools are needed. Where, the requirements analysis has a preponderant role to better characterize, understand and specify the application domain and the problem that the microgrids must solve. This work proposes the introduction of a formal analysis of requirements in the life cycle of microgrid systems, using IEC 61850 as a reference architecture. The requirements would be represented in an Object Oriented Requirements Engineering (GORE) approach, using specifically visual diagrams based on the KAOS (Keep All Objectives Satisfied) method, where the operation and control of the network will be formally represented. The requirements analysis is presented using a combined representation that uses the GORE and Petri Nets methodology for dynamic modeling and formal verification. A case study for small communities in the Amazon rainforest is used as a case study for the proposed method.
Palavras Chaves	System Design, Formal Modeling, GORE methodology, Petri Nets, microgrid.
Evento/Periódico	CBA-2020. XXIII Congresso Brasileiro de Automática.
Ano	2020
DOI	10.48011/asba.v2i1.1146
Autores	Postigo, M.A; Silva, J.M; Silva, J.R.

Tabela 12 – SBSE-2020

Título	**Microgrid System Design Based on Model Based Systems Engineering: the Case Study in the Amazon Region**
Abstract	Microgrid is a technically and economically viable opportunity to meet the demands of populations that, for various reasons, do not have access to electricity. The complexity of Smart Grid (SG) systems requires considerable engineering effort in the design process. Designing this type of complex system requires new approaches, methods, concepts and engineering tools. Where, requirements analysis plays a major role in better characterizing, understanding and specifying the domain of application that SG systems should solve. This work presents a systemic proposal based specifically on System Systems (SoS) which anticipates the formalization of requirements, aiming to understand, analyze and design SG within the scope of Model Based Systems Engineering (MBSE). The definition of a microgrid from the SoS perspective is presented in order to provide a complete view of its life cycle. Requirements would be represented in an Objective Oriented Requirements Engineering (GORE) approach, specifically using visual diagrams based on the Keep All Objectives Satisfied (KAOS) method, where network operation and control will be formally represented. A case study for small communities in the equatorial Amazon forest is used as a case study for the proposed method.
Palavras Chaves	Smart Grid; System Design; MBSE, Requeriments Engineering; KAOS.
Evento/Periódico	SBSE-2020. VIII Simpósio Brasileiro de Sistemas Elétricos.
Ano	2020
DOI	10.48011/sbse.v1i1.2373
Autores	Postigo, M.A; Silva, J.R.

Tabela 13 – Journal Brazilian Applied Science Review

Título	**Applying goal-oriented requirement engineering and reference architecture for microgrid systems**
Abstract	This paper presents a method for specifying requirements, applied to the operation of a microgrid. The challenge is to model Smart Grid (SG) systems requirements using formal methods based on schematic representations that will allows analysis, verification and validation, improving the reliability and performance of the design cycle. Considering the inherent complexity of these systems - generally heterogeneous, open and distributed - formal modeling is a key issue for the design of automated electrical systems to fit user experience expectations. In this article we propose a systemic approach that combines consolidated reference architectures and modern goal-oriented requirements engineering methods in the early steps of microgrid design. The formalism proposed is intended to obtain formal requirements models associated with standards and reference models proposed and accepted by the scientific and practitioner community.
Palavras Chaves	Requirements Engineering, KAOS, IEC 61850, Microgrid.
Evento/Periódico	Journal Brazilian Applied Science Review
Ano	2020
DOI	10.34115/basrv4n3-014
Autores	Postigo, M.A; Silva, J.M; Silva, J.R.

Tabela 14 – SBAI-2019

Título	**FORMAL REQUIREMENTS FOR MICROGRID USING KAOS AND REFERENCE ARCHITECTURE**
Abstract	This paper presents a method for specifying requirements, applied to the operation of a microgrid. The challenge is to model Smart Grid (SG) systems requirements using formal methods based on schematic representations that will allows analysis, verification and validation, improving the reliability and performance of the design cycle. Considering the inherent complexity of these systems - generally heterogeneous, open and distributed - formal modeling is a key issue for the design of automated electrical systems to fit user experience expectations. In this article we propose a systemic approach that combines consolidated reference architectures and modern goal-oriented requirements engineering methods in the early steps of microgrid design. The formalism proposed is intended to obtain formal requirements models associated with standards and reference models proposed and accepted by the scientific and practitioner community.
Palavras Chaves	Requirements Engineering, KAOS, IEC 61850, Microgrid.
Evento/Periódico	SBAI-2019. 14º Simpósio Brasileiro de Automação Inteligente
Ano	2019
DOI	10.17648/sbai-2019-111155
Autores	Postigo, M.A; Silva, J.M; Silva, J.R.

Tabela 15 – CBA-2018

Título	**FORMAL REQUIREMENTS SPECIFICATION FOR MICROGRID BASED IN ARCHITECTURE IEC61850**
Abstract	Smart Grid (SG) systems are considered a sustainable alternative to power supply problems all over the world. SG systems are open, distributed and heterogeneous, what raises a huge demand over its design process. At the present, SG projects are usually done by applying deductive methods, where requirement specifications do not reach a complete model. On the other hand, considering the importance of the requirements phase, it is necessary to provide methods, tools, and concepts to better characterize, understand and specify the application domain of SG systems. This work proposes the introduction of a formal requirements analysis in the SG life cycle that fits IEC61850 architecture. Requirements would be represented in a goal-oriented approach (GORE) specifically using visual diagrams based on KAOS method that will be transformed in an extended Petri Nets representation to analyze SG automated behavior. Practical results are shown through an example showing the application of GORE methodology and Petri Nets modeling with the corresponding analysis to fit specific requirements associated to microgrid for the Amazon region.
Palavras Chaves	Microgrids, Architecture IEC 61850, Petri Nets, Formal modeling, Gore methods.
Evento/Periódico	CBA-2018. Microgrids, Architecture IEC 61850, Petri Nets, Formal modeling, Gore methods.
Ano	2018
DOI	10.20906/CPS/CBA2018-0577
Autores	Postigo, M.A; Silva, J.M; Silva, J.R.

Tabela 16 – SBSE-2018

Título	**Modeling in Petri Nets for Micro Smart Grid Operation based on IEC 61850 Architecture**
Resumo	Smart Grid (SG) systems are considered a sustainable alternative to power supply problems all over the world. SG systems are open, distributed and heterogeneous, what raises a huge demand over its design process. At the present, SG projects are usually done by applying deductive methods, where requirement-specifications do not reach a complete model. This article presents a proposal for application of Systems Design methods in the development of a micro smart grid (MSG), through a case study, applied to the communities of the Amazon region. The methodology consists of using formal mechanisms already consolidated in system design, whose main objective is to analyze implementation problems and to have feedback. For the specification of requirements, was used to the IEC 61850 architecture, the behavior dynamics of the system was analyzed and verified using the Petri Nets (RdP), a tool that allowed to verify and identify possible contradictory states and undesired conditions. The results showed it was possible to develop a Petri Nets model based on the IEC 61850 architecture that allowed to model the automation of the MSG system operation, and to design a control solution for MSG, considering the particularity and constraints of the Amazon region.
Palavras Chaves	Petri Nets, Smart Grid, IEC 61850
Evento/Periódico	SBSE 2018 - VII Simpósio Brasileiro de Sistemas Elétricos. Niteroi RJ, Brazil.
Ano	2018
DOI	10.17648/sbai-2019-111155
Autores	Postigo, M.A; Silva, J.R.