



FERRAMENTA PARA INSTANCIACÃO E EXECUCÃO DO PROCESSO DE ANÁLISE DE DESEMPENHO DO AMBIENTE SPEAKER

Rodrigo Figueiredo Magalhães

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientadores: Ana Regina Cavalcanti da Rocha
Gleison dos Santos Souza

Rio de Janeiro
Março de 2015

FERRAMENTA PARA INSTANCIACÃO E EXECUCÃO DO PROCESSO DE
ANÁLISE DE DESEMPENHO DO AMBIENTE SPEAKER

Rodrigo Figueiredo Magalhães

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO
LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA
(COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE
EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof.^a Ana Regina Cavalcanti da Rocha, D.Sc.

Prof. Gleison dos Santos Souza, D.Sc.

Prof. Guilherme Horta Travassos, D.Sc.

Prof.^a Sheila dos Santos Reinehr, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

MARÇO DE 2015

Magalhães, Rodrigo Figueiredo

Ferramenta para Instanciação e Execução do Processo de Análise de Desempenho do Ambiente SPEAKER/ Rodrigo Figueiredo Magalhães. – Rio de Janeiro: UFRJ/COPPE, 2015.

XII, 116 p.: il.; 29,7 cm.

Orientadores: Ana Regina Cavalcanti da Rocha

Gleison dos Santos Souza

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2015.

Referências Bibliográficas: p. 110-116.

1. Análise de Desempenho de Processos de Software.
2. Instanciação e Execução de Linhas de Processos de Software e Componentes de Processo.
3. Engenharia de Software. I. Rocha, Ana Regina Cavalcanti da *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

*A toda minha família,
à Ana Regina Rocha
e ao Gleison dos Santos*

AGRADECIMENTOS

Agradeço primeiramente a Deus, pois sem ele eu não conseguiria realizar mais este sonho. Ele que sempre esteve presente em minha vida me guiando, e me ajudando em todos os momentos.

Agradeço também aos meus pais Grimaldo e Eduvirges, e minha irmã Rutiléia, que sempre me apoiaram nas horas difíceis, me ajudando a superar minhas dificuldades, sempre com muito amor e carinho.

Aos meus familiares, pelo apoio e carinho.

Aos amigos que conquistei durante este período e também aos que continuam ao meu lado me apoiando em todos os momentos.

À Taísa Guidini, por toda a ajuda durante esse processo e principalmente pela amizade e companheirismo que construímos ao longo do curso de Mestrado desta Universidade.

À minha orientadora, Professora Ana Regina Rocha, pela oportunidade de concretizar este sonho, pelos ensinamentos, aprendizado e pela sua amizade e dedicação.

Ao meu co-orientador, Professor Gleison Santos, pelo grande apoio no entendimento da ferramenta utilizada no desenvolvimento desta dissertação, pelas sugestões e incentivos que foram dados ao longo da elaboração deste trabalho.

À Professora Sheila dos Santos Reinehr e ao Professor Guilherme Horta Travassos por aceitarem participar da banca.

A todos aqueles que diretamente ou indiretamente me ajudaram a cumprir mais esta etapa da minha vida. Em especial ao amigo Thiago Silva de Souza, que além de meu orientador durante a graduação, foi uma das pessoas que sempre me apoiaram e me incentivaram a entrar no mestrado.

À Natalia Schots, pelas contribuições valiosas e pela amizade.

Às funcionárias e aos funcionários do PESC, por sua colaboração nos procedimentos administrativos.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

FERRAMENTA PARA INSTANCIÇÃO E EXECUÇÃO DO PROCESSO DE ANÁLISE DE DESEMPENHO DO AMBIENTE SPEAKER

Rodrigo Figueiredo Magalhães

Março/2015

Orientadores: Ana Regina Cavalcanti da Rocha
Gleison dos Santos Souza

Programa: Engenharia de Sistemas e Computação

As organizações buscam a cada dia mais a melhoria contínua dos seus produtos, e isto também é uma realidade para as organizações de software. Uma das formas para se alcançar este objetivo é o uso de técnicas e métodos para realizar a análise de desempenho de seus processos. A implementação da Análise de Desempenho de processos não é uma tarefa trivial, dependendo de diversos fatores para sua realização. Por este motivo, é importante o apoio para auxiliar a sua execução.

O objetivo desta dissertação é apoiar a instanciação e a execução de processos para análise de desempenho de processos de software. Para isso, foi construída uma ferramenta integrada ao ambiente SPEAKER que permite a execução das atividades relacionadas ao processo de análise de desempenho a partir de linhas processos definidas neste ambiente. Para avaliar o uso da ferramenta foi simulada a execução de uma linha de processo de software a partir de um exemplo de uso encontrado na literatura.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

A TOOL TO INSTANTIATION AND EXECUTION OF THE PROCESS
PERFORMANCE ANALYSIS ON SPEAKER ENVIRONMENT

Rodrigo Figueiredo Magalhães

March/2015

Advisors: Ana Regina Cavalcanti da Rocha

Gleison dos Santos Souza

Department: Systems and Computing Engineering

Organizations seeks for continuously improve on the quality of their products every day, and this is also a reality for software organizations. The use of methods and techniques to perform process performance analysis is one way to achieve this goal. However, software process performance analysis is not trivial task to implement on the organizations. For this reason, it is very important to have a tool support to help implement this methods and techniques to perform process performance analysis.

The goal of this dissertation is support the instantiation and execution of process performance analysis of software process activities. To do this, a tool was built and integrated to SPEAKER environment that allows the execution of the activities related with the process performance analysis using process lines. To evaluate the tool, an execution of the process line was simulated using data found on the literature.

ÍNDICE

| | |
|---|-----------|
| CAPÍTULO 1 - INTRODUÇÃO | 1 |
| 1.1 <i>Motivação</i> | 1 |
| 1.2 <i>Objetivos da Dissertação</i> | 4 |
| 1.3 <i>Metodologia</i> | 4 |
| 1.4 <i>Organização da Dissertação</i> | 5 |
| CAPÍTULO 2 - DESEMPENHO E EXECUÇÃO DE PROCESSOS DE SOFTWARE | 6 |
| 2.1 <i>Introdução</i> | 6 |
| 2.2 <i>Análise de Desempenho de Processos de Software</i> | 7 |
| 2.2.1 <i>Gráficos de Controle</i> | 13 |
| 2.2.1.1 <i>Gráficos de Controle para dados de variáveis</i> | 17 |
| 2.2.1.2 <i>Gráficos de Controle para dados de atributos</i> | 25 |
| 2.3 <i>Trabalhos Relacionados à Execução de Processos</i> | 29 |
| 2.3.1 <i>Uma Abordagem Flexível para Execução de Processos de Software</i> | 29 |
| 2.3.2 <i>Apoio ao Controle Estatístico de Processos de Software integrado a um ADS (WebAPSEE)</i> | 32 |
| 2.3.3 <i>Charon: uma Ferramenta para a Modelagem, Simulação, Execução e Acompanhamento de Processos de Software</i> | 34 |
| 2.3.4 <i>Uma Abordagem Dirigida por Modelo para Gerenciar e Customizar a Variabilidade em Processos de Software</i> | 35 |
| 2.4 <i>Considerações Finais</i> | 37 |
| CAPÍTULO 3 - O AMBIENTE SPEAKER | 38 |
| 3.1 <i>Introdução</i> | 38 |
| 3.2 <i>O Ambiente de Alta Maturidade - A2M</i> | 39 |
| 3.3 <i>O Ambiente SPEAKER</i> | 42 |
| 3.3.1 <i>Módulos do Ambiente SPEAKER</i> | 43 |
| 3.3.1.1 <i>Sistema Baseado em Conhecimento – SBC</i> | 43 |
| 3.3.1.2 <i>Biblioteca de Elementos de Processos Reutilizáveis para Análise de Desempenho</i> | 45 |
| 3.3.1.3 <i>Ferramenta de Instanciação e Execução do Processo</i> | 47 |
| 3.4 <i>Análise de Desempenho de Processos de Software no ambiente SPEAKER</i> | 48 |
| 3.5 <i>Considerações Finais</i> | 59 |
| CAPÍTULO 4 - FERRAMENTA PARA INSTANCIAÇÃO E EXECUÇÃO DO PROCESSO DE ANÁLISE DE DESEMPENHO – FIE..... | 60 |
| 4.1 <i>Introdução</i> | 60 |
| 4.2 <i>Requisitos para Definição da Ferramenta</i> | 61 |

| | | |
|--|---|------------|
| 4.3 | <i>Estrutura para Instanciação e Execução do Processo de Análise de Desempenho.....</i> | 62 |
| 4.3.1 | Principais Funcionalidades Implementadas..... | 67 |
| 4.3.2 | Interface com o Sistema Baseado em Conhecimento – SBC | 68 |
| 4.4 | <i>Definição do Processo de Análise de Desempenho.....</i> | 72 |
| 4.5 | <i>Considerações Finais</i> | 82 |
| CAPÍTULO 5 - EXEMPLO DE USO | | 84 |
| 5.1 | <i>Introdução</i> | 84 |
| 5.2 | <i>Execução do Processo</i> | 84 |
| 5.2.1 | Cenário 1..... | 84 |
| 5.2.2 | Cenário 2..... | 96 |
| 5.2.3 | Cenário 3..... | 101 |
| 5.3 | <i>Considerações Finais</i> | 105 |
| CAPÍTULO 6 - CONCLUSÃO | | 107 |
| 6.1 | <i>Considerações Finais</i> | 107 |
| 6.2 | <i>Contribuições.....</i> | 108 |
| 6.3 | <i>Limitações.....</i> | 108 |
| 6.4 | <i>Trabalhos Futuros</i> | 109 |
| REFERÊNCIAS..... | | 110 |

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1 - Ambiente SPEAKER – Visão Arquitetural..... | 3 |
| Figura 2 - Gráfico sequencial de Cobertura de Código (adaptado de FLORAC E CARLETON (1999))..... | 9 |
| Figura 3 - Diagramas de Pontos - Esforço por Tamanho de um produto (adaptado de FLORAC E CARLETON, 1999) | 10 |
| Figura 4 - Diagrama de Causa e Efeito (ROCHA <i>et al.</i> , 2012)..... | 10 |
| Figura 5 – Gráfico de Barras - Análise de Defeitos em Atividades de Software (adaptado de FLORAC e CARLETON (1999))..... | 11 |
| Figura 6 – Gráfico de Pareto - Defeitos encontrados em um produto nos dois primeiros anos de operação (adaptado de FLORAC e CARLETON (1999)) | 11 |
| Figura 7 - Layout básico de um gráfico de controle (ROCHA <i>et al.</i> , 2012)..... | 14 |
| Figura 8 - Testes de estabilidade (ROCHA <i>et al.</i> , 2012)..... | 15 |
| Figura 9 - Testes de estabilidade (GONÇALVES, 2014) | 16 |
| Figura 10 - Fórmulas para cálculo dos limites dos gráficos X-bar e R (ROCHA <i>et al.</i> , 2012)..... | 18 |
| Figura 11 - Gráfico X-bar e R (Adaptado de (FLORAC e CARLETON, 1999))..... | 19 |
| Figura 12 - Fórmulas para cálculo dos limites dos gráficos X-bar e S (ROCHA <i>et al.</i> , 2012)..... | 20 |
| Figura 13 - Gráfico X-bar e S (Adaptado de (FLORAC e CARLETON, 1999)) | 21 |
| Figura 14 - Fórmulas para cálculo dos limites dos gráficos XmR (ROCHA <i>et al.</i> , 2012) | 22 |
| Figura 15 - Gráfico XmR (Adaptado de (FLORAC e CARLETON, 1999))..... | 23 |
| Figura 16 - Fórmulas para cálculo dos limites dos gráficos XMmR (ROCHA <i>et al.</i> , 2012)..... | 24 |
| Figura 17 - Gráfico XMmR (Adaptado de (FLORAC e CARLETON, 1999)) | 24 |
| Figura 18 - Fórmulas para cálculo dos limites do gráfico C (ROCHA <i>et al.</i> , 2012)..... | 26 |
| Figura 19 - Gráfico C (Adaptado de (FLORAC e CARLETON, 1999)) | 26 |
| Figura 20 - Fórmulas para cálculo dos limites do gráfico U (ROCHA <i>et al.</i> , 2012)..... | 27 |
| Figura 21 - Gráfico U (Adaptado de (FLORAC e CARLETON, 1999))..... | 28 |
| Figura 22 - Fórmulas para conversão dos dados – Gráfico Z (ROCHA <i>et al.</i> , 2012) | 28 |
| Figura 23 - Gráfico Z (Rocha <i>et al.</i> , 2012)..... | 29 |
| Figura 24- Representação gráfica dos principais construtores da APSEE-PML (REIS,2003)..... | 30 |
| Figura 25 - Visão Geral do APSEE (REIS, 2003)..... | 31 |
| Figura 26 - Transição de Estados de uma atividade sendo executada (REIS, 2003) | 32 |
| Figura 27 - Ambiente de Modelagem de Processos (Murta <i>et al.</i> , 2002)..... | 35 |
| Figura 28 - Exemplo de Tela do A2M - Menu de Ferramentas | 40 |
| Figura 29 - Estratégia para definição de processos com reutilização (BARRETO, 2011) | 41 |
| Figura 30 - Ambiente SPEAKER – Visão em Camadas..... | 42 |
| Figura 31 - Atividades da Análise de Desempenho de Processos (SCHOTS <i>et al.</i> , 2014a)..... | 44 |
| Figura 32 - Linha de Processo - Verificar Estabilidade (GONÇALVES, 2014)..... | 49 |
| Figura 33 - Linha de Processo - Determinar Capacidade (GONÇALVES, 2014) | 50 |
| Figura 34 - Linha de Processo - Estabelecer Modelo de Desempenho | 51 |

| | |
|---|-----|
| Figura 35 - Análise Subprocesso, Análise Medida, Análise Linha de Processo e Versão Executada..... | 64 |
| Figura 36 - A2MComponenteProcesso - Alteração para cadastro de script..... | 64 |
| Figura 37 - Elemento de Processo Executável, Resultado Execução, Valor Parâmetro Script..... | 66 |
| Figura 38 - Estados de Execução de um Elemento de Processo Executável..... | 67 |
| Figura 39 - Diagrama de Sequência - <i>incluirElementoExecucao</i> | 70 |
| Figura 40 - Diagrama de Sequência - <i>alterarResultadoExecucao</i> | 71 |
| Figura 41 - Diagrama de Sequência - <i>consultarResultadosExecucao</i> | 72 |
| Figura 42 - Processo para Definição do Processo de Análise de Desempenho..... | 74 |
| Figura 43 - Linha de processos para verificar a estabilidade do subprocesso (GONÇALVES, 2014)..... | 85 |
| Figura 44 - Tela da ferramenta FIE - Acompanhar Subprocessos, Medidas e Linhas de Processos Analisadas..... | 86 |
| Figura 45 - Tela da ferramenta FIE - Acompanhar Versões Executadas..... | 87 |
| Figura 46 - Tela da ferramenta FIE - Acompanhar Elementos de processo Executados..... | 87 |
| Figura 47 - Tela da ferramenta FIE - Acompanhar Elemento de Processo..... | 89 |
| Figura 48- Tela da ferramenta FIE - Cadastrar Resultado da Execução..... | 90 |
| Figura 49 - Tela da ferramenta FIE - Consultar Versão Executada..... | 91 |
| Figura 50 - Tela da ferramenta FIE - Script para geração dos resultados..... | 92 |
| Figura 51 – Resultado da Verificação da distribuição dos dados..... | 93 |
| Figura 52 - Gráfico de Distribuição Normal e Histograma..... | 94 |
| Figura 53 - Gráfico XmR gerado na execução do componente “Construir gráfico de controle”..... | 95 |
| Figura 54 - Versão final instanciada e executada do processo..... | 96 |
| Figura 55 - Linha de processos para determinar a capacidade do subprocesso (GONÇALVES, 2014)..... | 97 |
| Figura 56 - Tela da ferramenta FIE - Consultar Análises Realizadas - Linhas de Processo..... | 98 |
| Figura 57 - Tela da ferramenta FIE - Edição de Elemento de Processo Pai..... | 99 |
| Figura 58 – Resultado da capacidade do processo..... | 100 |
| Figura 59 - Processo Instanciado e Executado Cenário 2..... | 100 |
| Figura 60 - Linha de processo para estabelecer o modelo de desempenho do subprocesso (GONÇALVES, 2014)..... | 101 |
| Figura 61 - Tela da ferramenta FIE - Consultar Subprocessos Analisados..... | 102 |
| Figura 62 - Resultado da execução da atividade "Selecionar método para construção do modelo de desempenho"..... | 103 |
| Figura 63 - Modelo de Desempenho..... | 104 |
| Figura 64 - Processo final instanciado e executado..... | 104 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 - Constantes para cálculo dos limites de controle dos gráficos <i>X-bar</i> e <i>R</i> | 19 |
| Tabela 2 - Constantes para cálculo dos limites de controle dos gráficos <i>X-bar</i> e <i>S</i> | 21 |
| Tabela 3 - Atividades e Subatividades do Processo | 33 |
| Tabela 4 - Atividades e Tarefas da Análise de Desempenho de Processos (SCHOTS <i>et al.</i> , 2014a)..... | 44 |
| Tabela 5 - Elementos de Processo (GONÇALVES, 2014) | 46 |
| Tabela 6 - Elementos de processos definidos (SCHOTZ <i>et al.</i> , 2014a)..... | 47 |
| Tabela 7 - Descrição dos elementos gráficos da abordagem de BARRETO (2011)..... | 48 |
| Tabela 8 - Template utilizado para a definição dos componentes de processo (GONÇALVES, 2014) | 51 |
| Tabela 9 - Template utilizado para a definição das atividades (GONÇALVES, 2014) . | 51 |
| Tabela 10 - Componentes de Processo (GONÇALVES, 2014) | 52 |
| Tabela 11 - Atividade “Registrar a categorização dos dados da medida” (GONÇALVES, 2014) | 55 |
| Tabela 12 - Componente de processo concreto “Construir gráfico de controle <i>XmR</i> com Minitab” (GONÇALVES, 2014)..... | 55 |
| Tabela 13 - Parâmetros do método incluir elemento execução | 69 |
| Tabela 14 - Parâmetros do método atualizar resultado execução..... | 70 |
| Tabela 15 - Parâmetros do método consultar resultados execução | 71 |
| Tabela 16 - Matriz de Rastreabilidade dos requisitos da ferramenta FIE com os métodos de comunicação com o SBC | 72 |
| Tabela 17 - Valores coletados | 90 |

CAPÍTULO 1 - INTRODUÇÃO

Este capítulo apresenta as principais questões que motivaram a realização deste trabalho, o objetivo da pesquisa, a metodologia adotada e como esta dissertação encontra-se organizada.

1.1 Motivação

As organizações buscam cada vez mais a garantia da qualidade dos produtos e a melhoria do desempenho de seus processos. Um processo executado com qualidade tende a produzir produtos com mais qualidade (FUGGETA, 2000). O uso de técnicas e métodos para apoiar a análise de desempenho é a forma mais utilizada pelas organizações para atingir estes objetivos.

Os modelos de maturidade de processos de software MR-MPS-SW – Modelo de Referência MPS para Software (SOFTEX, 2012) e CMMI-DEV – *Capability Maturity Model Integration for Development* (CMMI Product Team, 2010), nos níveis de alta maturidade exigem que haja a análise de desempenho dos processos de software considerados mais críticos pela organização. Para o MR-MPS-SW esse resultado é requerido a partir do nível B, através dos Atributos de Processo (AP) 4.1 e 4.2, e para o CMMI-DEV com a área de processo “Desempenho do Processo Organizacional” (OPP), cuja implementação é exigida a partir do nível 4.

A implementação da análise de desempenho de processos não é uma tarefa trivial, depende de diversos fatores, entre eles, de um profissional que tenha conhecimento de estatística e outro com conhecimento dos processos da organização, dificultando sua implementação e aumentando o custo. Outros fatores que podem dificultar a execução da análise de desempenho são: a falta de um procedimento para planejamento e coleta de medidas adequadas; a falta de conhecimento sobre as técnicas e métodos para realizar a análise de desempenho dos processos e a falta de conhecimento sobre os dados necessários para realizar uma análise adequada (FLORENCE, 2001; TARHAN e DEMIRÖRS, 2006; PAULK e HYDER, 2007; BORJA 2007; CARD, 2007; CARD *et al.*, 2008; MAHANTI e EVANS, 2012).

Com o objetivo de disponibilizar um ambiente que auxilie na execução das atividades necessárias para realizar a análise de desempenho de processos de software, o

grupo de Qualidade de Software do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ está desenvolvendo o ambiente SPEAKER (*Software Process PErformance Analysis Knowledge-based EnviRonment*).

Este ambiente é composto por um corpo de conhecimento disponibilizado através de um Sistema Baseado em Conhecimento (SBC) sobre os conceitos, atividades e técnicas de análise de desempenho de processos para apoiar as organizações na execução adequada da análise de desempenho de seus processos. Além do sistema baseado em conhecimento, o ambiente SPEAKER também possui linhas de processo de software para apoiar a execução das atividades do processo de análise de desempenho e, por último, uma ferramenta para apoiar a instanciação e execução do processo de análise de desempenho através do conhecimento disponibilizado no SBC.

Os seguintes requisitos foram definidos para o ambiente SPEAKER (SCHOTS *et al.*, 2013):

- REQ01 – O ambiente deve prover o conhecimento necessário para realizar a análise de desempenho de processos de software, guiando o responsável em todas as atividades a serem realizadas;
- REQ02 – O ambiente deve apoiar a execução de todas as atividades previstas para realizar a análise de desempenho de processos de software, a saber: identificação dos objetivos quantitativos organizacionais; identificação dos subprocessos críticos; avaliação da adequabilidade das medidas para análise de desempenho; verificação da estabilidade; verificação da capacidade; e estabelecimento do modelo de desempenho;
- REQ03 – O ambiente deve gerenciar a execução das atividades da análise de desempenho de processos com o apoio da gerência do conhecimento;
- REQ04 – O ambiente deve armazenar os resultados de execução das atividades da análise de desempenho de processos, a fim de permitir a execução das próximas atividades adequadas a cada situação;
- REQ05 – O ambiente deve ser aderente aos níveis de maturidade B do MR-MPS-SW e 4 do CMMI-DEV e, também ao nível 4 de capacidade da ISO/IEC 15504.

Para atender a estes requisitos, o ambiente SPEAKER possui três ferramentas principais. São elas: (i) Um sistema baseado em conhecimento (SBC) que guia o responsável durante a execução da análise de desempenho, apresentando os conhecimentos necessários para cada atividade, e que mantém o corpo do conhecimento do ambiente, e; (ii) uma ferramenta para instanciação e execução do processo de análise de desempenho (FIE) que permite o controle das atividades executadas e o armazenamento dos resultados obtidos (SCHOTZ, 2013; SCHOTZ *et al.*, 2013; SCHOTZ *et al.* 2014a), e; (iii) uma ferramenta para apoiar a definição de processos baseada em uma abordagem para reutilização e para definição de processos com reutilização (BARRETO, 2011). A Figura 1 apresenta uma visão arquitetural do ambiente SPEAKER onde se pode identificar as principais ferramentas.

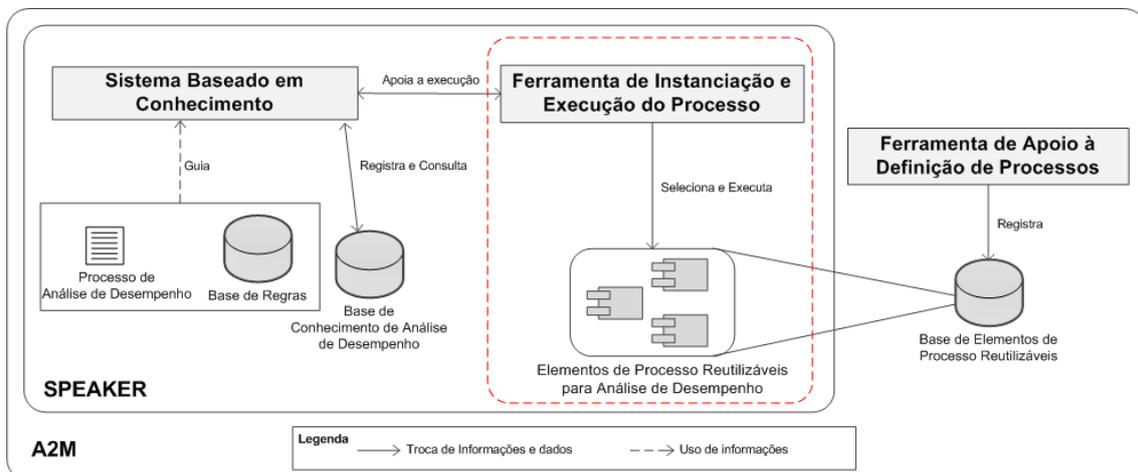


Figura 1 - Ambiente SPEAKER – Visão Arquitetural

A motivação para o desenvolvimento desta dissertação surgiu da necessidade do ambiente SPEAKER de controlar a instanciação e a execução das atividades do processo de análise de desempenho de acordo com informações providas por um sistema baseado em conhecimento.

A definição do ambiente SPEAKER ocorreu durante as reuniões do grupo de Qualidade de Software do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ pelos participantes envolvidos neste processo. Os objetivos e requisitos gerais do ambiente SPEAKER foram definidos por todos os envolvidos (SCHOTZ, 2013; SCHOTZ *et al.*, 2013; SCHOTZ *et al.* 2014a), e a partir destes, cada componente

do grupo definiu os objetivos e requisitos específicos para o módulo que lhe foi atribuído. Esses objetivos e requisitos são necessários para o funcionamento do ambiente SPEAKER.

1.2 Objetivos da Dissertação

O objetivo desta dissertação, no contexto do ambiente SPEAKER, é prover o apoio ferramental para a instanciação e execução de processos para análise de desempenho de processos de software a partir de linhas de processo e de uma base de elementos de processo reutilizáveis. Este apoio ferramental é apresentado na Figura 1 e destacado no retângulo cuja linha é pontilhada.

Outro objetivo da dissertação é atender aos requisitos REQ02 e REQ03 do ambiente SPEAKER referentes ao apoio e ao gerenciamento da execução do processo de análise de desempenho.

As linhas de processo que são executadas pela ferramenta FIE, objetivo principal desta dissertação, foram definidas por GONÇALVES (2014).

A instanciação do processo é direcionada por informações geradas pela execução de um sistema baseado em conhecimento. Este sistema encontra-se em desenvolvimento por uma tese de doutorado (SCHOTS, 2013; SCHOTS *et al.*, 2013; SCHOTS *et al.*, 2014a).

1.3 Metodologia

Para o desenvolvimento deste trabalho foram realizadas as seguintes etapas para condução de uma pesquisa (WAZLAWICK, 2009): i) definição do tema de pesquisa; ii) revisão da literatura; iii) definição do objetivo da pesquisa; iv) elaboração da solução proposta; v) avaliação da solução proposta.

As quatro primeiras etapas foram realizadas de forma iterativa, de modo que a definição do tema de pesquisa, que é análise de desempenho de processos de software, surgiu a partir das discussões realizadas ao longo das disciplinas de mestrado que foram cursadas e das discussões e experiências do Grupo de Qualidade de Software da COPPE/UFRJ.

A revisão da literatura foi realizada a partir de livros específicos sobre Análise de Desempenho de Processos (WHEELER e CHAMBERS, 1992; FLORAC e

CARLETON, 1999; WHEELER, 2008; ROCHA *et al.*, 2012), artigos, dissertações de mestrado e teses de doutorado relacionados ao tema Análise de Desempenho de Processos e execução ou simulação de processos. Esta revisão foi realizada de maneira Ad hoc.

A partir do resultado destas duas primeiras etapas, foi definido o objetivo da pesquisa no contexto do ambiente SPEAKER.

A definição da solução proposta específica e objeto desta dissertação, foi realizada considerando a integração com a tese de doutorado responsável pelo desenvolvimento do SBC (SCHOTZ *et al.*, 2014a) e a dissertação de mestrado responsável pela definição das linhas de processo e componentes de processo que apoiam a execução da análise de desempenho (GONÇALVEZ, 2014).

Para avaliar a ferramenta desenvolvida, os exemplos de uso definidos por GONÇALVES (2014) foram adaptados para demonstrar a utilização da ferramenta FIE.

1.4 Organização da Dissertação

Esta dissertação está organizada em 6 capítulos. O presente capítulo apresentou a motivação para o desenvolvimento deste trabalho, os objetivos da pesquisa e a metodologia utilizada.

O capítulo 2 apresenta os conceitos relacionados à análise de desempenho de processos e execução ou simulação de processos, além da aplicação destes conceitos na área de software. Também apresenta trabalhos relacionados ao tema desta dissertação.

O capítulo 3 descreve o ambiente SPEAKER através de seus principais módulos e sua infraestrutura e como o ambiente SPEAKER está inserido em um ambiente mais amplo, o ambiente de Alta Maturidade (A2M). Este ambiente também é descrito neste capítulo.

O capítulo 4 descreve as funcionalidades da ferramenta FIE para instanciação e execução do processo de análise de desempenho para processos de software, que é o principal objetivo desta dissertação.

O capítulo 5 apresenta um exemplo de uso realizado para avaliar a utilização da ferramenta FIE.

O capítulo 6 apresenta as considerações finais deste trabalho, as contribuições da dissertação, algumas limitações identificadas e os trabalhos futuros.

CAPÍTULO 2 - DESEMPENHO E EXECUÇÃO DE PROCESSOS DE SOFTWARE

Este capítulo apresenta os conceitos relacionados à análise de desempenho e execução ou simulação de processos e a aplicação destes conceitos na área de software. Também apresenta trabalhos relacionados à aplicação destes conceitos.

2.1 Introdução

As organizações buscam a melhoria contínua dos resultados dos seus processos executados. FLORAC e CARLETON (1999) apresentam algumas motivações para auxiliar esta busca. Essas questões estão relacionadas com o alcance dos objetivos de negócios da organização, a satisfação do cliente quanto ao produto ou serviço entregue, o retorno do investimento, a redução de custos na produção do produto ou serviço, a melhora na resposta às necessidades dos clientes ou melhora das funcionalidades dos produtos, a melhora da posição competitiva e o crescimento necessário para sobrevivência da organização.

A fim de apoiar a busca contínua da qualidade dos processos, FLORAC e CARLETON (1999) citam a abordagem de Deming, que determina algumas ações para que as organizações sejam competitivas, melhorem a qualidade dos produtos e aumentem sua produtividade. As ações estão relacionadas com o foco nos processos que geram os produtos e serviços para melhorar a qualidade e a produtividade, em assegurar que o processo possui o apoio necessário para sua execução, em corrigir os problemas que existam no comportamento do processo e não culpar as pessoas envolvidas, em reconhecer que existem variações em todos os processos e encará-las como uma oportunidade de melhoria, ou seja, a redução dessas variações acarretam em melhorias no processo. E, por último, utilizar as informações das variações, quando necessário, no processo de tomada de decisões.

Sobre a realização da análise de desempenho de processos de software, podemos citar os modelos de maturidade CMMI-DEV (CMMI Product Team, 2010) e o MR-MPS-SW (SOFTEX, 2012) que em seus altos níveis (4 e 5 no caso do CMMI-DEV e B e A no caso do MR-MPS-SW) recomendam a análise de desempenho dos processos de

software com o objetivo de conhecer seu comportamento. A partir deste conhecimento é possível determinar a capacidade do processo de atender aos objetivos de desempenho e qualidade estabelecidos pela organização e por projetos específicos. O controle estatístico de processos foi proposto inicialmente para a área de manufatura visando apoiar a implementação de programas de melhoria contínua em linhas de produção e, posteriormente, adaptado para a área de software (ROCHA *et al.*, 2012).

O objetivo deste capítulo é apresentar os conceitos relacionados à análise de desempenho e à execução de processos de software, os diferentes tipos gráficos de controle utilizados como parte da análise dos dados, além de trabalhos relacionados a estes temas.

A seção 2.2 apresenta os conceitos relacionados à análise de desempenho de processos de software e os gráficos de controle que são utilizados para avaliação do comportamento do processo para dados variáveis e para dados de atributos. A seção 2.3 apresenta trabalhos relacionados à execução e simulação de processos de software encontrados na literatura.

2.2 Análise de Desempenho de Processos de Software

A análise de desempenho de processos surgiu na década de 1920 (WHEELER e CHAMBERS, 1992) na área da manufatura com o objetivo de quantificar e analisar o comportamento de um processo. Através dessa análise é possível identificar mudanças e tendências ocorridas ao longo do tempo no comportamento do processo da organização (ROCHA *et al.*, 2012).

As organizações realizam comparações entre o valor atual e outro valor para tentar medir seus processos. O uso desse tipo de relatório gerencial não apresenta uma visão correta do que se passou com o processo analisado, fazendo com que essas comparações sejam limitadas (WHEELER, 2008).

FLORAC e CARLETON (1999) definiram algumas atividades necessárias para medir o comportamento do processo, são elas: (i) esclarecer os objetivos de negócio; (ii) identificar e priorizar os problemas; (iii) identificar e definir medidas; (iv) coletar, verificar e armazenar os dados; (v) analisar o comportamento do processo, e; (vi) avaliar o desempenho do processo.

A avaliação do desempenho do processo é a determinação da estabilidade e da capacidade do processo. Um processo estável é aquele que possui um comportamento repetível, ou seja, previsível. Um processo é dito capaz se ele, além de ser estável, atender aos limites estabelecidos como objetivos de qualidade e metas da organização para sua execução (WHEELER e CHAMBERS, 1992; ROCHA *et al.*, 2012).

Ao aplicar as técnicas de análise de desempenho é possível verificar se o processo possui variações que caracterizam o seu comportamento. Essas variações podem ser rotineiras (devido ao fenômeno natural e inerente ao processo), e são denominadas ruídos ou variações excepcionais, quando são devido a outros fatores. Em um processo estável, todas as variações são causadas por fatores inerentes ao processo. A presença da variação excepcional é um sinal de que há relações de causa e efeito dominantes que afetam o processo e que a execução do processo não está sob controle, e que o processo não é estável. Neste caso, as causas excepcionais devem ser analisadas e tratadas (FLORAC e CARLETON, 1999; WHEELER, 2008).

Um processo pode ser estável e não ser capaz, isto porque a capacidade de um processo está relacionada com o bom desempenho do processo, ou seja, um processo é dito capaz quando consegue atingir os objetivos de qualidade e desempenho da organização e atender aos requisitos do cliente para um projeto. O desempenho do processo é chamado de voz do processo e os requisitos do cliente são a voz do cliente. Em um processo capaz, a voz do processo é capaz de atender a voz do cliente (WHEELER, 2008).

Uma análise inicial dos dados deve ser realizada para que se entenda as informações que foram obtidas sobre o processo que será avaliado. WHEELER (2008) utiliza dois tipos de gráficos para realizar esta análise, são os *gráficos sequenciais* e o *diagrama de pontos (ou histograma)*. Os *gráficos sequenciais*, também conhecidos como gráficos de séries temporais, possuem os dias, meses ou anos no eixo horizontal, e os valores no eixo vertical. As mudanças em um gráfico sequencial devem ser vistas da esquerda para a direita. A Figura 2 apresenta um exemplo de gráfico sequencial adaptado de FLORAC e CARLETON (1999) sobre cobertura de código em produtos de uma fábrica de software.

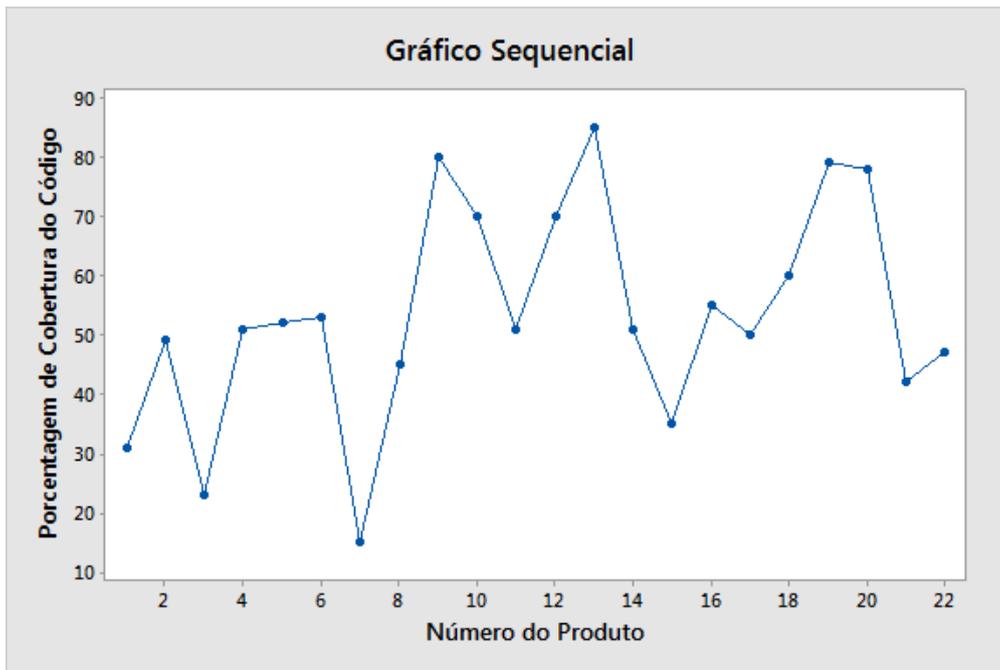


Figura 2 - Gráfico sequencial de Cobertura de Código (adaptado de FLORAC E CARLETON (1999))

O *diagrama de pontos* ou *gráfico de dispersão* é uma acumulação dos diferentes valores, quando eles ocorrem, sem tentar expor a sequência de tempo. Os valores estão escritos no eixo horizontal e o eixo vertical representa a frequência para os diferentes valores observados. A Figura 3 apresenta um exemplo de um diagrama de pontos sobre o esforço gasto com desenvolvimento de um produto em relação ao seu tamanho, que é representado pela quantidade de linhas de código executadas. Este exemplo foi adaptado de FLORAC E CARLETON (1999).

Além dos *gráficos sequenciais* e o *diagrama de pontos*, FLORAC e CARLETON (1999) sugerem o uso do *diagrama de causa e efeito*, do *gráfico de barras*, e do *gráfico de pareto*. O *diagrama de causa e efeito*, também conhecido como *diagrama de Ishikawa* permite que o usuário examine, mapeie e priorize uma série de fatores que afetam um processo em particular, um problema ou qualquer outro resultado. A Figura 4 apresenta um exemplo de um digrama de causa e efeito (ROCHA *et al.*, 2012) que apresenta as razões que levam ao retrabalho em subprocessos de uma organização específica. Foram identificadas quatro causas de retrabalho.

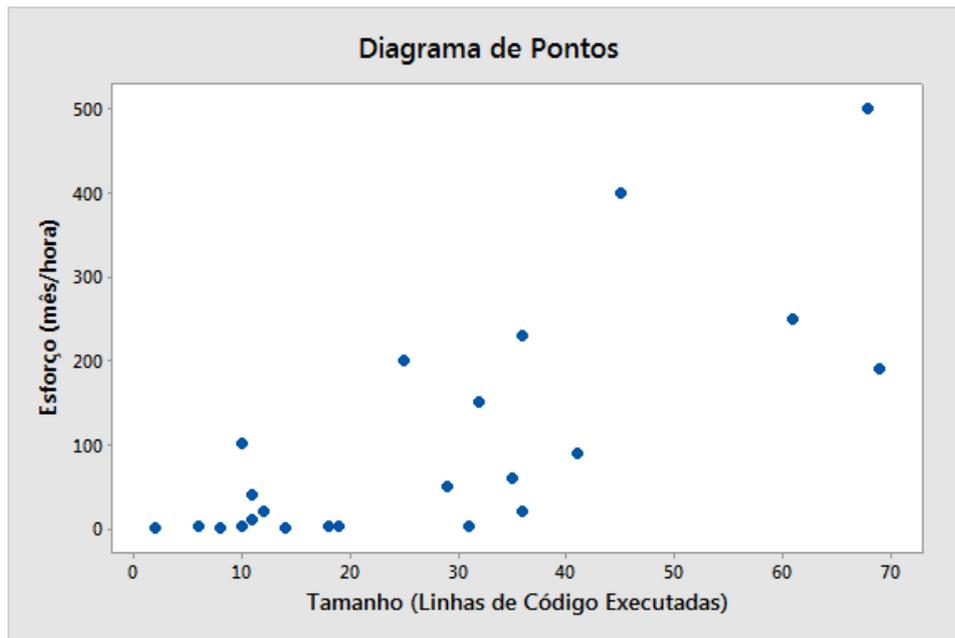


Figura 3 - Diagramas de Pontos - Esforço por Tamanho de um produto (adaptado de FLORAC E CARLETON, 1999)

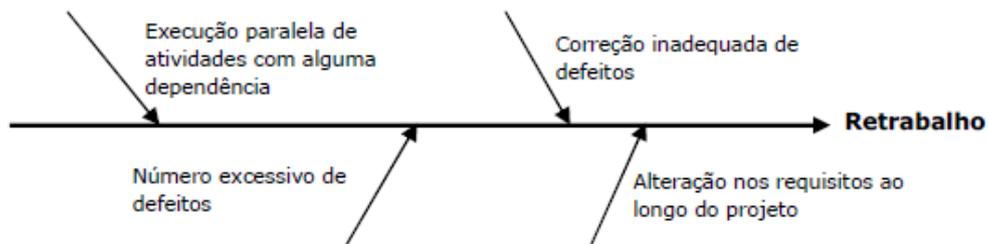


Figura 4 - Diagrama de Causa e Efeito (ROCHA *et al.*, 2012)

O *gráfico de barra* é utilizado para investigar o formato de um grupo de dados. Este tipo de gráfico é bem parecido com o *histograma*, porém não precisa ser baseado em medidas de variáveis contínuas ou contagem de frequência. Por último, há o *gráfico de pareto* que é um gráfico para ordenar as causas alternativas ou outros resultados e auxiliar a determinar quais ações ou oportunidades de melhorias devem ser priorizadas (FLORAC e CARLETON, 1999). A Figura 5 e a Figura 6 apresentam um exemplo do *gráfico de barra* e *gráfico de pareto* adaptados de FLORAC e CARLETON (1999). A Figura 5 apresenta uma análise realizada a partir do percentual de defeitos encontrados em atividades relacionadas ao desenvolvimento de softwares. A partir deste gráfico é possível identificar que a fase de codificação possui o maior percentual de defeitos em

relação as outras fases de desenvolvimento de um software. A Figura 6 apresenta um exemplo do *gráfico de pareto* sobre defeitos encontrados em um produto durante seus dois primeiros anos de operação. O tipo que ocasionou mais erros durante este período foi o *Erro de Sintaxe*, de acordo com o gráfico apresentado.

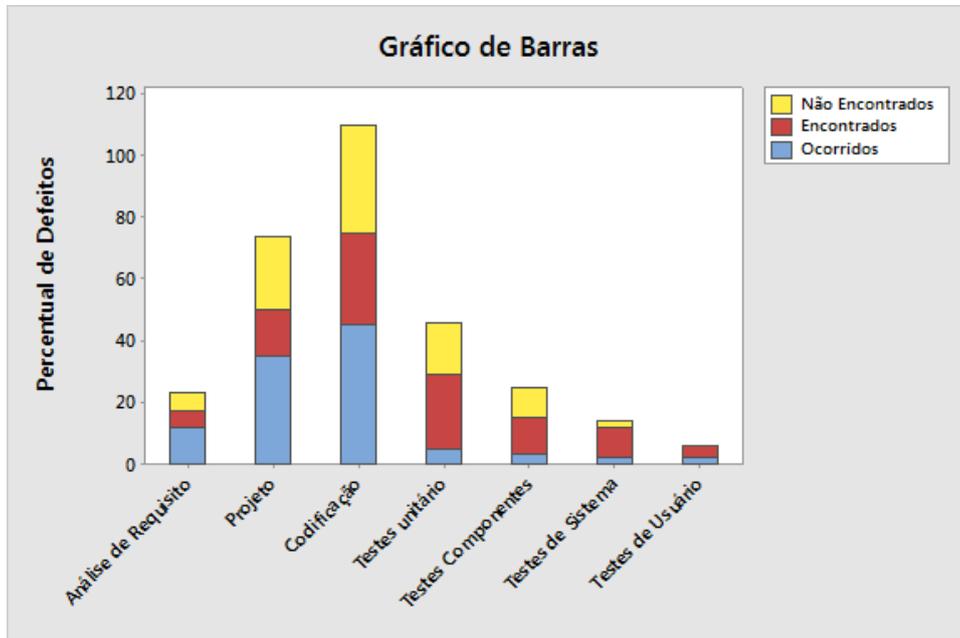


Figura 5 – Gráfico de Barras - Análise de Defeitos em Atividades de Software (adaptado de FLORAC e CARLETON (1999))

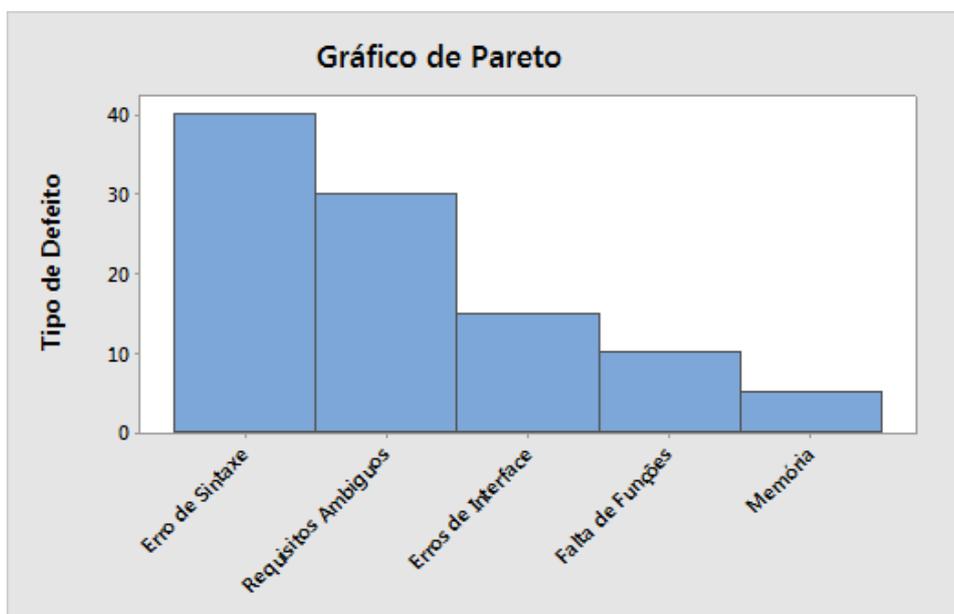


Figura 6 – Gráfico de Pareto - Defeitos encontrados em um produto nos dois primeiros anos de operação (adaptado de FLORAC e CARLETON (1999))

Para auxiliar a exposição dos dados analisados, WHEELER (2008) utiliza duas regras definidas por Shewhart¹, são elas: (i) Os dados devem sempre ser apresentados de maneira a preservar a evidência nos dados quanto a todos prognósticos que possam ser feitos a partir desses dados; (ii) quando quer que uma média, amplitude ou um histograma sejam usados para resumir dados, o resumo não deve iludir o usuário e levá-lo a tomar alguma atitude, que não tomaria se os dados fossem apresentados em um gráfico sequencial.

Baseado nas regras definidas por Shewhart, WHEELER (2008) define dois princípios que auxiliam o entendimento e a análise dos dados. O primeiro princípio afirma que: *“Nenhum dado tem sentido fora de seu contexto”*. A apresentação dos dados em gráficos torna seu entendimento mais acessível para a mente humana. O segundo princípio afirma que: *“Embora todo o conjunto de dados contenha ruído, alguns conjuntos de dados podem conter sinais. Portanto, antes que você consiga detectar um sinal dentro de determinado conjunto de dados, você deve primeiro filtrar o ruído”*.

Somente o uso destes gráficos e princípios não são capazes de alcançar o objetivo da análise de desempenho de processos. Para isso, um método estatístico deve ser utilizado objetivando a análise e permitindo que todos que o analisarem tenham a mesma interpretação. O método mais utilizado é a geração de um gráfico de controle (ou gráfico de comportamento do processo). Através do gráfico de controle é possível verificar se o processo possui variações rotineiras ou excepcionais através dos dados representados de forma sequencial em um gráfico e do cálculo da linha central e dos limites naturais do processo obtidos a partir dos dados. Os diversos tipos de gráficos de controle serão detalhados na próxima seção.

Além do gráfico de controle, há o método de comparações para as especificações e o método de comparações para as médias (WHEELER, 2008). O método de comparação dos dados com as especificações veio de uma prática da manufatura de comparar as medidas de produto com os limites de especificações. Essa comparação define a posição do valor atual relativo a algum valor e resulta em informar se o valor atual é aceitável ou não de acordo com as especificações. As especificações representam a voz do cliente. Este tipo de abordagem informa onde o processo se

¹ Walter Andrew Shewhart foi um famoso físico, engenheiro e estatístico americano. Conhecido como o pai do controle estatístico da qualidade.

encontra no momento, porém, não é possível avaliar como foi a evolução até chegar a este ponto.

O método de comparações para as médias realiza a comparação dos dados com a média do processo. Assim como na especificação, este método também informa se o valor atual está acima ou abaixo da média. Porém, enquanto o método de especificação compara o valor atual com a voz do cliente, este método compara o valor atual com um valor a partir do processo em si mesmo (somente parte da voz do processo).

2.2.1 Gráficos de Controle

Os gráficos de controle, também conhecidos como gráfico de comportamento do processo, foram criados por Shewhart na área da manufatura com o objetivo de apresentar as variações no comportamento dos processos. O comportamento do processo é avaliado através de um gráfico de frequência que apresenta os dados em uma ordem temporal, a diferença de um gráfico de controle para um gráfico sequencial comum é que este possui limites de controle (superior e inferior) e uma linha central que é representada pela média (ou mediana) dos dados do processo (ROCHA *et al.*, 2012; FLORAC e CARLETON, 1999).

Segundo FLORAC e CARLETON (1999), os limites e a linha central representam estimativas calculadas a partir dos dados do processo. Eles não podem ser calculados de maneira arbitrária, pois devem revelar o que o processo realmente pode fazer, ou seja, seus próprios limites de desempenho (ROCHA *et al.*, 2012).

A representação gráfica de um gráfico de controle auxilia a categorização da extensão da variação rotineira, os limites de um gráfico de controle permitem diferenciar entre a variação rotineira e a variação excepcional. Os valores que se encontram fora dos limites são considerados variações excepcionais, ou causas especiais. Porém, podem existir causas especiais dentro dos limites de controle do processo. Existem métodos de análise que auxiliam a identificação dos pontos que, mesmo estando dentro do limite, merecem uma atenção especial (WHEELER e CHAMBERS, 1992; FLORAC e CARLETON, 1999).

Segundo WHEELER (2008), um erro comum quando se analisa os dados de uma medida é pensar que diferenças percentuais grandes indicam um sinal e que pequenas indicam a falta de um sinal. Na verdade, é preciso criar o gráfico de comportamento do processo para visualizar se este valor é ou não um sinal. O gráfico de

comportamento do processo filtra o ruído provável a fim de detectar os sinais potenciais em qualquer conjunto de dados.

O *layout* básico de um gráfico de controle é apresentado na Figura 7. A distância dos limites superior e inferior estão a ± 3 sigma (σ), onde o sigma é o desvio padrão dos valores plotados no gráfico (FLORAC e CARLETON, 1999).

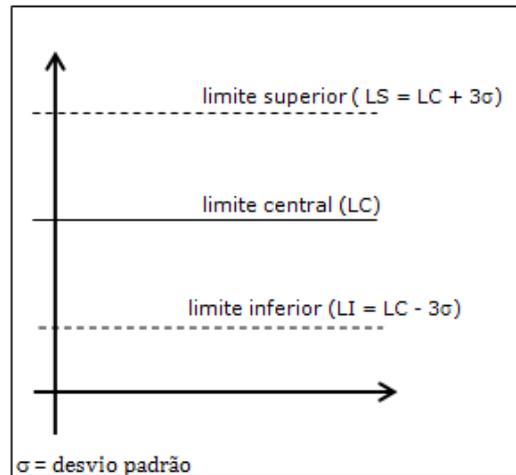


Figura 7 - Layout básico de um gráfico de controle (ROCHA *et al.*, 2012)

Existem diversos tipos de testes para se verificar a estabilidade do processo. Os quatro mais utilizados são (WHEELER e CHAMBERS, 1992; FLORAC e CARLETON, 1999):

- *Teste 1*: quando algum ponto encontra-se fora dos limites de controle de 3σ .
- *Teste 2*: quando pelo menos dois de três valores sucessivos encontram-se do mesmo lado e mais de 2σ da linha central.
- *Teste 3*: quando pelo menos quatro de cinco valores sucessivos encontram-se do mesmo lado e mais de 1σ da linha central.
- *Teste 4*: quando pelo menos oito valores sucessivos estiverem no mesmo lado da linha central.

Segundo FLORAC e CARLETON (1999) pode-se afirmar que um processo não é estável quando falhar em um dos testes, isto claro, se os quatro testes citados acima forem executados em conjunto. O uso efetivo dos quatro testes em conjunto também reduz o número de falsos alarmes. A Figura 8 apresenta um exemplo hipotético (ROCHA *et al.*, 2012) de um processo instável que descreve a execução dos quatro

testes de estabilidade. Para ilustrar as distâncias de até 3σ , foram definidas as zonas A, B e C que representam respectivamente a distância da linha central a 1σ , de 1σ a 2σ e de 2σ a 3σ .

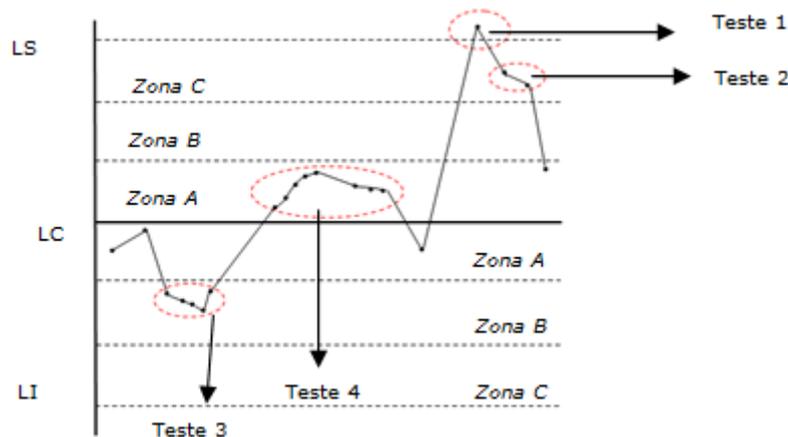


Figura 8 - Testes de estabilidade (ROCHA *et al.*, 2012)

Além dos principais testes de estabilidade apresentados anteriormente, existem outros testes que também podem ser executados para verificar a estabilidade de um processo (BALDASSARE *et al.*, 2010). Estes testes auxiliam na descoberta de padrões de oscilação e tendências, são eles :

- *Teste 5*: quando oito valores sucessivos estão do mesmo lado da linha central com nenhum ponto entre a linha central e 1σ .
- *Teste 6*: conhecido como teste de estratificação, ocorre quando quinze valores encontram-se entre a linha central e 1σ .
- *Teste 7*: conhecido como teste de tendência oscilatória, ocorre quando quatorze valores encontram-se em sequência alternadamente para cima e para baixo.
- *Teste 8*: conhecido como teste de tendência linear, acontece quando seis valores estão em uma sequência crescente ou decrescente.

A Figura 9 (GONÇALVES, 2014) apresenta um exemplo de aplicação para os oito testes de estabilidade apresentados, dividindo-os em três conjuntos: (i) Testes sigma; (ii) testes de limites, e; (iii) testes de tendência.

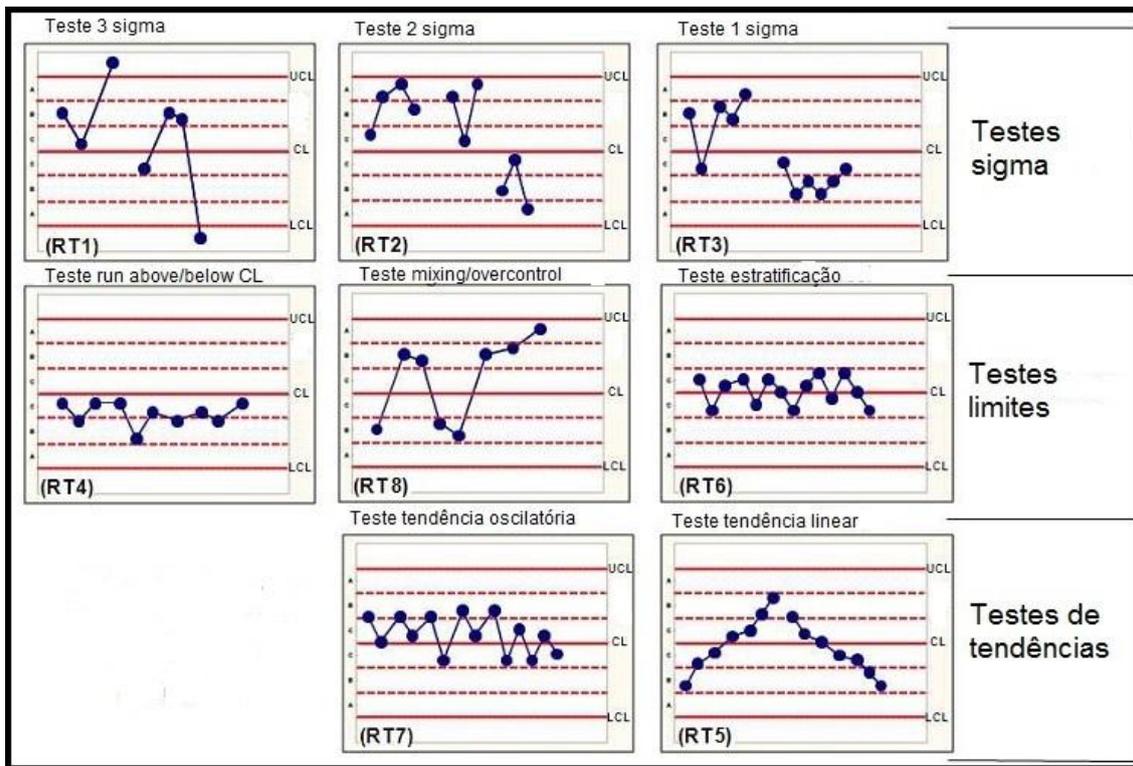


Figura 9 - Testes de estabilidade (GONÇALVES, 2014)

O uso dos gráficos de controle é o começo para se obter o conhecimento do comportamento processo, pois auxilia a formular as perguntas corretas e a não desperdiçar recursos para encontrar respostas em ruídos. Sempre é necessário olhar para o todo para não se pensar em melhorar uma parte do processo e acabar piorando outras.

Existem diversos tipos de gráfico de controle que podem ser aplicados a dois tipo de dados: *dados variáveis* e *dados de atributos*. Todos os tipos de gráficos de controle utilizam o *layout* apresentado na Figura 7. Os dados variáveis são medidas de fenômenos contínuos ou uma contagem que descreva um tamanho ou status e os dados de atributos referem-se a contagens discretas ou uma lista de características (WHEELER e CHAMBERS, 1992; FLORAC e CARLETON, 1999; ROCHA *et al.*, 2012). Para a escolha correta do tipo de gráfico de controle a ser utilizado, deve se levar em consideração, além do tipo de dados, o contexto em que os dados são coletados e utilizados.

As próximas duas seções apresentam os diferentes tipos de gráfico de controle de acordo com o seu tipo de dados analisados, apresentando sua utilização, fórmulas e procedimentos para geração dos gráficos.

2.2.1.1 Gráficos de Controle para dados de variáveis

Serão apresentados os seguintes tipos de gráficos para o tipo de dados variáveis:

- a) Gráfico de Controle de Média e Amplitude (*Average and Range Chart - X-bar e R*)
- b) Gráfico de Controle de Média e Desvio Padrão (*Average and Standard Deviation Chart - X-bar e S*)
- c) Gráfico de Controle do Valor Individual e Amplitudes Móveis (*Individuals and Moving Range Chart – XmR*)
- d) Gráfico de Controle do Valor Individual e Amplitude Móvel Mediana (*Individuals and Median Moving Range Chart – XMmR*)

a) Gráfico de Controle de Média e Amplitude (*Average and Range Chart - X-bar e R*)

O Gráfico de Controle de Média e Amplitude é adequado para analisar o comportamento do processo através de subgrupos², e combina os gráficos X-bar e R. O gráfico X-bar (*average*) analisa a média dos valores em cada subgrupo e o gráfico R (*range*) indica a variação (dispersão) entre os subgrupos. Este tipo de gráfico se limita a subgrupos formados por até 10 observações.

A análise do comportamento do processo é realizada através dos dois gráficos, para isso, os testes de estabilidade 1, 2, 3 e 4 apresentados anteriormente podem ser executados para verificar a estabilidade do processo. O gráfico X-bar permite analisar a tendência do processo ao longo do tempo, enquanto o gráfico R permite analisar a variação dentro de cada subgrupo.

A maneira como os dados estão organizados em subgrupos deve ser compatível com a estrutura presente nos dados. Isso significa que cada subgrupo será criado a partir de uma pequena região do espaço, ou tempo, ou produto, para garantir a homogeneidade dos subgrupos. Existem 6 princípios para auxiliar a formação de subgrupos (WHEELER e CHAMBERS, 1992):

² Subgrupos são formados a partir de medidas coletadas em um período curto de tempo sobre as mesmas condições.

- *1º Princípio*: Nunca defina subgrupos com informações diferentes. Cada subgrupo deve ser logicamente homogêneo.
- *2º Princípio*: Minimizar a variação dentro de cada subgrupo, com isso, cria-se um gráfico de controle sensível à presença de sinais.
- *3º Princípio*: Maximizar a oportunidade de variação entre os subgrupos. Se houver alguma possibilidade de duas coisas serem diferentes, tenha certeza de que elas estão em subgrupos diferentes.
- *4º Princípio*: Médias de ruído, não de sinais. Sempre será um erro calcular médias de sinais ou de potenciais sinais.
- *5º Princípio*: Trate o gráfico de acordo com a utilização dos dados. A frequência do subgrupo deve refletir a monitoração e a frequência da decisão apropriada para o processo, ou seja, se valores singulares são coletados, então subgrupos de tamanho 1 devem ser criados.
- *6º Princípio*: Estabelecer definições para o processo de amostragem. Procedimentos sistemáticos para obtenção de amostras e técnicas de definição de medidas são importantes fundações para dados úteis.

A Figura 10 apresenta as fórmulas para o cálculo dos limites dos gráficos X-bar e R.

| | |
|--|--|
| <p>X-bar</p> $UCL_{\bar{X}} = \bar{\bar{X}} + A_2 \bar{R}$ $CL_{\bar{X}} = \bar{\bar{X}}$ $LCL_{\bar{X}} = \bar{\bar{X}} - A_2 \bar{R}$ | <p><i>Onde:</i></p> <p>$\bar{\bar{X}}$ deve ser calculado para cada subgrupo de tamanho n, para cada um dos k subgrupos.</p> $\bar{X}_k = \frac{X_1 + X_2 + \dots + X_n}{n} \quad \text{e} \quad \bar{\bar{X}} = \frac{\bar{X}_1 + \bar{X}_2 + \dots + \bar{X}_k}{k}$ |
| <p>R</p> $UCL_{\bar{R}} = D_4 \bar{R}$ $CL_{\bar{R}} = \bar{R}$ $LCL_{\bar{R}} = D_3 \bar{R}$ | <p><i>Onde:</i></p> <p>R deve ser calculado para cada subgrupo de tamanho n, para cada um dos k subgrupos.</p> $R_k = X_{\max} - X_{\min} \quad \text{e} \quad \bar{R} = \frac{R_1 + R_2 + \dots + R_k}{k}$ |

Figura 10 - Fórmulas para cálculo dos limites dos gráficos X-bar e R (ROCHA *et al.*, 2012)

Os valores das constantes A_2 , D_3 e D_4 presentes nas fórmulas variam de acordo com o tamanho dos subgrupos (n). Esses valores são apresentados na Tabela 1.

Tabela 1 - Constantes para cálculo dos limites de controle dos gráficos *X-bar* e *R*.

| <i>n</i> | A_2 | D_3 | D_4 |
|----------|-------|-------|-------|
| 2 | 1,880 | - | 3,268 |
| 3 | 1,023 | - | 2,574 |
| 4 | 0,729 | - | 2,282 |
| 5 | 0,577 | - | 2,114 |
| 6 | 0,483 | - | 2,004 |
| 7 | 0,719 | 0,076 | 1,924 |
| 8 | 0,373 | 0,136 | 1,864 |
| 9 | 0,337 | 0,184 | 1,816 |
| 10 | 0,308 | 0,223 | 1,777 |

A Figura 11 apresenta um exemplo dos gráficos *X-bar* e *R* gerados a partir da ferramenta Minitab (Minitab Inc., 2013), utilizando valores de um exemplo de FLORAC e CARLETON (1999) sobre esforço diário no desenvolvimento de um produto durante 16 semanas.

Os gráficos exibidos na Figura 11 são respectivamente o *X-bar* e o *R*. A partir das informações exibidas nos gráficos, pode se concluir que o subprocesso analisado é estável, pois nenhum dos testes de estabilidade executados falharam.

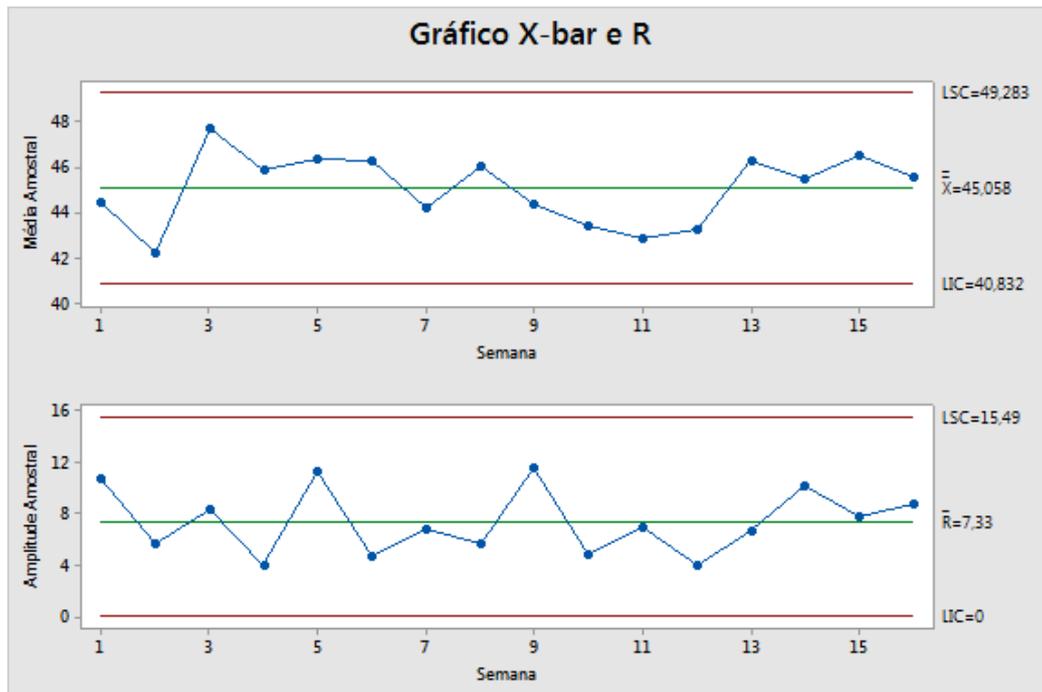


Figura 11 - Gráfico *X-bar* e *R* (Adaptado de (FLORAC e CARLETON, 1999))

b) Gráfico de Controle de Média e Desvio Padrão (Average and Standard Deviation Chart - X-bar e S)

O Gráfico de Controle de Média e Desvio Padrão é adequado para analisar o comportamento do processo nas mesmas condições que o gráfico X-bar e R, porém permite analisar subgrupos com mais de 10 observações. O gráfico X-bar (average) desta vez, analisa o desvio padrão dos valores em cada subgrupo, enquanto o gráfico S (range) analisa a variação interna dos subgrupos. Este gráfico pode ser utilizado por subgrupos que contenham menos de 10 observações.

A análise do comportamento do processo é realizada através dos dois gráficos, para isso, os testes de estabilidade 1, 2, 3 e 4 apresentados anteriormente podem ser executados para verificar a estabilidade do processo.

A Figura 12 apresenta as fórmulas para o cálculo dos limites dos gráficos X-bar e S.

| | |
|--|---|
| <p>X-bar</p> $UCL_{\bar{X}} = \bar{\bar{X}} + A_3 \bar{S}$ $CL_{\bar{X}} = \bar{\bar{X}}$ $LCL_{\bar{X}} = \bar{\bar{X}} - A_3 \bar{S}$ | <p>Onde:</p> <p>$\bar{\bar{X}}$ deve ser calculado para cada subgrupo de tamanho n, para cada um dos k subgrupos.</p> $\bar{X}_k = \frac{X_1 + X_2 + \dots + X_n}{n} \quad \text{e} \quad \bar{\bar{X}}_k = \frac{\bar{X}_1 + \bar{X}_2 + \dots + \bar{X}_k}{k}$ |
| <p>R</p> $UCL_{\bar{S}} = B_4 \bar{S}$ $CL_{\bar{S}} = \bar{S}$ $LCL_{\bar{S}} = B_3 \bar{S}$ | <p>Onde:</p> <p>S é o desvio padrão que deve ser calculado para cada subgrupo de tamanho n.</p> $S = \sqrt{\frac{\sum_{i=1}^{i=n} (X_i - \bar{X})^2}{n-1}} \quad \text{e} \quad \bar{S} = \frac{\sum_{i=1}^{i=k} S_i}{k}$ |

Figura 12 - Fórmulas para cálculo dos limites dos gráficos X-bar e S (ROCHA *et al.*, 2012)

Os valores para as constantes A_3 , B_3 e B_4 presentes nas fórmulas variam de acordo com o tamanho dos subgrupos (n). Os valores das constantes são apresentados na Tabela 2 para $n \leq 15$. O procedimento para obter as constantes onde $n > 15$ encontram-se em FLORAC e CARLETON (1999).

A Figura 13 apresenta um exemplo do gráfico X-bar e S gerados a partir da ferramenta Minitab (Minitab Inc., 2013), utilizando valores de um exemplo de

FLORAC e CARLETON (1999) sobre defeitos encontrados em inspeções de códigos em 4 releases de um produto.

Os gráficos exibidos na Figura 13 são respectivamente o X-bar e o S. A partir das informações exibidas nos gráficos, pode se concluir que o subprocesso analisado não é estável, pois o primeiro ponto do gráfico X-bar encontra-se fora do limite superior, como podemos observar o ponto está representado por um quadrado e o número 1 é exibido, informando que o *Teste 1* falhou.

Tabela 2 - Constantes para cálculo dos limites de controle dos gráficos X-bar e S.

| n | A_3 | B_3 | B_4 |
|-----|-------|-------|-------|
| 2 | 2,659 | - | 3,267 |
| 3 | 1,954 | - | 2,568 |
| 4 | 1,628 | - | 2,266 |
| 5 | 1,427 | - | 2,089 |
| 6 | 1,287 | 0,030 | 1,970 |
| 7 | 1,182 | 0,118 | 1,882 |
| 8 | 1,099 | 0,185 | 1,815 |
| 9 | 1,032 | 0,239 | 1,761 |
| 10 | 0,975 | 0,284 | 1,716 |
| 11 | 0,927 | 0,322 | 1,678 |
| 12 | 0,886 | 0,354 | 1,646 |
| 13 | 0,850 | 0,382 | 1,619 |
| 14 | 0,817 | 0,407 | 1,593 |
| 15 | 0,789 | 0,428 | 1,572 |

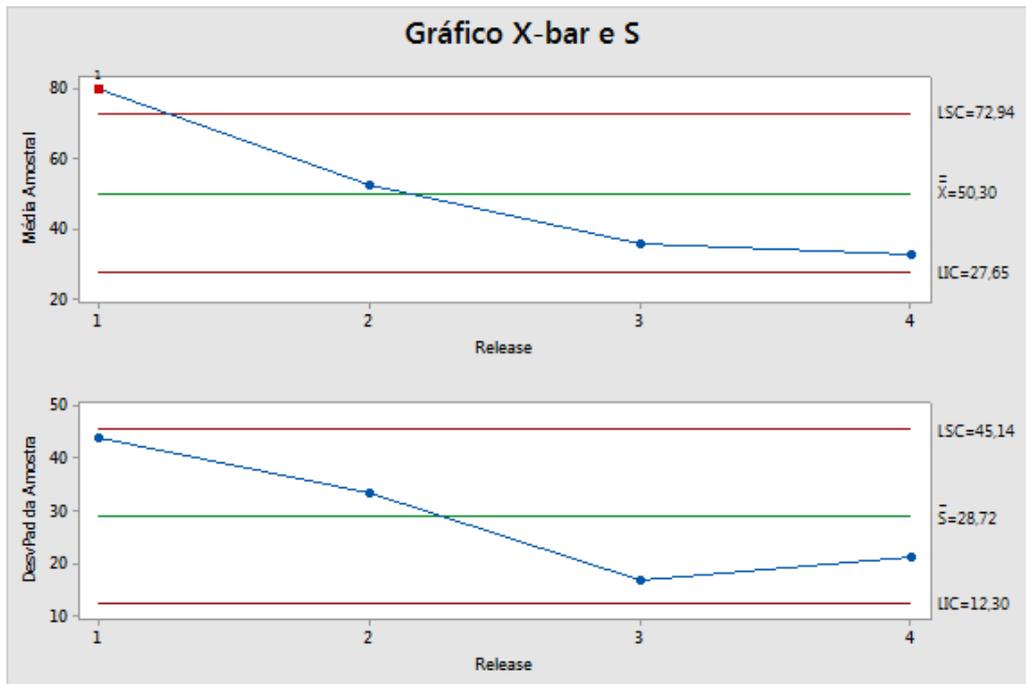


Figura 13 - Gráfico X-bar e S (Adaptado de (FLORAC e CARLETON, 1999))

c) Gráfico de Controle do Valor Individual e Amplitudes Móveis (Individuals and Moving Range Chart – XmR

O Gráfico de Controle do Valor Individual e Amplitudes Móveis é adequado quando uma mesma variável é medida frequentemente. O gráfico X é um gráfico sequencial que representa os valores individuais enquanto o gráfico mR representa a variação de um valor em relação ao valor anterior. Os limites para o gráfico X são chamados de *Limites Naturais do Processo*, pois representam a variabilidade associada a medidas individuais.

A análise do comportamento do processo é realizada através dos dois gráficos. Os testes de estabilidade 1 e 4 devem ser aplicados ao gráfico mR e os testes 1, 2, 3 e 4 devem ser aplicados ao gráfico X para verificar a estabilidade do processo.

O gráfico XmR também pode ser utilizado para dados de atributos, isto quando os dados se referem a contagens de entidades que representam o tamanho total de uma população ou seu status. Este gráfico só não é adequado para eventos raros ou quando os valores medidos possuem muitos zeros (FLORAC e CARLETON, 1999).

A Figura 14 apresenta as fórmulas para o cálculo dos limites dos gráficos XmR.

| | | |
|-----------|---|--|
| X | $\text{UNPL}_X = \bar{X} + 2660\overline{mR}$ $\text{CL}_X = \bar{X}$ $\text{LNPL}_X = \bar{X} - 2660\overline{mR}$ | <p>Onde:</p> $\bar{X} = \frac{1}{k} \sum_{i=1}^{i=k} X_i$ <p>k = número de observações</p> |
| mR | $\text{UCL}_R = 3,268\overline{mR}$ $\text{CL}_R = \overline{mR}$ $\text{LCL}_R = \text{não existe}$ | <p>Onde:</p> $\overline{mR} = \frac{1}{r} \sum_{i=1}^{i=r} mR_i$ $mR_i = X_{i+1} - X_i \text{ para } 1 \leq i \leq k-1$ <p>r = k-1 e k = número de observações</p> |

Figura 14 - Fórmulas para cálculo dos limites dos gráficos XmR (ROCHA *et al.*, 2012)

A Figura 15 apresenta um exemplo do gráfico XmR gerado a partir da ferramenta Minitab (Minitab Inc., 2013), utilizando valores de um exemplo de FLORAC e CARLETON (1999) sobre horas gastas com atividades de manutenção de um produto.

A partir das informações exibidas no gráfico da Figura 15, pode se concluir que o subprocesso analisado é estável, pois nenhum dos testes de estabilidade executados falharam.

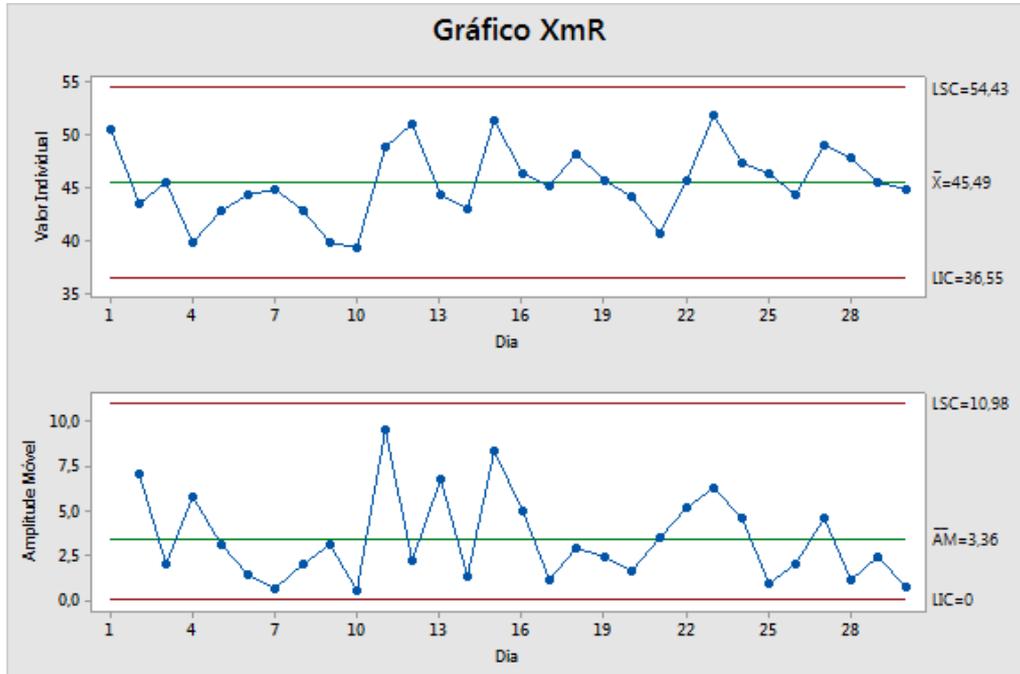


Figura 15 - Gráfico XmR (Adaptado de (FLORAC e CARLETON, 1999))

d) Gráfico de Controle do Valor Individual e Amplitude Móvel Mediana (Individuals and Median Moving Range Chart – XMmR)

O Gráfico de Controle do Valor Individual e Amplitude Móvel Mediana é similar ao gráfico XmR, porém ao invés da média, utiliza a mediana para realizar os cálculos. A mediana pode ser mais sensível às causas especiais. Assim como o gráfico XmR o gráfico XMmR, possui apenas um grupo de dados.

A análise do comportamento do processo é realizada através dos dois gráficos. Os testes de estabilidade 1 e 4 devem ser aplicados ao gráfico MmR e os testes 1, 2, 3 e 4 devem ser aplicados ao gráfico X para verificar a estabilidade do processo.

A Figura 16 apresenta as fórmulas para o cálculo dos limites dos gráficos XMmR.

| | |
|--|--|
| <p>X</p> $UNPL_X = \bar{X} + 3,145MmR$ $CL_X = \bar{X}$ $LNPL_X = \bar{X} - 3,145MmR$ | <p>Onde:</p> $\bar{X} = \frac{1}{k} \sum_{i=1}^{i=k} X_i$ <p>k = número de observações</p> |
| <p>MmR</p> $UCL_R = 3,865MmR$ $CL_R = MmR$ $LCL_R = \text{não existe}$ | <p>Para determinar MmR, os valores de cada moving range devem ser colocados em ordem crescente. Se o número de moving ranges (k) for ímpar, a mediana moving range (MmR) estará na posição $[(k-1)/2] + 1$. Se k for par, a mediana moving range será a média dos valores das posições $k/2$ e $k/2 + 1$.</p> |

Figura 16 - Fórmulas para cálculo dos limites dos gráficos XMmR (ROCHA *et al.*, 2012)

A Figura 17 apresenta um exemplo do gráfico XMmR gerado a partir da ferramenta Minitab (Minitab Inc., 2013), utilizando os mesmos valores do gráfico gerado na Figura 15.

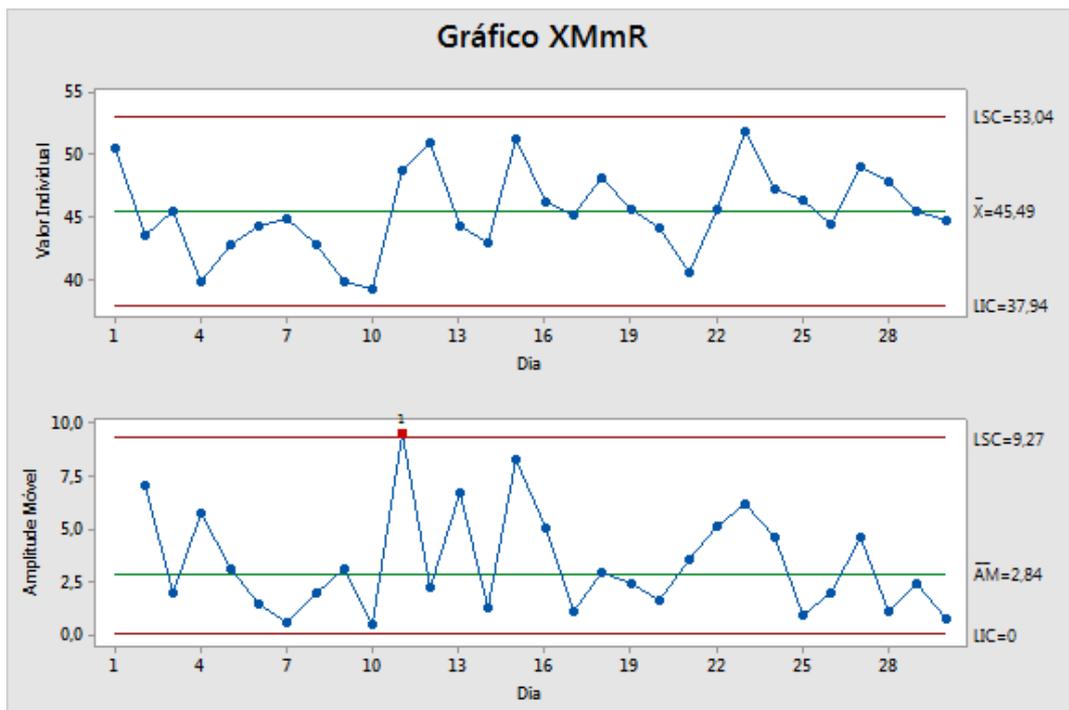


Figura 17 - Gráfico XMmR (Adaptado de (FLORAC e CARLETON, 1999))

Porém, diferente do resultado apresentado pelo gráfico XmR, o gráfico XMmR possui um ponto no formato quadrado (décimo ponto do gráfico MmR) que representa falha na execução de um teste de estabilidade, para este caso um de seus pontos fora do limite superior de controle. Isso demonstra que os mesmos dados podem apresentar resultados diferentes para cada tipo de gráfico.

2.2.1.2 Gráficos de Controle para dados de atributos

Serão apresentados os seguintes tipos de gráficos para o tipo de dados de atributos:

- a) Gráfico C (*C Chart*)
- b) Gráfico U (*U Chart*)
- c) Gráfico Z (*Z Chart*)

a) *Gráfico C*

O Gráfico C é adequado quando se deseja realizar a contagem de eventos em uma mesma área de observação. Os dados devem seguir uma distribuição de Poisson.³ Nas condições corretas, o gráfico C pode ser utilizado para representar, por exemplo, o número de defeitos encontrados em uma versão de um sistema (ROCHA *et al.*, 2012).

Os gráficos C, U e Z possuem apenas um gráfico, diferente dos gráficos apresentados anteriormente. A análise do comportamento desses gráficos é realizada a partir do teste de estabilidade 1.

A Figura 18 apresenta as fórmulas para o cálculo dos limites do gráfico C.

³ Distribuição discreta de probabilidade aplicável a ocorrências de um evento em um intervalo especificado.

| | |
|--|---|
| $UCL_c = \bar{c} + 3\sqrt{\bar{c}}$ $CL_c = \bar{c}$ $LCL_c = \bar{c} - 3\sqrt{\bar{c}}$ | <p>Onde:</p> $\bar{c} = \frac{\text{número total de defeitos}}{\text{número de observações}}$ |
|--|---|

Figura 18 - Fórmulas para cálculo dos limites do gráfico C (ROCHA *et al.*, 2012)

A Figura 19 apresenta um exemplo do gráfico C gerado a partir da ferramenta Minitab (Minitab Inc., 2013), utilizando valores de um exemplo de FLORAC e CARLETON (1999) sobre quedas não programadas de um sistema em execução, as quedas são computadas como erros reportados pelos usuários do sistema. A partir do gráfico gerado pela ferramenta, podemos analisar que dois pontos estão fora dos limites superiores, eles estão identificados na forma de um quadrado e possuem o número do teste de estabilidade que falhou, neste caso o *Teste 1*.

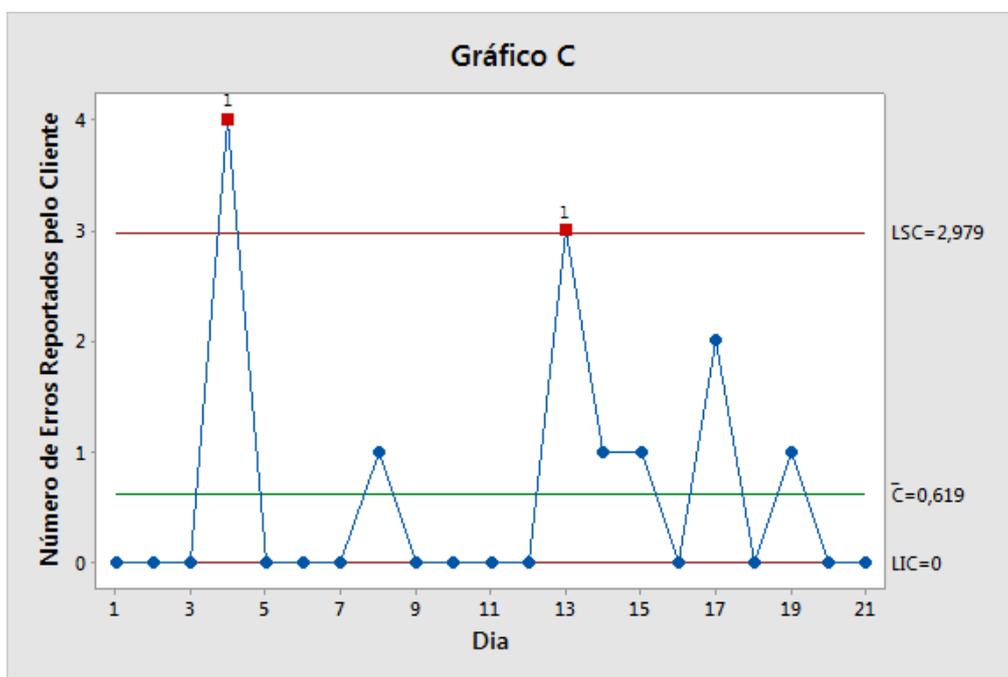


Figura 19 - Gráfico C (Adaptado de (FLORAC e CARLETON, 1999))

b) Gráfico U

O Gráfico U, assim como o gráfico C, é adequado para a contagem de eventos, a única diferença é que para o gráfico U os eventos devem ser medidos em áreas de observações diferentes. Os dados também devem seguir uma distribuição de Poisson.

Antes da geração do gráfico U os valores devem ser convertidos em taxas, como, por exemplo, número de defeitos por KSLOC. Os limites de controle são calculados para cada observação devido à diferença das áreas de observação.

A Figura 20 apresenta as fórmulas para o cálculo dos limites do gráfico U.

| | |
|--|--|
| $UCL_u = \bar{u} + 3\sqrt{\frac{\bar{u}}{a_i}}$ $CL_u = \bar{c}$ $LCL_u = \bar{u} - 3\sqrt{\frac{\bar{u}}{a_i}}$ | <p>Onde:</p> $\bar{u} = \frac{\sum c_i}{\sum a_i}, \text{ com } i \text{ variando de } 1 \text{ a } k$ <p>c_i = valor da i-ésima observação a_i = tamanho da i-ésima área de observação k = número de observações</p> |
|--|--|

Figura 20 - Fórmulas para cálculo dos limites do gráfico U (ROCHA *et al.*, 2012)

A Figura 21 apresenta um exemplo do gráfico U gerado a partir da ferramenta Minitab (Minitab Inc., 2013), utilizando valores de um exemplo de FLORAC e CARLETON (1999) sobre inspeções de códigos realizadas em um produto. O primeiro ponto do gráfico encontra-se fora do limite superior de controle (identificado por um ponto em forma de um quadrado e o número do teste que falhou) indicando a instabilidade do subprocesso analisado.

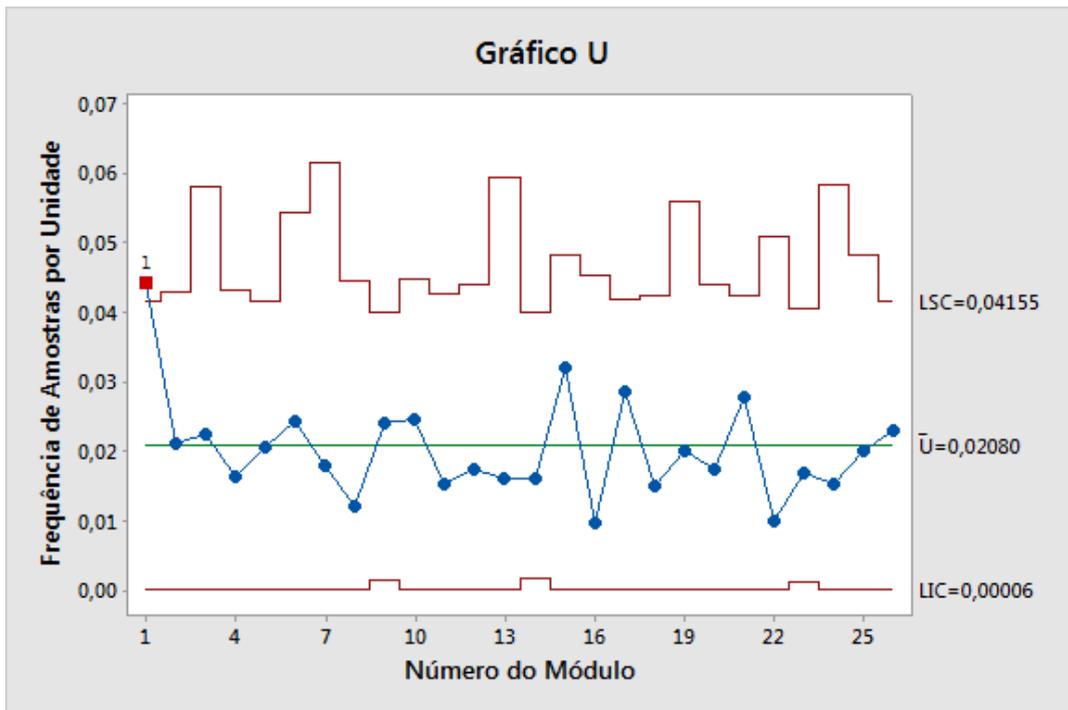


Figura 21 - Gráfico U (Adaptado de (FLORAC e CARLETON, 1999))

c) Gráfico Z

O Gráfico Z é utilizado para converter os valores de um gráfico U para a escala baseada no desvio padrão (σ). Após a conversão, os dados são obtidos em unidades sigma, sendo assim, os limites de controle serão zero para o limite central e 3σ e -3σ para os limites inferior e superior.

A Figura 22 apresenta as fórmulas para conversão dos valores.

$$\sigma u_i = \sqrt{\frac{u}{a_i}}$$

$$Z_i = \frac{u_i - \bar{u}}{\sigma u_i}$$

Figura 22 - Fórmulas para conversão dos dados – Gráfico Z (ROCHA *et al.*, 2012)

A Figura 23 apresenta um exemplo do gráfico U sobre defeitos/KSLOC (medida utilizada para determinar o tamanho de um sistema ou programa, significa mil linhas de código) por caso de uso implementado detectados em inspeções de um sistema específico (ROCHA *et al.*, 2012).

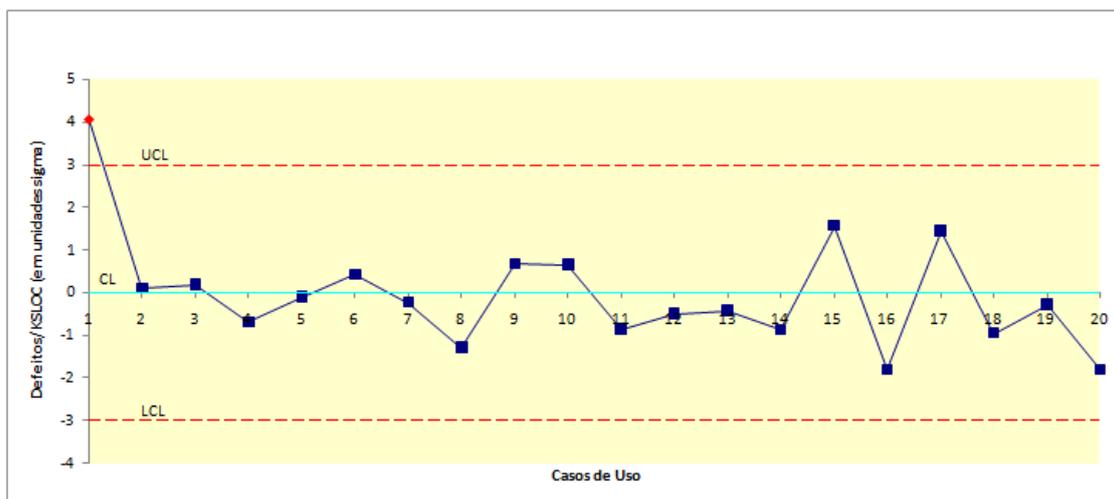


Figura 23 - Gráfico Z (Rocha *et al.*, 2012)

2.3 Trabalhos Relacionados à Execução de Processos

Através de uma revisão ad hoc da literatura foi possível identificar trabalhos que estão relacionados à execução/simulação de processos, tanto para análise de desempenho de processos de software, quanto para processos de software. Os trabalhos serão apresentados nas próximas subseções. A seção 2.3.1 apresenta o trabalho de REIS (2003), sobre uma abordagem flexível para execução de processos de software. A seção 2.3.2 apresenta o trabalho de GONÇALVES (2012) sobre a definição de um processo para execução da análise de desempenho de processos de software e a criação de uma ferramenta para apoiar esta execução. As seções 2.3.3 e 2.3.4 apresentam os trabalhos de MURTA *et al.* (2002) e ALEIXO *et al.* (2011) referentes à simulação e execução de processos de software.

2.3.1 Uma Abordagem Flexível para Execução de Processos de Software

O trabalho de REIS (2003) retrata a proposta para criação de um meta-modelo conceitual que auxilie na construção e manipulação de modelos de processos de software, e que permita a integração de mecanismos para modelagem, instanciação e execução de processos de software, denominada APSEE.

O meta-modelo possui uma linguagem visual para a modelagem de processos, a APSEE-PML. A representação gráfica das principais construções da linguagem são apresentadas na Figura 24. Do ponto de vista da sincronização de eventos em

processo/atividade, a execução de processo leva em consideração as atividades e seus fragmentos (subatividades), a disponibilidade de recursos e as pré e pós-condições das atividades.

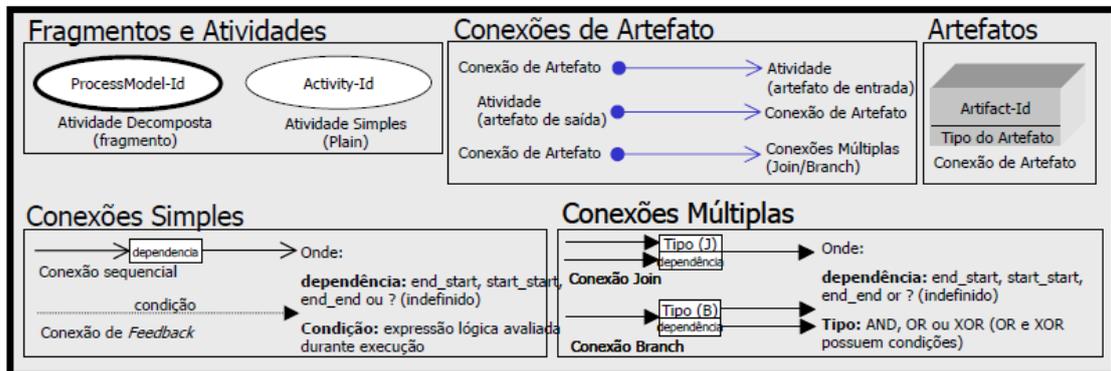


Figura 24- Representação gráfica dos principais construtores da APSEE-PML (REIS,2003)

O modelo APSEE está dividido em três camadas principais: meta-modelo, mecanismos de gerência de processos e nível de interação com o usuário. A Figura 25 apresenta uma visão geral do funcionamento do APSEE. Os componentes e ferramentas que compõem estas camadas são:

- **Componente do Meta-modelo:** esta camada possui os componentes: (i) *Processo de Software*, que são modelos de processo de software em seus diferentes estados; (ii) *Templates para Reutilização*, que são os processos abstratos descritos que podem ser reutilizados; (iii) *Hierarquia de Tipos*, são tipos definidos pelo meta-modelo e são utilizados na descrição de processos abstratos; (iv) *Organização*, incorpora o modelo de recursos de apoio utilizados pelas atividades, o modelo de pessoas da organização (agentes), suas habilidades, afinidades, cargos e grupos de trabalho; (v) *Artefatos de Software* que correspondem aos itens de dados manipulados e criados durante a execução do processo de software; (vi) *Ferramentas*, responsáveis por armazenar as informações sobre as ferramentas que são utilizadas na execução das atividades; (vii) *Políticas*, são as regras definidas pelo usuário para a execução do processo, e; (viii) *Conhecimento sobre o Processo*, componente responsável por definir e armazenar as métricas e estimativas para os componentes de processo.

- **Mecanismos para Gerência de Processos:** (i) *APSEE-PML*, linguagem para a modelagem do processo; (ii) *Mecanismo de Execução do Processo*, coordena a execução das atividades do processo; (iii) *APSEE-Planner*, auxilia na instanciação do processo, e; (iv) *APSEE-Reuse*, componente que apoia a reutilização de processos.
- **Interação e visualização do processo:** Este mecanismo provê mecanismos de interação especializados para os diferentes usuários.

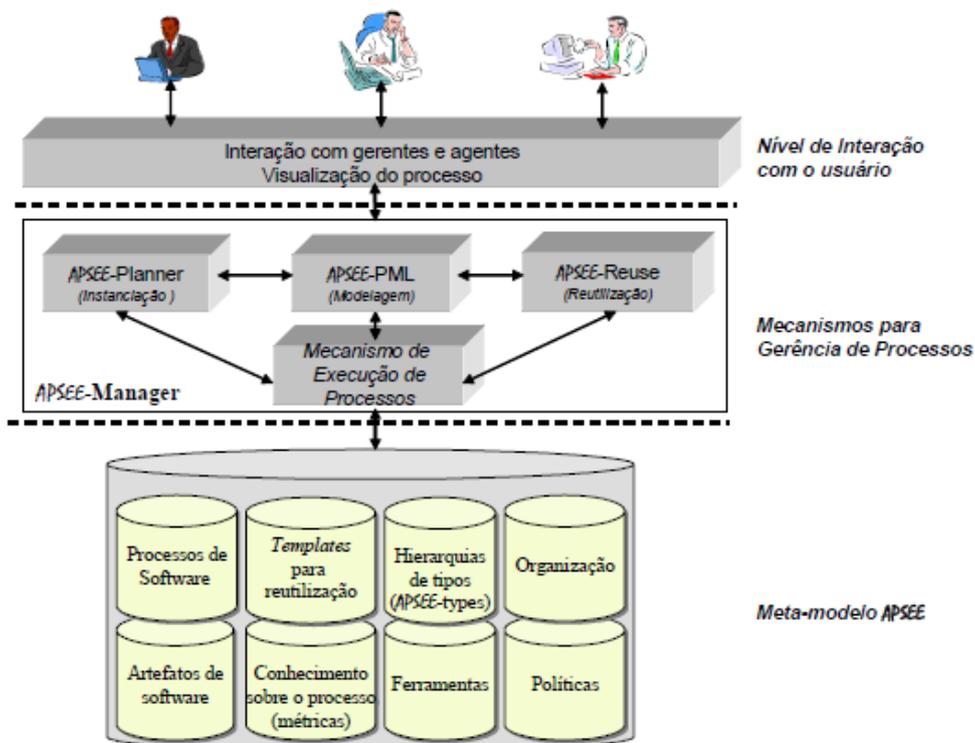


Figura 25 - Visão Geral do APSEE (REIS, 2003)

Para a execução de uma atividade, o modelo APSEE leva em consideração as atividades a serem executadas e suas decomposições, a disponibilidade dos recursos utilizados, as condições (pré e pós), as conexões entre as atividades indicando o fluxo de controle dos dados, os estados de execução das atividades e a situação atual do componente (através de condições e consulta a métricas do componente). Os possíveis estados de uma atividade em execução são apresentados na Figura 26.

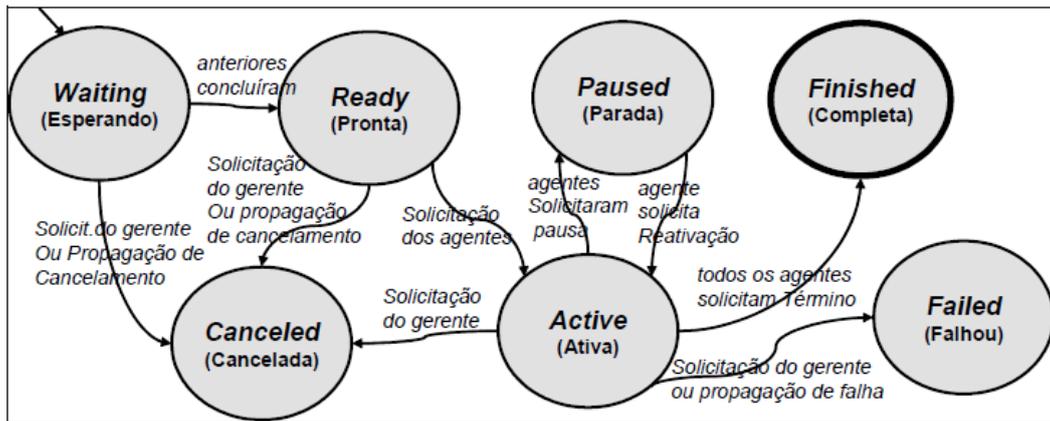


Figura 26 - Transição de Estados de uma atividade sendo executada (REIS, 2003)

O WebAPSEE é um ambiente para automação de processos de processos de software, que são genericamente denominados *Process-Centered Software Engineering Environments* (PSEEs). Este ambiente encontra-se em desenvolvimento pelo grupo de pesquisa de Engenharia de Software da Universidade Federal do Pará (UFPA) e utiliza a abordagem APSEE. O trabalho descrito na seção 2.3.2 (GONÇALVES, 2013) está inserido neste ambiente.

2.3.2 Apoio ao Controle Estatístico de Processos de Software integrado a um ADS (WebAPSEE)

O trabalho de GONÇALVES (2012) define um processo padrão para apoio ao controle estatístico de processos de software e uma ferramenta para apoiar esta execução. A ferramenta está integrada ao WebAPSEE, um Ambiente de Desenvolvimento de Software Centrado em Processos (ADS) que provê a execução de processos modelados através de uma linguagem visual própria denominada WebAPSEE-PML (Process Modeling Language).

O ambiente WebAPSEE possui duas interfaces, o *Manager Console*, que possibilita a interação entre o gerente de software e o PSEE e a *TaskAgenda* que possibilita a interação entre o ADS e o desenvolvedor.

A ferramenta proposta por GONÇALVES (2012) permite a realização das seguintes funcionalidades:

- Apoiar o planejamento da medição e a análise das métricas tanto no domínio de projetos quanto organizacional.

- Planejamento de indicadores estatísticos (gráficos de controle).
- Identificação de subprocessos críticos.
- Seleção de projetos e entidades similares.
- Identificação de problemas e causas.
- Planejamento de ações de melhorias.

Dentre os gráficos de controle apresentados nesta seção, a ferramenta apoia a geração dos gráficos XmR, XMmR, X-Bar R e X-Bar S para dados variáveis e os gráficos U e C para os dados de atributo.

As atividades e subatividades definidas por GONÇALVES (2012) estão descritas na Tabela 3.

Tabela 3 - Atividades e Subatividades do Processo

| Atividades | Subatividades |
|--|--|
| AT1 - Planejar Medição para o CEP | SA1.1- Definir Objetivos de Negócio |
| | SA1.2 - Identificar os Subprocessos Críticos |
| | SA1.3 - Definir Objetivos de Medição |
| | SA1.4 - Definir Questões |
| | SA1.5 - Definir Indicadores Estatísticos |
| | SA1.6 - Definir as Métricas |
| | SA1.7 - Identificar os Projetos Similares (metodologia organizacional) |
| | SA1.8 - Identificar Entidades Similares |
| | SA1.9 - Definir Critérios para Caracterização de Projetos e Entidades do Processo a serem Utilizadas pelo Controle Estatístico de Processo de Software |
| | SA1.10 - Caracterizar os Projetos |
| AT2 - Coletar as Métricas | SA2.1 - Coletar as Métricas |
| AT3 - Analisar os Resultados | SA3.1 - Analisar os Indicadores Estatísticos |
| AT4 - Estabilizar e Controlar o Processo | SA4.1 - Identificar os Problemas |
| | SA4.2 - Identificar as Causas |
| | SA4.3 - Planejar Ações de Melhoria e Executar as Ações de Melhoria |
| | SA4.4 - Executar as Ações de Melhoria |
| | SA4.5 - Estabelecer a Baseline de Desempenho do Processo de Software |
| | SA4.6 - Definir os Modelos de Desempenho do Processo de Software |
| | SA4.7 - Identificar a Capacidade do Processo de Software |

Esta ferramenta provê o apoio à geração de gráficos de controle visando apenas a avaliação da estabilidade do processo e executando os testes de estabilidade 1, 2, 3 e 4 descritos neste capítulo. A análise dos dados é realizada para um gráfico de controle

individualmente. A ferramenta não executa uma linha de processos e sim um único processo definido.

Não há nenhum apoio para a verificação da capacidade do processo, para a realização desta atividade é necessária a exportação dos dados e a utilização de alguma ferramenta externa.

2.3.3 Charon: uma Ferramenta para a Modelagem, Simulação, Execução e Acompanhamento de Processos de Software

O trabalho de MURTA *et al.* (2002) relata a criação de uma ferramenta extensível, baseada em agentes inteligentes, que fornece suporte à modelagem, simulação, instanciação, execução, acompanhamento, monitoramento e evolução de processos de software. A modelagem gráfica do processo é baseada na notação do digrama de atividades da UML. A Figura 27 apresenta o ambiente de modelagem de processos.

Após a modelagem do processo, um processo de desenvolvimento de software pode ser instanciado. Esta etapa é precedida por uma etapa de simulação, que verifica a existência de erros na modelagem e estima os tempos de execução dos processos compostos. Caso nenhum erro tenha sido encontrado, o processo é instanciado no projeto.

A ferramenta possui uma arquitetura baseada em agentes inteligentes que permitem a separação entre os dados que representam a execução do processo e a máquina que efetua a sua execução. Inicialmente quatro agentes foram construídos:

- **Agente de Simulação:** verifica o processo modelado quanto a sua corretude.
- **Agente de Execução:** verifica o andamento do processo validando as atividades finalizadas, decisões que foram tomadas ou o estado da base de conhecimentos.
- **Agente de Acompanhamento:** viabiliza a interação com o desenvolvedor, permitindo a visualização das atividades pendentes e as decisões que devem ser tomadas.

- **Agente de Retrocesso:** permite o retorno da base de conhecimento para um determinado tempo no passado, para que seja possível contornar erros na utilização da ferramenta.

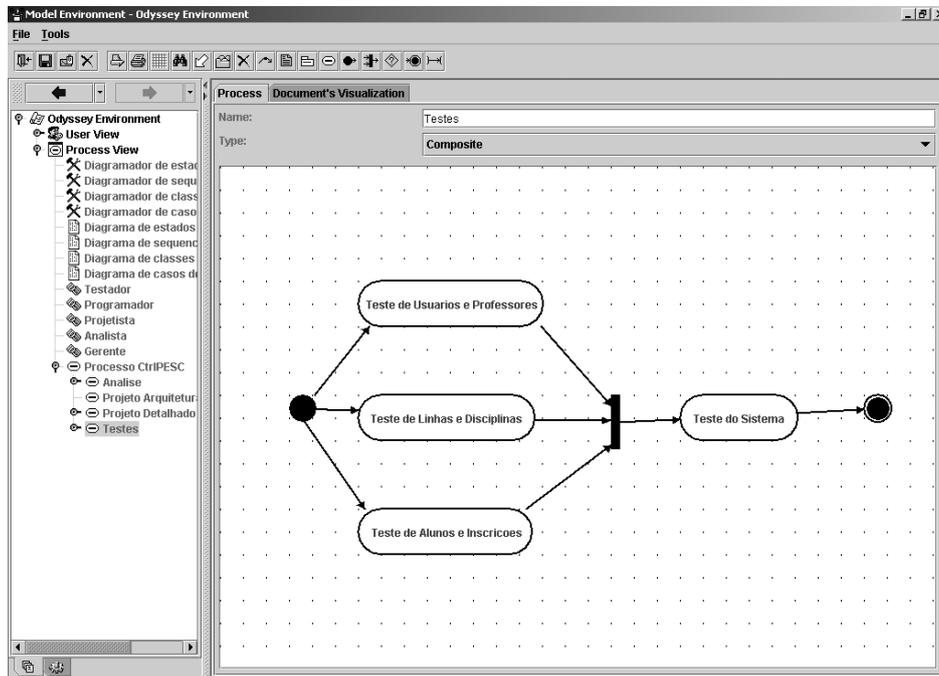


Figura 27 - Ambiente de Modelagem de Processos (Murta *et al.*, 2002)

Como principais características da ferramenta, pode-se listar: (i) modelagem gráfica do processo; (ii) simulação do processo; (iii) instanciação do processo; (iv) acompanhamento do processo; (v) monitoramento da execução; (vi) evolução do processo, e; (vii) suporte a pró-atividade e extensão da máquina de processos.

Esta ferramenta é uma máquina de processos genérica que não apoia a execução da análise de desempenho ou qualquer atividade relacionada ao controle estatístico de processos.

2.3.4 Uma Abordagem Dirigida por Modelo para Gerenciar e Customizar a Variabilidade em Processos de Software

O trabalho de ALEIXO *et al.* (2011) retrata uma abordagem para gerenciar e customizar a variabilidade em processos de software. Segundo os autores, esta abordagem promove um aumento na produtividade com relação à reutilização de processos, na integração e automação de atividades relacionadas à execução de

processos de software. A abordagem é baseada nos princípios e técnicas de linhas de produto.

A aplicação desta abordagem não necessita do desenvolvimento de novas ferramentas, pois sua implementação pode ser realizada com o uso de ferramentas existentes dirigidas por modelo. A especificação do processo de software é criada na ferramenta Eclipse Process Framework (EPF) e automaticamente transformada em uma especificação escrita na linguagem jPDL Workflow, que por sua vez pode ser implantada e executada em uma ferramenta de jBPM Workflow. Os principais passos da abordagem são:

- **Modelagem e Especificação do Processo:** O primeiro passo da abordagem é a modelagem e definição do processo de software. A ferramenta sugerida para este passo é a EPF, utilizando o meta-modelo Unified Metamodel Architecture (UMA).
- **Integração de Métricas de Processos de Software:** após a modelagem e especificação do processo, engenheiros de processo podem selecionar e especificar as métricas que serão monitoradas automaticamente e coletadas durante a execução do processo.
- **Processo para Transformação em Workflow:** a especificação do processo na ferramenta EPF é automaticamente transformada em uma descrição de workflow, ou seja, é transformado para a linguagem jPDL que é usada por jBPM.
- **Implantação e Execução do Workflow:** a última atividade da abordagem refere-se a implantação e execução do workflow em uma ferramenta de execução de workflows. Nesta etapa é realizada a monitoração das atividades processo e quantificação das métricas definidas.

Esta abordagem é utilizada para a execução de processos de software e não possui nenhum apoio à execução das atividades da análise de desempenho de processos de software.

2.4 Considerações Finais

Neste capítulo, foi apresentada uma revisão da literatura que caracteriza a análise de desempenho de processos e a execução de seu processo através dos principais conceitos, e de sua aplicabilidade na área de software. Além de trabalhos relacionados à execução e simulação de processos de software.

O trabalho de GONÇALVES (2012) é único a fazer referência à execução de atividades relacionadas à análise de desempenho de processos, porém, conforme descrito, o apoio da ferramenta é voltado mais para a geração de gráficos de controle e executa as atividades que foram definidas em um único processo. Além de não possuir nenhum conhecimento sobre como as atividades devem ser executadas ou como seus resultados devem ser analisados para obtenção da estabilidade ou capacidade do processo. O trabalho de REIS (2003) apresenta o ambiente onde esta ferramenta está inserida e alguns conceitos sobre execução de processos que foram utilizados na definição da ferramenta FIE.

Os trabalhos de MURTA *et al.* (2002) e ALEIXO *et al.* (2011) retratam respectivamente uma ferramenta e uma abordagem para execução de processos de software genéricos, não possuindo suporte algum à execução das atividades de análise de desempenho ou ao controle estatístico de processos.

A revisão da literatura realizada neste trabalho foi uma revisão Ad hoc sobre o tema pesquisado. Ferramentas e trabalhos sobre simulação de processos não estão inseridas neste contexto pois a ferramenta desenvolvida para atender aos objetivos deste trabalho possui particularidades, que serão descritas nos próximos capítulos, que impedem a reutilização de ferramentas utilizadas para esta finalidade.

CAPÍTULO 3 - O AMBIENTE SPEAKER

Este capítulo descreve o ambiente SPEAKER (Software Process pErformance Analysis Knowledge-base EnviRoment) através de seus principais módulos e sua infraestrutura. Descreve também o ambiente A2M (Ambiente de Alta Maturidade) onde o ambiente SPEAKER está inserido.

3.1 Introdução

O ambiente SPEAKER tem o objetivo de apoiar a captura, armazenamento e disponibilização do conhecimento sobre análise de desempenho de processos de software, bem como fornecer o apoio ferramental necessário para execução das atividades envolvidas (SCHOTS *et al.*, 2013).

Conforme descrito no Capítulo 1, além da dificuldade de identificar e coletar medidas adequadas, a maioria das dificuldades enfrentadas pelas organizações ao implantar a análise de desempenho de processos está relacionada à falta de conhecimento e de experiência em análise de desempenho de processos por parte das pessoas responsáveis pela gestão de processos nas organizações. Neste contexto, acredita-se que a estruturação e disponibilização de um corpo de conhecimento em análise de desempenho de processos de software, bem como um apoio ferramental para auxiliar a execução desta atividade podem minimizar estas dificuldades.

O objetivo deste capítulo é apresentar o ambiente SPEAKER e descrever a infraestrutura utilizada para o seu desenvolvimento. O capítulo também apresenta o Ambiente de Alta Maturidade (A2M), onde o ambiente SPEAKER está inserido, que fornece a infraestrutura para apoiar o desenvolvimento de novas ferramentas (SCHOTS *et al.*, 2013; BARRETO, 2011).

A seção 3.2 apresenta o Ambiente de Alta Maturidade (A2M). A seção 3.3 apresenta o ambiente SPEAKER com toda a infraestrutura desenvolvida até o momento, além da visão arquitetural e em camadas do ambiente e seus requisitos. A seção também apresenta os módulos que fazem parte do ambiente. A seção 3.4 apresenta as linhas de processos definidas por GONÇALVES (2014) que apoiam a execução das atividades do

processo de análise de desempenho e serão executadas pelo ambiente SPEAKER e explica suas características e dependências.

3.2 O Ambiente de Alta Maturidade - A2M

O Ambiente de Alta Maturidade (A2M) é um ambiente que se encontra em desenvolvimento pelo grupo de pesquisas de Qualidade de Software do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ como forma de apoiar a realização de práticas de alta maturidade em processos de software. Essas práticas, em sua maioria, são bastante complexas e, portanto, apoiá-las pode trazer benefícios para as organizações que querem adotá-las (BARRETO, 2011).

A infraestrutura do A2M utiliza é desenvolvida utilizando a linguagem JAVA, para ser utilizado via *web*. Possui um infraestrutura com componentes auxiliares que podem ser reutilizados pelas diferentes ferramentas existentes. Alguns dos serviços disponíveis atualmente pelo ambiente são: persistência, controle de acesso, criação de logs, tratamento de exceções, elementos visuais reutilizáveis, internacionalização, upload e download de arquivos, entre outros.

Este ambiente possui atualmente as seguintes ferramentas disponíveis para utilização:

- A2MObjetivo – Esta ferramenta possui as seguintes finalidades: (i) apoiar a execução de planejamento estratégico, tático e operacional em organizações de software, considerando a alta maturidade; (ii) apoiar a monitoração dos objetivos definidos, e; (iii) recomendar ações corretivas decorrentes da monitoração dos objetivos (SOARES BARRETO, 2011);
- A2MMedidas – Esta ferramenta possui um repositório de medidas adequado ao controle estatístico de processos de software para organizações que almejam alcançar a alta maturidade em seus processos de software. A definição do repositório está baseada na revisão da literatura, em uma ontologia de medição de software e em um processo para transformação de uma ontologia de domínio em um modelo de entidades e relacionamentos (SIMÕES, 2011);

- A2MComponenteProcesso – Ferramenta para apoiar a definição de processos baseada em uma abordagem para reutilização e para definição de processos com reutilização (BARRETO, 2011).

A Figura 28 exemplifica uma das telas iniciais do A2M (tela de seleção das ferramentas). A tela demonstra para o usuário conectado estavam disponíveis as ferramentas de controle de acesso, gerências das ferramentas disponíveis no ambiente, ferramenta de monitoração dos objetivos estratégicos, ferramenta de definição de componentes de processo e ferramenta de execução do processo para análise de desempenho. Todas as ferramentas seguem o padrão ilustrado, ou seja, lista de opções à esquerda com as funcionalidades, área central com o conteúdo das ferramentas e informações adicionais à direita.

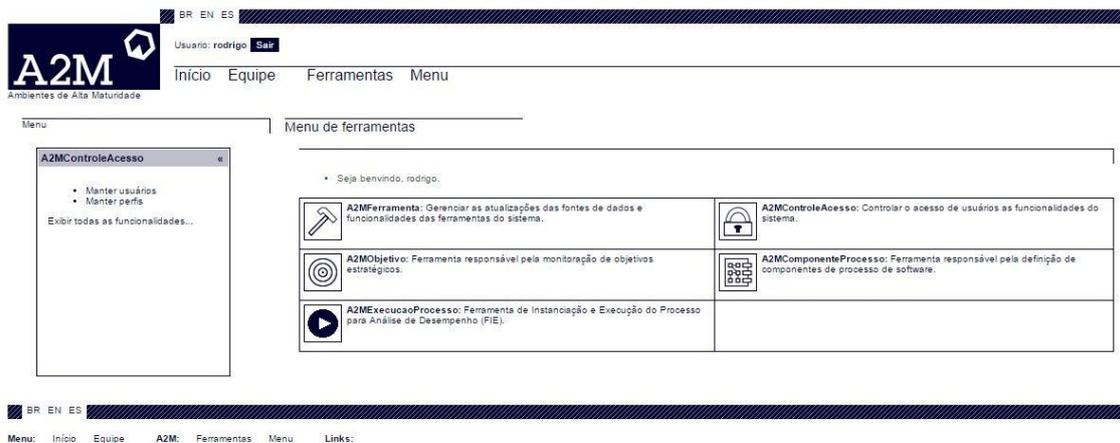


Figura 28 - Exemplo de Tela do A2M - Menu de Ferramentas

O ambiente SPEAKER está inserido no contexto do Ambiente de Alta Maturidade (A2M) que fornece a infraestrutura necessária para o desenvolvimento das novas ferramentas. O A2M tem como objetivo apoiar a realização de práticas de alta maturidade em organizações de desenvolvimento de software. Além das ferramentas principais, o SPEAKER também utiliza uma biblioteca de elementos de processos reutilizáveis que apoia a definição de processos através da ferramenta de apoio à definição de processos, esta ferramenta foi desenvolvida por BARRETO (2011). O processo de análise de desempenho que será executado é descrito neste formato e armazenado nesta biblioteca.

Para auxiliar a realização da análise de desempenho de processos, o A2M em sua versão corrente permite partir da definição de objetivos estratégicos de uma organização, passando pela definição e monitoração de objetivos quantitativos de qualidade e desempenho de processos até a definição de processos para projetos específicos.

A ferramenta A2MComponenteProcesso fornece o apoio à definição de processos baseada em reutilização. Esta ferramenta possui uma estratégia que é apresentada na Figura 29.

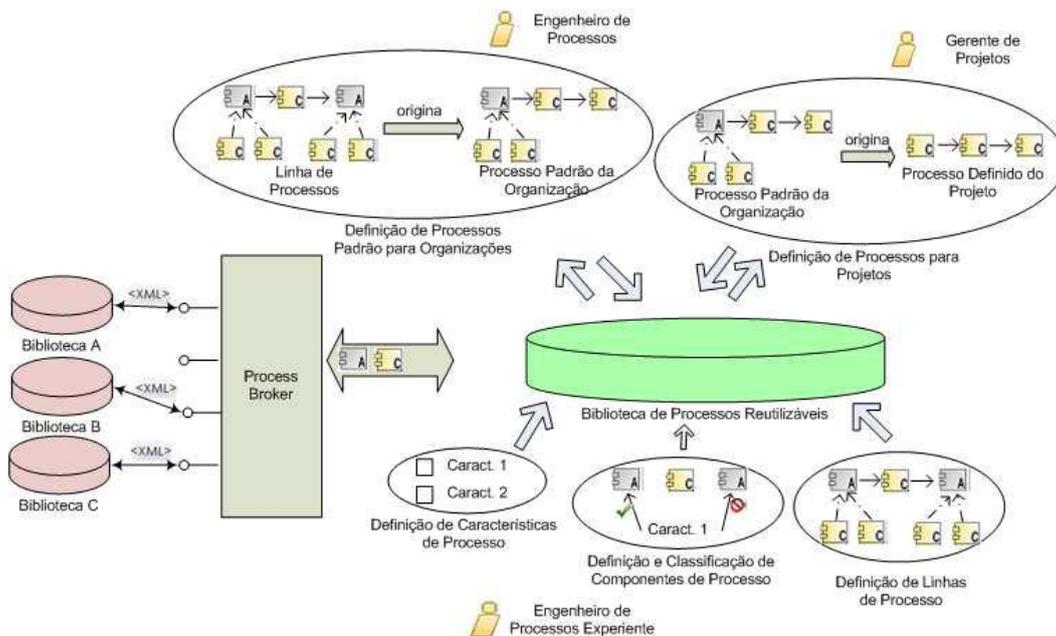


Figura 29 - Estratégia para definição de processos com reutilização (BARRETO, 2011)

Esta estratégia possui uma biblioteca de processos reutilizáveis que se relaciona com as ferramentas de definição de características de processo, de definição e classificação de componentes de processo e definição de linhas de processo que apoiam a definição de processos para reutilização. A biblioteca de processos reutilizáveis também se relaciona com a definição de processos para organização e a definição de processos para projetos que apoiam a definição de processos com reutilização.

Os principais usuários envolvidos na utilização da ferramenta são engenheiros de processo, gerentes de projetos e engenheiros de processos experientes.

A ferramenta também possui um *process broker*, cuja finalidade é facilitar o intercâmbio de itens reutilizáveis e demais informações relevantes entre bibliotecas de processos reutilizáveis.

3.3 O Ambiente SPEAKER

O ambiente SPEAKER possui o objetivo de apoiar a captura, armazenamento e disponibilização do conhecimento sobre análise de desempenho de processos de software e apoiar a execução das atividades da análise de desempenho de processos. A partir do uso desse conhecimento o ambiente SPEAKER auxilia a análise de desempenho de processos organizações que estejam buscando atingir a alta maturidade dos modelos de maturidade ou que desejam conhecer e gerenciar melhor seus processos.

A partir de uma revisão da literatura (SCHOTS *et al.*, 2013) foi possível identificar um conjunto de requisitos necessários para o ambiente SPEAKER, estes requisitos foram apresentados no capítulo 1.

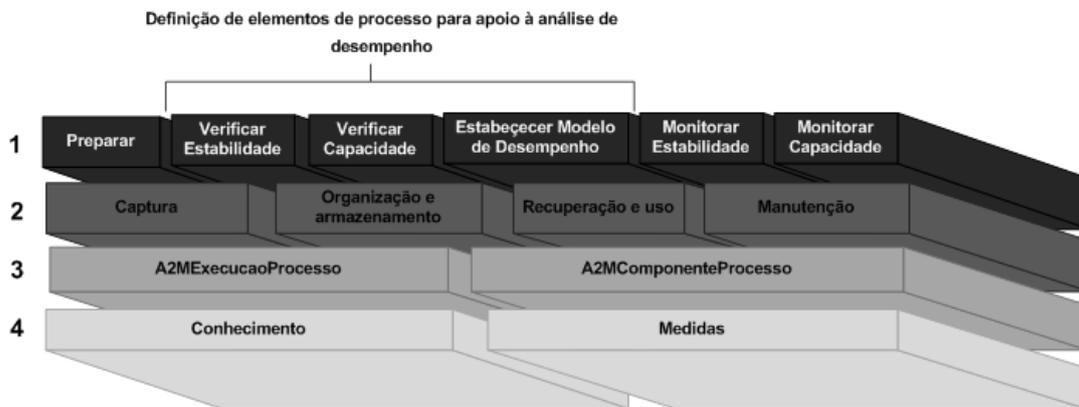


Figura 30 - Ambiente SPEAKER – Visão em Camadas

A Figura 1 exibida no capítulo 1, apresentou uma visão arquitetural do ambiente SPEAKER, outra visão disponível do ambiente é apresentada na Figura 30. Esta visão está dividida em camadas e demonstra que o ambiente SPEAKER pode ser apresentado como um ambiente que provê apoio à execução das atividades de análise de desempenho de processos (camada 1), a partir do uso de uma infraestrutura de conhecimento (camada 2) e de uma infraestrutura computacional (camada 3), fazendo uso da base de conhecimento e da base de medidas da organização (camada 4). A base de medidas da organização será controlada através de uma planilha que é fornecida pelo ambiente SPEAKER. Esta planilha possui colunas definidas para auxiliar o usuário do ambiente SPEAKER a caracterizar as medidas coletadas dos subprocessos executados pela organização.

3.3.1 Módulos do Ambiente SPEAKER

Esta seção tem como objetivo apresentar os principais módulos do ambiente SPEAKER explicando seu funcionamento e sua importância dentro do ambiente. Os principais módulos do ambiente são: (i) Sistema Baseado em Conhecimento (SBC); (ii) Biblioteca de Elementos de Processos Reutilizáveis para Análise de Desempenho, e; (iii) Ferramenta de Instanciação e Execução do Processo.

3.3.1.1 Sistema Baseado em Conhecimento – SBC

Um sistema baseado em conhecimento está sendo desenvolvido no contexto de uma tese de doutorado pelo Grupo de Qualidade de Software do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ a fim de disponibilizar e manter o corpo de conhecimento necessário para executar a análise de desempenho de processos. SCHOTZ *et al.*, 2014a definem as principais atividades da gerência do conhecimento que serão executadas pelo SBC:

- Identificação e aquisição do conhecimento: esta atividade envolve tanto o conhecimento explícito quanto o tácito, e está sendo realizada com o objetivo de criar o corpo de conhecimento inicial do SPEAKER e apoiar o cadastro de novos conhecimentos a partir de lições aprendidas;
- Organização e armazenamento: o conhecimento deve ser organizado e armazenado de forma que seja de fácil acesso e entendimento pelos profissionais;
- Recuperação e uso: esta atividade permite que o conhecimento seja utilizado pelos profissionais para auxiliá-los durante a execução da análise de desempenho;
- Manutenção: esta atividade é realizada no contexto específico de uma organização na qual o ambiente SPEAKER está sendo utilizado, permitindo o cadastro, alteração ou exclusão de itens de conhecimento de acordo com suas necessidades.

Um conjunto de atividades e tarefas de análise de desempenho foi identificado (SCHOTZ *et al.*, 2014a) na literatura técnica, tanto na área de software como em áreas não relacionadas a software (principalmente manufatura). A Figura 31 apresenta as atividades e tarefas definidas que serviram de base para a identificação do conhecimento necessário para executá-las. Estas atividades são detalhadas na Tabela 4.

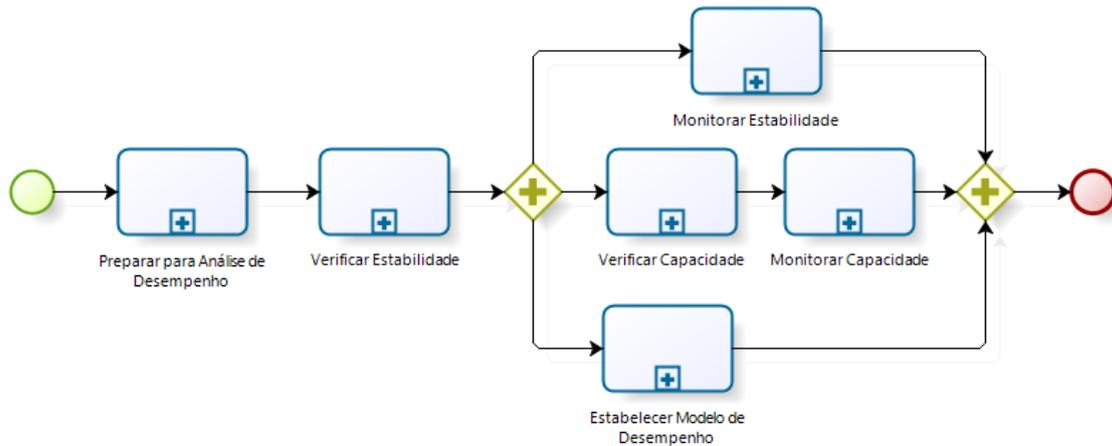


Figura 31 - Atividades da Análise de Desempenho de Processos (SCHOTS *et al.*, 2014a)

As atividades da Análise de Desempenho de Processos são executadas pelo SBC e disponibilizam o conhecimento necessário para sua execução. Este conhecimento é composto tanto pelo conhecimento explícito como o conhecimento tácito. No contexto do ambiente SPEAKER, o conhecimento explícito sobre análise de desempenho encontra-se disperso em livros, artigos, dissertações e teses. O conhecimento tácito por sua vez, está na mente dos especialistas e, portanto, precisa ser identificado e capturado para se tornar explícito e disponível.

O conhecimento do SBC será disponibilizado de diferentes formas, são elas:

- Processo: descrição das atividades, bem como suas sequências e dependências.
- Recomendações: técnicas sugeridas de acordo com a atividade que está sendo executada.
- Boas práticas: consenso entre os especialistas.
- Lições aprendidas: realizadas pelos profissionais após executada.

Tabela 4 - Atividades e Tarefas da Análise de Desempenho de Processos (SCHOTS *et al.*, 2014a)

| Atividades | Tarefas |
|--------------------------|--|
| Preparar para Análise de | Identificar objetivos quantitativos organizacionais de |

| Atividades | Tarefas |
|-----------------------------------|--|
| Desempenho | qualidade e desempenho |
| | Identificar subprocessos críticos |
| | Avaliar os subprocessos quanto à adequação à análise de desempenho |
| Verificar Estabilidade | Agrupar dados de projetos similares |
| | Selecionar tipo de gráfico de controle |
| | Construir gráfico de controle |
| | Aplicar testes de estabilidade e tendências |
| | Identificar e realizar ações corretivas para estabilizar processo (se necessário) |
| | Confirmar estabilidade (se pertinente) |
| | Estabelecer <i>baseline</i> de desempenho |
| Verificar Capacidade | Determinar capacidade do subprocesso |
| | Comparar capacidade com objetivos quantitativos de qualidade e desempenho |
| | Identificar e realizar ações corretivas para tornar o processo capaz (se necessário) |
| Estabelecer Modelos de Desempenho | Identificar variável dependente |
| | Identificar possíveis variáveis independentes |
| | Selecionar método apropriado de acordo com o tipo das variáveis envolvidas |
| | Desenvolver a equação de regressão, modelo probabilístico ou simulação |
| | Calibrar e testar o modelo |
| Monitorar Estabilidade | Atualizar gráfico de controle com novos dados coletados |
| | Verificar necessidade de recalcular <i>baseline</i> de desempenho |
| | Aplicar testes de estabilidade |
| | Confirmar estabilidade |
| Monitorar Capacidade | Monitorar Estabilidade (etapa) |
| | Verificar Capacidade (etapa) |

A partir da infraestrutura definida para o SBC, os conhecimentos necessários são integrados às atividades da análise de desempenho a fim de apoiar sua execução com base nas regras catalogadas no corpo de conhecimento. Desta forma o ambiente SPEAKER funcionará como um guia/assistente, que conduzirá o profissional a realizar as atividades de análise de desempenho (SCHOTZ *et al.*, 2014a).

3.3.1.2 Biblioteca de Elementos de Processos Reutilizáveis para Análise de Desempenho

A Biblioteca de Elementos de Processos Reutilizáveis definida por BARRETO (2011) apresenta uma abordagem para definição de processos baseada em reutilização. Esta abordagem adapta conceitos de reutilização de produtos de software para o contexto de definição de processos, e conta com apoio ferramental para definição de

linhas de processo e elementos de processo. Dentre os principais conceitos dessa abordagem temos *linhas de processo*, que são linhas de produtos onde os produtos são processos de software e *elemento de processo* que é um encapsulamento de informações e comportamentos de processo em um dado nível de granularidade, onde este nível de granularidade varia de acordo com o uso pretendido. Estes elementos podem ser: (i) *atividades*, que são *elementos de processo* completamente definidos; ou (ii) *componentes de processo*, que são *elementos de processo* que constituem a unidade básica de definição de processos com reutilização. Há dois tipos de *componentes de processo*: *concreto* ou *abstrato*. Sendo que um *componente concreto* não admite qualquer variação e um *componente abstrato* possui variabilidades, ou seja, possui mais de uma forma de realização. Os componentes concretos são executados da forma que foram descritos, enquanto os componentes abstratos necessitam que escolhas sejam realizadas até que ele chegue ao nível de um componente concreto e possa ser executado. Uma *linha de processo* é capaz de modelar variabilidades através de pontos de variação e de determinar quais elementos são opcionais ou obrigatórios. Estes pontos de variação são representados por *componentes abstratos*.

GONÇALVES (2014) definiu os elementos e linhas de processos necessários para a execução do processo de análise de desempenho de processos de software. Estes elementos encontram-se disponíveis na biblioteca e foram definidos de acordo com os requisitos exigidos pelos modelos de maturidade MR-MPS-SW e CMMI-DEV e pela norma ISO/IEC 15504.

A Tabela 5 apresenta um exemplo dos elementos de processo definidos por GONÇALVES (2014) juntamente com suas variabilidades (quando aplicável) e a justificativa para sua definição.

Tabela 5 - Elementos de Processo (GONÇALVES, 2014)

| Elementos de Processo | Variantes | Justificativa |
|---|--|---|
| Registrar a categorização dos dados da medida (<i>Atividade</i>) | - | Este elemento de processo auxilia na categorização dos dados da medida e posteriormente na escolha do tipo de gráfico que será utilizado para analisar o subprocesso. |
| Construir gráfico de controle (<i>Componente abstrato</i>) | Construir gráfico de controle XmR com Statistica (<i>Componente concreto</i>) | Este elemento de processo auxilia na construção de gráficos de controle que são |

| | | |
|--|--|---|
| | Construir gráfico de controle XmR com Minitab (<i>Componente concreto</i>) | utilizados para analisar o comportamento do processo, ou seja, verificar a estabilidade do subprocesso a partir da distinção dos tipos de variações que o desempenho do subprocesso apresenta. |
| | Construir gráfico de controle X-bar e R com Minitab (<i>Componente concreto</i>) | |
| | Construir gráfico de controle X-bar e S com Statistica (<i>Componente concreto</i>) | |
| Registrar escolha da solução selecionada para melhorar a capacidade do subprocesso (<i>Atividade</i>) | | Este elemento de proceso auxilia na escolha de uma solução que pode ser implementada para melhorar a capacidade do subprocesso analisado. |
| Determinar capacidade (<i>Componente abstrato</i>) | Determinar capacidade de análise de XmR com Minitab (<i>Componente concreto</i>) | Este elemento de processo auxilia na determinação da capacidade do subprocesso, ou seja, determina se o subprocesso é capaz de atingir os objetivos da organização e atender aos requisitos do cliente. |
| | Determinar capacidade de análise de XbarR com Statistica (<i>Componente concreto</i>) | |
| Calibrar e testar o modelo de desempenho (<i>Componente concreto</i>) | - | Este elemento de proceso auxilia na calibração do modelo de desempenho, que consiste em verificar se o modelo ainda reflete o desempenho futuro do subprocesso. |

Para o processo de análise de desempenho foram definidas as seguintes linhas de processo: “Verificar estabilidade”, “Determinar capacidade” e “Estabelecer modelo de desempenho”. Os elementos de processo foram definidos para cada uma dessas linhas. A Tabela 6 apresenta o total de elementos definidos para cada linha de processo.

Tabela 6 - Elementos de processos definidos (SCHOTZ *et al.*, 2014a)

| Critério | Linha de Processo “Verificar Estabilidade” | | Linha de Processo “Determinar capacidade” | | Linha de Processo “Estabelecer modelo de desempenho” | |
|---|--|------------|---|------------|--|------------|
| | Componentes | Atividades | Componentes | Atividades | Componentes | Atividades |
| Quantidade de elementos de processo definidos | 33 | 5 | 10 | 1 | 4 | 2 |

3.3.1.3 Ferramenta de Instanciação e Execução do Processo

O último módulo a ser apresentado é o objeto de pesquisa desta dissertação e tem como objetivo fornecer um apoio ferramental ao ambiente SPEAKER para

instanciar e executar o processo de análise de desempenho. As funcionalidades e requisitos da ferramenta FIE estão descritos no capítulo 4, por isso, a ideia desta seção é apresentar o objetivo principal da ferramenta.

A Ferramenta para Instanciação e Execução do Processo de Análise de Desempenho (FIE) recebe as demandas do SBC quanto à técnica a ser utilizada no próximo passo da execução da análise de desempenho, para que, então, a ferramenta FIE possa selecionar o elemento de processo definido por GONÇALVES (2014) na biblioteca de elementos de processos reutilizáveis e possa disponibilizar suas informações para execução deste elemento. Os resultados desta execução são armazenados nesta ferramenta e então disponibilizados para o SBC, para que, a decisão sobre a continuidade da execução do processo seja tomada.

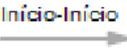
Como a ferramenta FIE é objetivo desta dissertação, a descrição completa de seus objetivos, requisitos e funcionalidades estão presentes no Capítulo 4.

3.4 Análise de Desempenho de Processos de Software no ambiente SPEAKER

A execução do processo de análise de desempenho no ambiente SPEAKER ocorre através das linhas de processo definidas por GONÇALVES (2014). Cada linha de processo implementa uma parte deste processo. No que diz respeito à representação gráfica (notação utilizada), os conceitos definidos anteriormente são representados pelos elementos gráficos que constam na Tabela 7. As linhas de processos de GONÇALVES (2013) foram modeladas a partir dessa notação.

Tabela 7 - Descrição dos elementos gráficos da abordagem de BARRETO (2011)

| Notação | Nome do elemento | Descrição |
|---|---|--|
|  | Componente de processo concreto obrigatório | Representa elementos não configuráveis (variante) |
|  | Componente de processo concreto opcional | Representa a opcionalidade de elementos não configuráveis (variante) |
|  | Componente de processo abstrato obrigatório | Representa pontos de variação |
|  | Componente de processo abstrato opcional | Representa opcionalidade de pontos de variação |
|  | Atividade obrigatória | Representa uma atividade obrigatória em uma arquitetura de processos |

| | | |
|---|---|---|
|  | Atividade opcional | Representa uma atividade opcional em uma arquitetura de processos |
|  | Conexão obrigatório entre elementos de processo | Representa relacionamento obrigatório entre elementos de processo |
|  | Conexão opcional entre elementos de processo | Representa relacionamento opcional entre elementos de processo |
|  | Item início da arquitetura | Representa ponto de início de uma arquitetura |
|  | Item fim da arquitetura | Representa ponto de fim de uma arquitetura |

GONÇALVES (2014) definiu três linhas de processos para executar as atividades relacionadas à Análise de Desempenho de Processos, são elas: (i) Verificar estabilidade do subprocesso; (ii) Determinar a capacidade do subprocesso, e; (iii) Estabelecer o modelo de desempenho do subprocesso. As principais características das linhas de processo serão descritas nos próximos parágrafos.

A primeira linha de processo definida é “Verificar estabilidade”. Esta linha abrange os elementos de processo necessários ao contexto da estabilidade de processos, e possui as seguintes características:

- **Definido por:** COPPE/UFRJ
- **Participantes Necessários:** usuário do ambiente SPEAKER.
- **Características Atendidas:** MR-MPS-SW – Nível B, CMMI-DEV – Nível 4, ISO/IEC 15504 e Verificação de estabilidade.

Arquitetura da Linha de Processos:

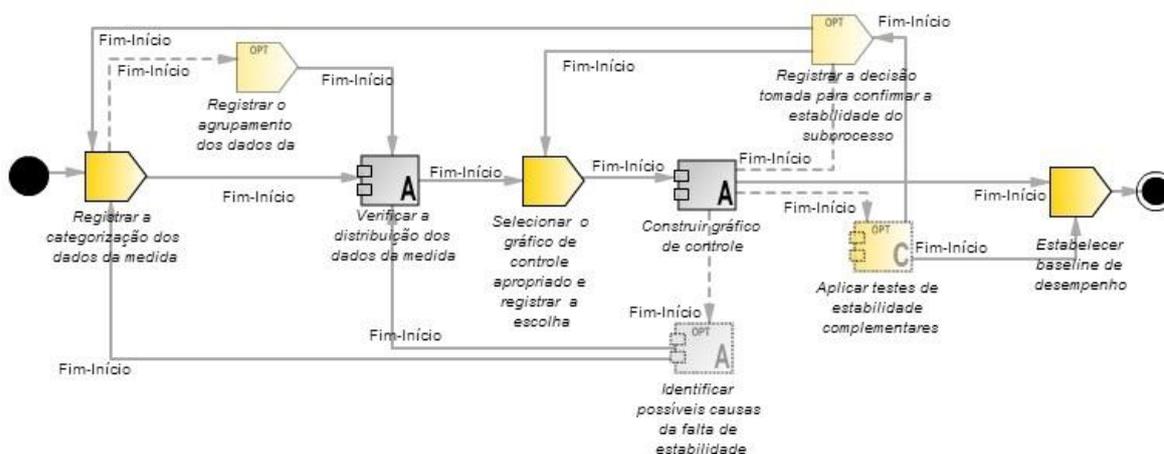


Figura 32 - Linha de Processo - Verificar Estabilidade (GONÇALVES, 2014)

A Tabela 6 apresentou a quantidade de elementos definidos para cada uma das linhas de processo. A segunda linha de processo definida é “Determinar a capacidade”.

Esta linha abrange elementos de processo necessários ao contexto da capacidade de processos. Para a execução desta linha de processo, o subprocesso analisado precisa ser estável, ou seja, já tenha executado a linha de processo “Verificar estabilidade” e obtido sua estabilidade como resultado. Esta linha de processo possui as seguintes características:

- **Definido por:** COPPE/UFRJ
- **Participantes Necessários:** usuário do ambiente SPEAKER.
- **Características Atendidas:** MR-MPS-SW – Nível B, CMMI-DEV – Nível 4, ISO/IEC 15504 e Determinar a capacidade.

Arquitetura da Linha de Processos:

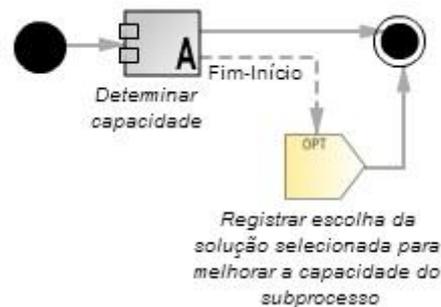


Figura 33 - Linha de Processo - Determinar Capacidade (GONÇALVES, 2014)

A terceira e última linha de processo definida é “Estabelecer modelo de desempenho”. Esta linha abrange elementos de processo necessários ao contexto do modelo de desempenhos de processos. Para a execução desta linha de processo, o subprocesso analisado também precisa ser estável, ou seja, é necessário que já se tenha executado as linhas de processo “Verificar estabilidade” e obtido sua estabilidade como resultado. Esta linha de processo possui as seguintes características:

- **Definido por:** COPPE/UFRJ
- **Participantes Necessários:** usuário do ambiente SPEAKER.
- **Características Atendidas:** MR-MPS-SW – Nível B, CMMI-DEV – Nível 4, ISO/IEC 15504 e Gera o Modelo de Desempenho.

Arquitetura da Linha de Processos:



Figura 34 - Linha de Processo - Estabelecer Modelo de Desempenho

Para a definição dos componentes e atividades de cada linha de processo, GONÇALVES (2014) utilizou os *templates* para definição de processos do grupo de Qualidade de Software da COPPE/UFRJ. O *layout* com os campos e descrição dos *templates* estão descritos na Tabela 8 e Tabela 9.

Tabela 8 - Template utilizado para a definição dos componentes de processo (GONÇALVES, 2014)

| <Nome do Componente> | |
|-----------------------------|--|
| Identificador: | <Identificador único do componente de processo > |
| Nome: | <Nome do componente de processo > |
| Tipo: | <Concreto/Abstrato> |
| Descrição: | <Descrição do Componente> |
| Definido por: | <Nome da Organização que definiu o Componente de Processo> |
| Crítérios de Entrada: | <Descrição do critério de entrada> |
| Crítérios de Saída: | <Descrição do critério de saída> |
| Responsável: | <Papel responsável pela execução do componente> |
| Participantes: | <Participante (s) na execução do componente, além do responsável> |
| Ferramentas de Apoio: | <Ferramentas de apoio utilizadas durante a execução do componente> |
| Artefatos Requeridos: | <Artefatos requeridos (Parâmetros de entrada do componente)> |
| Artefatos Produzidos: | <Artefatos produzidos (Parâmetros de saída do componente)> |
| Características Atendidas: | <Características atendidas pelo Componente de Processo> |
| Medidas: | <Sigla (Mnemônico) e Descrição das medidas associadas ao componente segundo Plano de Medição da Organização> |
| Variantes deste componente: | <Lista de componentes variantes, para os casos de componentes abstratos que possuam variantes> |
| Arquitetura Interna: | <Componentes ou Atividades que formam a estrutura interna do componente, caso exista > |
| Script: | <Script para execução do componente> |

Tabela 9 - Template utilizado para a definição das atividades (GONÇALVES, 2014)

| | |
|-----------------------|------------------------------------|
| Atividade: | <Nome da Atividade> |
| Descrição: | <Descrição da Atividade> |
| Crítérios de Entrada: | <Descrição do critério de entrada> |
| Crítérios de Saída: | <Descrição do critério de saída> |

| | |
|-----------------------|---|
| Saída: | |
| Responsável: | <Papel responsável pela execução da atividade> |
| Participantes: | <Participante (s) na execução da atividade, além do responsável> |
| Ferramentas de Apoio: | <Ferramentas de apoio utilizadas durante a execução da atividade> |
| Artefatos Requeridos: | <Artefatos requeridos (Parâmetros de entrada da atividade)> |
| Artefatos Produzidos: | <Artefatos produzidos (Parâmetros de saída da atividade)> |

Para a definição de um elemento de processo (componente ou atividade) GONÇALVES (2014) utilizou os seguintes critérios: (i) É relevante para ser reutilizado em diferentes definições de processos; ii) Pode ser considerado um subprocesso, podendo ser realizado de uma ou diversas maneiras; iii) Possui uma arquitetura interna contendo outros componentes de granularidade menor e/ou atividades, e; (iv) Possui atividades voltadas para verificação da estabilidade, determinação da capacidade ou construção de modelo de desempenho. A Tabela 10 apresenta uma lista completa de todos os componentes definidos.

Tabela 10 - Componentes de Processo (GONÇALVES, 2014)

| Componentes | Variantes | Arquitetura interna |
|--|--|--|
| COP.ADP.EST.ABS.0001 Verificar a distribuição dos dados da medida | COP.ADP.EST.CON.0002 Verificar a distribuição Normal dos dados com Minitab | - |
| | COP.ADP.EST.CON.0003 Verificar a distribuição de Poisson dos dados com Minitab | - |
| | COP.ADP.EST.CON.0004 Verificar a distribuição Normal dos dados com Statistica | - |
| | COP.ADP.EST.CON.0005 Verificar a distribuição de Poisson dos dados com Statistica | - |
| COP.ADP.EST.ABS.0006 Construir gráfico de controle | COP.ADP.EST.CON.0007 Construir gráfico de controle XmR com Statistica | COP.ADP.EST.CON.0008 Gerar Gráfico XmR |
| | | COP.ADP.EST.CON.0009 Aplicar testes de estabilidade padrões |
| | | COP.ADP.EST.CON.0010 Aplicar testes de estabilidade |
| | COP.ADP.EST.CON.0011 Construir gráfico de controle XmR com Minitab | COP.ADP.EST.CON.0008 Gerar Gráfico XmR |
| | | COP.ADP.EST.CON.0009 Aplicar testes de estabilidade padrões |
| | | COP.ADP.EST.CON.0010 Aplicar testes de estabilidade |
| COP.ADP.EST.CON.0012 Construir gráfico de controle | COP.ADP.EST.CON.0013 Gerar Gráfico XMmR | |

| | | |
|---|---|--|
| | XMmR com Minitab | COP.ADP.EST.CON.0010 Aplicar testes de estabilidade padrões |
| | | COP.ADP.EST.CON.0011 Aplicar testes de estabilidade |
| | COP.ADP.EST.CON.0014 Construir gráfico de controle X-bar e R com Statistica | COP.ADP.EST.CON.0015 Gerar Gráfico X-bar e R |
| | | COP.ADP.EST.CON.0009 Aplicar testes de estabilidade padrões |
| | | COP.ADP.EST.CON.0010 Aplicar testes de estabilidade |
| | COP.ADP.EST.CON.0016 Construir gráfico de controle X-bar e R com Minitab | COP.ADP.EST.CON.0015 Gerar Gráfico X-bar e R |
| | | COP.ADP.EST.CON.0009 Aplicar testes de estabilidade padrões |
| | | COP.ADP.EST.CON.0010 Aplicar testes de estabilidade |
| | COP.ADP.EST.CON.0017 Construir gráfico de controle X-bar e S com Minitab | COP.ADP.EST.CON.0018 Gerar Gráfico X-bar e S |
| | | COP.ADP.EST.CON.0009 Aplicar testes de estabilidade padrões |
| | | COP.ADP.EST.CON.0010 Aplicar testes de estabilidade |
| | COP.ADP.EST.CON.0019 Construir gráfico de controle X-bar e S com Statistica | COP.ADP.EST.CON.0018 Gerar Gráfico X-bar e S |
| | | COP.ADP.EST.CON.0009 Aplicar testes de estabilidade padrões |
| | | COP.ADP.EST.CON.0010 Aplicar testes de estabilidade |
| | COP.ADP.EST.CON.0020 Construir gráfico de controle c com Statistica | COP.ADP.EST.CON.0021 Gerar Gráfico c |
| | | COP.ADP.EST.CON.0022 Aplicar teste de estabilidade único |
| | COP.ADP.EST.CON.0023 Construir gráfico de controle c com Minitab | COP.ADP.EST.CON.0021 Gerar Gráfico c |
| | | COP.ADP.EST.CON.0022 Aplicar teste de estabilidade único |
| | COP.ADP.EST.CON.0024 Construir gráfico de controle u com Statistica | COP.ADP.EST.CON.0025 Gerar Gráfico u |
| | | COP.ADP.EST.CON.0022 Aplicar teste de estabilidade único |
| | COP.ADP.EST.CON.0026 Construir gráfico de controle u com Minitab | COP.ADP.EST.CON.0025 Gerar Gráfico u |
| | | COP.ADP.EST.CON.0022 Aplicar teste de estabilidade único |
| COP.ADP.EST.CON.0027 Aplicar testes de estabilidade complementares | - | - |
| COP.ADP.EST.ABS.0028 Identificar possíveis causas da falta de estabilidade | COP.ADP.EST.CON.0029 Identificar possíveis causas com gráfico de Pareto – Statistica | - |

| | | |
|--|---|--|
| | COP.ADP.EST.CON.0030 Identificar possíveis causas com gráfico de Pareto – Minitab | - |
| | COP.ADP.EST.CON.0031 Identificar possíveis causas com gráfico de Causa e efeito | - |
| | COP.ADP.EST.CON.0032 Identificar possíveis causas com abordagem baseada em Grounded Theory (SCHOTS, 2010) | - |
| | COP.ADP.EST.CON.0033 Identificar possíveis causas com abordagem baseada na Teoria das Restrições (COSTA, 2012) | - |
| COP.ADP.CAP.ABS.0034 Determinar capacidade | COP.ADP.CAP.CON.0035 Determinar capacidade de análise de XmR com Minitab | COP.ADP.CAP.CON.0036 Construir histograma de frequência |
| | | COP.ADP.CAP.CON.0037 Calcular índice de capacidade (Cp) |
| | COP.ADP.CAP.CON.0038 Determinar capacidade de análise de XmR com Statistica | COP.ADP.CAP.CON.0036 Construir histograma de frequência |
| | | COP.ADP.CAP.CON.0037 Calcular índice de capacidade (Cp) |
| | COP.ADP.CAP.CON.0039 Determinar capacidade de análise de XMmR com Minitab | COP.ADP.CAP.CON.0036 Construir histograma de frequência |
| | | COP.ADP.CAP.CON.0037 Calcular índice de capacidade (Cp) |
| | COP.ADP.CAP.CON.0040 Determinar capacidade de análise de XbarR com Minitab | COP.ADP.CAP.CON.0036 Construir histograma de frequência |
| | | COP.ADP.CAP.CON.0037 Calcular índice de capacidade (Cp) |
| | COP.ADP.CAP.CON.0041 Determinar capacidade de análise de XbarR com Statistica | COP.ADP.CAP.CON.0036 Construir histograma de frequência |
| | | COP.ADP.CAP.CON.0037 Calcular índice de capacidade (Cp) |
| | COP.ADP.CAP.CON.0042 Determinar capacidade de análise de c com Minitab | - |
| | COP.ADP.CAP.CON.0043 Determinar capacidade de análise de u com Minitab | - |
| COP.ADP.MOD.ABS.0044 Desenvolver modelo de desempenho | COP.ADP.MOD.CON.0045 Desenvolver modelo de desempenho através de análise de regressão – Statistica | - |
| | COP.ADP.MOD.CON.0046 Desenvolver modelo de desempenho através de análise de regressão - Minitab | - |

| | | |
|--|---|---|
| COP.ADP.MOD.CON.0047 Calibrar e testar o modelo de desempenho | - | - |
|--|---|---|

A Tabela 11 e a Tabela 12 apresentam respectivamente um exemplo de definição para a atividade “Registrar a categorização dos dados da medida” e para o componente “Construir gráfico de controle XmR com Minitab”. Os exemplos foram descritos a partir das informações contidas nos *templates* apresentados anteriormente.

Tabela 11 - Atividade “Registrar a categorização dos dados da medida” (GONÇALVES, 2014)

| | |
|-----------------------|--|
| Atividade: | Registrar a categorização dos dados da medida |
| Descrição: | Registrar a categorização dos dados coletados para a medida do subprocesso que foi selecionado para a análise de desempenho. Alguns dos critérios que podem ser utilizados para categorizar os dados da medida são: Tamanho do projeto, Versão do subprocesso, Perfil da equipe, Complexidade, Domínio de aplicação. |
| Crítérios de Entrada: | Ter-se analisado o conjunto dos dados da medida do subprocesso selecionado para a análise de desempenho. |
| Crítérios de Saída: | Ter-se registrado a categorização dos dados da medida. |
| Responsável: | Usuário do Ambiente SPEAKER. |
| Participantes: | - |
| Ferramentas de Apoio: | Ferramenta de Instanciação e Execução para a Análise de Desempenho (FIE). |
| Artefatos Requeridos: | Planilha de medidas do SPEAKER, Identificação do subprocesso selecionado. |
| Artefatos Produzidos: | Registro da categorização dos dados da medida. |

Tabela 12 - Componente de processo concreto “Construir gráfico de controle XmR com Minitab” (GONÇALVES, 2014)

| Construir gráfico de controle XmR com Minitab | |
|--|---|
| Identificador: | COP.ADP.EST.CON.0011 |
| Nome: | Construir gráfico de controle XmR com Minitab |
| Tipo: | Concreto |
| Descrição: | Construir o gráfico de controle XmR usando o software Minitab, de acordo com os passos abaixo: i) calcular os limites superior e inferior, calcular a linha central e plotar os dados do gráfico X que representa o gráfico sequencial para os valores individuais; ii) aplicar os testes de estabilidade 3-sigma, linha central, 2-sigma e 1-sigma para o gráfico X; iii) calcular o limite superior, calcular a linha central e plotar os dados do gráfico mR que representa o gráfico sequencial para as amplitudes móveis; iv) aplicar os testes de estabilidade 3-sigma e linha central para o gráfico mR. |
| Definido por: | COPPE/UFRJ |
| Crítérios de Entrada: | Ter-se um conjunto de dados previamente analisado e adequado ao tipo de gráfico de controle escolhido. |
| Crítérios de Saída: | Ter-se o gráfico de controle XmR construído e os testes de estabilidade realizados. |
| Responsável: | Usuário do Ambiente SPEAKER. |
| Participantes: | - |
| Ferramentas de Apoio: | Software Minitab |
| Artefatos Requeridos: | Planilha de medidas do SPEAKER. |

| | |
|----------------------------|--|
| Artefatos Produzidos: | Gráfico de controle XmR com o resultado dos testes de estabilidade. |
| Características Atendidas: | MR-MPS-SW – Nível B; CMMI-DEV – nível 4; ISO/IEC 15504; Verificação da Estabilidade; Gráfico de controle; Dados de atributos; Dados de variáveis; Distribuição normal; Uma observação (sem subgrupo); Uso do Minitab. |
| Medidas: | - |
| Variante deste componente: | - |
| Arquitetura Interna: | |
| Script: | <p>GraficoXmR.mtb</p> <p>Name C2 "Média X" C3 "Desvio Padrão X" C4 "Ponto Plotado X" C5 "Ponto Plotado mR" C6 "Linha Central X" & C7 "Linha Central mR" C8 "Limite Inferior X" C9 "Limite Superior X" C10 "Limite Inferior mR" & C11 "Limite Superior mR" C12 "Teste 3-sigma X" C13 "Teste linha central X" C14 "Teste 2-sigma X" & C15 "Teste 1-sigma X" C16 "Teste 3-sigma mR" C17 "Teste linha central mR".</p> <pre> IMRChart 'Valores Coletados'; Title "Gráfico XmR"; Test 1 2 5 6; DefTest 2 8; Location 'Média X'; Variation 'Desvio Padrão X'; PPoints 'Ponto Plotado X' - 'Ponto Plotado mR'; CenLine 'Linha Central X' - 'Linha Central mR'; ConLimits 'Limite Inferior X' - 'Limite Superior mR'; TRResults 'Teste 3-sigma X' - 'Teste linha central mR'. </pre> |
| Gerar Gráfico XmR | |
| Identificador: | COP.ADP.EST.CON.0008 |
| Nome: | Gerar Gráfico XmR |
| Tipo: | Concreto |
| Descrição: | <p>Gerar o gráfico XmR de acordo com os passos abaixo. Neste tipo de gráfico, o gráfico X representa os valores individuais e o gráfico mR (moving range) representa a variação de um valor em relação ao valor anterior. Este tipo de gráfico é apropriado para analisar o comportamento de um processo que uma mesma medida é coletada frequentemente. Além disso, este tipo de gráfico se aplica a dados de variáveis e dados de atributos e não suporta subgrupos com mais de uma observação.</p> <p>1. Calcular o limite superior e a linha central do Gráfico mR utilizando as fórmulas e os passos abaixo:</p> <p>1.1 Calcular a média das amplitudes móveis \overline{mR}:</p> $\overline{mR} = \frac{\sum_{i=1}^{k-1} mR_i}{k-1}$ <p>Onde: k é número de observações</p> <p>E $mR_i = x_{i+1} - x_i$ representa o valor de cada amplitude móvel, sendo i inteiro e $1 < i < k-1$</p> <p>1.2 Calcular os limites do gráfico mR a partir das seguintes equações:</p> |

| | |
|---|---|
| | <p> Linha Central = \overline{mR} Limite Superior = $D_4 \cdot \overline{mR}$ </p> <p>Onde: $D_4 = 3,267$</p> <p>2. Calcular os limites superior e inferior, e a linha central do Gráfico X utilizando as fórmulas e os passos abaixo:</p> <p>2.1 Calcular a média \bar{x}:</p> $\bar{x} = \frac{\sum x}{k} \qquad \overline{mR} = \frac{\sum_{i=1}^{k-1} mR_i}{k-1}$ <p>Onde: k é número de observações</p> <p>2.2 Calcular os limites do gráfico X a partir das seguintes equações:</p> <p> Linha Central = \bar{x} onde: $E_2 = 2,660$ Limite Superior = $\bar{x} + E_2 \cdot \overline{mR}$ Limite Inferior = $\bar{x} - E_2 \cdot \overline{mR}$ </p> <p>Onde: $E_2 = 2,660$</p> <p>3. Plotar os dados no gráfico.</p> |
| Definido por: | COPPE/UF RJ |
| Cr itérios de Entrada: | Ter-se um conjunto de dados previamente analisado e adequado ao tipo de gráfico de controle escolhido. |
| Cr itérios de Saída: | Ter-se o gráfico de controle XmR construído. |
| Responsável: | Usuário do Ambiente SPEAKER. |
| Participantes: | - |
| Ferramentas de Apoio: | Software Statistica ou Software Minitab. |
| Artefatos Requeridos: | Planilha de medidas do SPEAKER. |
| Artefatos Produzidos: | Gráfico de controle XmR. |
| Características Atendidas: | Verificação da Estabilidade; Gráfico de controle; Dados de atributos; Dados de variáveis; Uma observação (sem subgrupo); Uso do Statistica; Uso do Minitab. |
| Medidas: | - |
| Variantes deste componente: | - |
| Arquitetura Interna: | - |
| Script: | - |
| Aplicar testes de estabilidade padrões | |
| Identificador: | COP.ADP.EST.CON.0009 |
| Nome: | Aplicar testes de estabilidade padrões |
| Tipo: | Concreto |
| Descrição: | Aplicar os testes de estabilidade 3-sigma, linha central, 2-sigma e 1-sigma para verificar se o subprocesso analisado é estável. O teste de estabilidade 3-sigma verifica se há algum valor além dos limites de controle 3-sigma, fornecendo a maior evidência da falta de estabilidade do processo. O teste de estabilidade linha central verifica se há pelo menos oito valores consecutivos localizados de um mesmo lado da linha central, fornecendo mais sensibilidade da falta de estabilidade do processo. O teste de estabilidade 2-sigma verifica se há pelo menos dois de três valores consecutivos do mesmo lado e a mais de 2-sigma da linha central, fornecendo evidência de pequenas mudanças no processo. O teste de estabilidade 1-sigma verifica se há pelo menos quatro de cinco valores consecutivos do mesmo lado e a mais de 1-sigma da linha central, fornecendo evidência de de pequenas mudanças no processo. Estes testes são aplicáveis aos gráficos X e Xbar. Se o gráfico não falhar em |

| | |
|---------------------------------------|---|
| | nemhum dos testes descritos o subprocesso é considerado estável e pode-se gerar a baseline do subprocesso. Mas caso o gráfico falhe em um dos testes, uma investigação deve ser realizada para verificar quais são as causas da falta de estabilidade do processo. |
| Definido por: | COPPE/UFRJ |
| Critérios de Entrada: | Ter-se construído o gráfico. |
| Critérios de Saída: | Ter-se aplicado os testes de estabilidade. |
| Responsável: | Usuário do Ambiente SPEAKER. |
| Participantes: | - |
| Ferramentas de Apoio: | Software Statistica ou Minitab. |
| Artefatos Requeridos: | Gráfico. |
| Artefatos Produzidos: | Gráfico com o resultado dos testes de estabilidade. |
| Características Atendidas: | Verificação da Estabilidade, Dados de atributo; Dados de variáveis; Distribuição Normal; Uma observação (sem subgrupo); De duas a dez observações por subgrupo; Mais de dez observações por subgrupo; Uso do Statistica; Uso do Minitab. |
| Medidas: | - |
| Variantes deste componente: | - |
| Arquitetura Interna: | - |
| Script: | - |
| Aplicar testes de estabilidade | |
| Identificador: | COP.ADP.EST.CON.0010 |
| Nome: | Aplicar testes de estabilidade |
| Tipo: | Concreto |
| Descrição: | Aplicar os testes de estabilidade 3-sigma e linha central para verificar se o subprocesso analisado é estável. O teste de estabilidade 3-sigma verifica se há algum valor além dos limites de controle 3-sigma, fornecendo a maior evidência da falta de estabilidade do processo. O teste de estabilidade linha central verifica se há pelo menos oito valores consecutivos localizados de um mesmo lado da linha central, fornecendo mais sensibilidade da falta de estabilidade do processo. Estes testes são aplicáveis aos gráficos mR, MmR, R e S. Se o gráfico não falhar em nenhum dos testes descritos o subprocesso é considerado estável e pode-se gerar a baseline do subprocesso. Mas caso o gráfico falhe em um dos testes, uma investigação deve ser realizada para verificar quais são as causas da falta de estabilidade do processo. |
| Definido por: | COPPE/UFRJ |
| Critérios de Entrada: | Ter-se construído o gráfico. |
| Critérios de Saída: | Ter-se aplicado os testes de estabilidade. |
| Responsável: | Usuário do Ambiente SPEAKER. |
| Participantes: | - |
| Ferramentas de Apoio: | Software Statistica ou Minitab. |
| Artefatos Requeridos: | Gráfico. |
| Artefatos Produzidos: | Gráfico com o resultado dos testes de estabilidade. |
| Características Atendidas: | Verificação da Estabilidade, Dados de atributo; Dados de variáveis; Distribuição Normal; Uma observação (sem subgrupo); De duas a dez observações por subgrupo; Mais de dez observações por subgrupo; Uso do Statistica; Uso do Minitab. |
| Medidas: | - |
| Variantes deste componente: | - |
| Arquitetura Interna: | - |
| Script: | - |

A descrição de todos os componentes, atividades e linhas de processo podem ser encontradas em GONÇALVES (2014).

3.5 Considerações Finais

Este capítulo apresentou o ambiente SPEAKER, detalhando os principais módulos e explicando o contexto em que esta dissertação se enquadra dentro do ambiente, o A2M. Também foram apresentados alguns conceitos da abordagem de BARRETO (2011) que serão utilizados no próximo capítulo e as linhas de processos que serão executadas pela ferramenta FIE. Para demonstrar a execução das linhas de processo, serão descritos exemplos de utilização da ferramenta FIE no capítulo 5 com exemplos que utilizam cada uma das linhas descritas acima.

CAPÍTULO 4 - FERRAMENTA PARA INSTANCIÇÃO E EXECUÇÃO DO PROCESSO DE ANÁLISE DE DESEMPENHO – FIE

Este capítulo descreve as funcionalidades da ferramenta para instanciação e execução do processo de análise de desempenho para processos de software, objetivo desta dissertação.

4.1 Introdução

As ferramentas disponíveis para a execução do processo de análise de desempenho nas organizações auxiliam apenas na geração dos gráficos de controle, como, por exemplo, o Minitab (Minitab Inc., 2013) e o Statistica (StatSoft Inc., 2013), ou apoiam a execução por meio de outros ambientes, que é o caso da WebAPSEE (GONÇALVES, 2012 e REIS, 2003). A Ferramenta para Instanciação e Execução do Processo de Análise de Desempenho (FIE), tem como objetivo apoiar a execução do processo de Análise de Desempenho de processos de software dentro do ambiente SPEAKER.

A ferramenta FIE foi desenvolvida dentro do ambiente A2M utilizando todos os padrões detalhados no Capítulo 3. As linhas de processo que serão executadas foram definidas por GONÇALVES (2014) utilizando a ferramenta A2MComponenteProcesso, desenvolvida por BARRETO (2011). Toda a comunicação entre a ferramenta FIE e a ferramenta A2MComponenteProcesso foi realizada por meio de métodos já disponíveis dentro da aplicação, ou seja, nenhuma alteração foi implementada para a realização desta comunicação.

Este capítulo apresenta a estrutura definida para o desenvolvimento da ferramenta FIE por meio de seus requisitos, funcionalidades e o processo definido para atender a definição dinâmica do processo de análise de desempenho de acordo com o contexto organizacional e com as características do processo analisado.

A seção 4.2 apresenta os requisitos definidos a partir dos requisitos do ambiente SPEAKER. A seção 4.3 apresenta a estrutura definida para armazenar e auxiliar o usuário na execução do processo de análise de desempenho, além das regras de

validação que foram implementadas e a interface definida para a comunicação com o SBC. A seção 4.4 apresenta o processo definido que foi implementado pela ferramenta para possibilitar a definição dinâmica do processo e a seção 4.5 apresenta as considerações finais sobre este capítulo.

4.2 Requisitos para Definição da Ferramenta

O objetivo principal da ferramenta FIE é apoiar a instanciação e a execução do processo para análise de desempenho de processos de software. A partir deste objetivo e de um estudo realizado sobre o processo de Análise de Desempenho para a definição do ambiente SPEAKER (Capítulo 3), foram definidos e derivados os requisitos da ferramenta FIE. Esses requisitos são descritos a seguir:

- REQ01 – A ferramenta FIE deve apoiar a instanciação de processos de análise de desempenho a partir de elementos de processo disponíveis na Biblioteca de Elementos de Processos Reutilizáveis, e de acordo com a análise realizada pelo SBC.
- REQ02 – A ferramenta FIE deve apoiar a execução das atividades da análise de desempenho e armazenar os resultados desta execução;
- REQ03 – A ferramenta FIE deve disponibilizar os resultados da execução das atividades para o SBC.
- REQ04 – A ferramenta FIE deve ser capaz de controlar a execução de uma única atividade do processo de análise de desempenho mais de uma vez, de acordo com as características do subprocesso analisado.
- REQ05 - A ferramenta FIE só executa componentes de processo concretos, ou seja, se em algum momento for solicitada a execução de um componente de processo abstrato, uma mensagem de erro será enviada, informando sobre o ocorrido.
- REQ06 – A ferramenta FIE implementa apenas dois níveis de execução para a arquitetura de processo interna de um componente, isto significa que caso um componente de processo possua em sua arquitetura de processo interna outro componente de processo com arquitetura de processo interna definida, somente o primeiro nível será executado. Esta regra foi criada pois poderia gerar um número muito grande de níveis de execução.

- REQ07 – Para encerrar a execução de uma análise linha de processo específica, todos os elementos de processo executáveis devem possuir o atributo referente à utilização do resultado na definição do resultado da análise. Esta regra foi criada para que se possa gerar a versão final do processo executado.
- REQ08 – No momento da geração da versão final da execução do processo, caso um mesmo elemento de processo tenha sido executado mais de uma vez, e por algum erro na comunicação com o SBC o atributo que indica sua utilização estiver marcado com SIM, como padrão a ferramenta FIE utilizará o último resultado executado. A geração da versão final do processo executado é explicada no item 4.4 deste capítulo.

Os detalhes dos requisitos gerais serão descritos nas próximas seções juntamente com seu desenvolvimento.

4.3 Estrutura para Instanciação e Execução do Processo de Análise de Desempenho

Alguns conceitos foram criados e adotados para possibilitar a instanciação e a execução do processo de análise de desempenho e estruturar a análise dentro da ferramenta. Esses conceitos⁴ serão explicados e apresentados nos modelos definidos para a ferramenta.

A Figura 35 apresenta o modelo definido para estruturar a análise de desempenho e possui os seguintes conceitos associados: *análise subprocesso*, *análise medida*, *análise linha de processo e versão*. O conceito de *análise subprocesso* está relacionado com a organização que definiu o subprocesso de software que está sendo avaliado e com o subprocesso avaliado. A *análise medida* está relacionada à medida que está sendo avaliada, uma *análise subprocesso* possui uma ou mais *análises medidas*, isto porque um subprocesso pode ser definido como estável com relação a uma medida específica. A *análise linha de processo* refere-se a linha de processo, que são linhas de produto onde os produtos são processos em um dado nível de granularidade (BARRETO, 2011), que será executada pela ferramenta FIE. Uma *análise linha de*

⁴ Será utilizada notação com os termos em letras minúsculas e em itálico para se referir a conceitos presentes na modelagem proposta ao longo do texto.

processo está associada a uma *análise medida* e uma *análise medida* possui uma ou mais *análises linhas de processo* associadas, isto porque mais de uma Linha de Processo pode ser executada, como, por exemplo, a linha de processo “Verificar estabilidade” e “Verificar a capacidade” podem ser executadas para a medida “Quantidade de horas diárias despendidas em manutenção” com os valores coletados para o subprocesso “Manutenção de projetos” da organização “Empresa XYZ”. Uma *análise linha de processo* possui data início, data fim e um *status análise* para controlar sua execução. O *status análise* pode ser “Em execução” ou “Finalizado”.

Como a execução das linhas de processo será realizada dinamicamente, a ferramenta FIE deve suportar diversas execuções do mesmo *elemento de processo*. Um *elemento de processo* segundo a abordagem de BARRETO (2011) é um encapsulamento de informações e comportamentos de processo em um dado nível de granularidade. Representa uma ação de transformação realizada por um processo. Para implementar esta funcionalidade, a ferramenta FIE possui o conceito de *versão*, que está relacionada a uma *análise linha de processo*, onde uma *análise linha de processo* possui uma ou mais versões do processo executadas. Uma *versão* é criada nas seguintes situações: (i) No início da instanciação da linha de processo, ou; (ii) Quando um *elemento de processo* for executado mais de uma vez.

Para implementar a definição de scripts para a geração dos gráficos de controle (GONÇALVES, 2014) foi necessária a alteração da ferramenta A2MComponenteProcesso, onde foi incluído um novo atributo em *elemento de processo*. Este atributo recebeu o nome de *script* e contém o arquivo criado para ser executado pela ferramenta descrita no cadastro do *elemento de processo*. Por sua vez, alguns scripts necessitavam de parametrização, ou seja, antes da execução o usuário deve informar o valor de algumas informações importantes, como, por exemplo, o número de subgrupos definidos. Para implementar esta alteração, uma nova entidade chamada *parâmetros script* foi criada, e a tela de cadastro do *elemento de processo*, agora possibilita o usuário a informar o script e quais os seus parâmetros. A Figura 36 apresenta essa alteração.

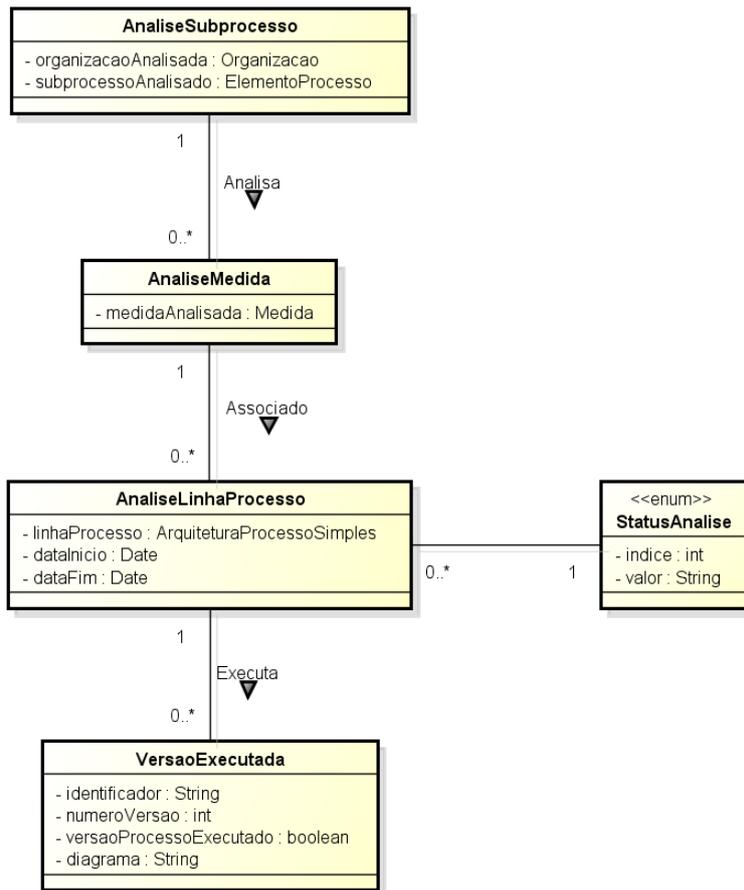


Figura 35 - Análise Subprocesso, Análise Medida, Análise Linha de Processo e Versão Executada

Cadastro de Componentes de Processo

Editar Arquitetura Interna

Confirmar **Cancelar**

Dados Básicos | Características | Variação | Medidas | Baselines de Desempenho | Script

Nome Arquivo: GraficoXbarS1.mtb Tamanho(bytes): 881

Script:

Parâmetros do Script

| Parâmetro | Descrição |
|----------------|-------------------------|
| P_QTD_SUBGRUPO | Quantidade de subgrupos |

Confirmar **Cancelar**

Figura 36 - A2MComponenteProcesso - Alteração para cadastro de script

Segundo a abordagem definida por BARRETO (2011), um *elemento de processo* pode ser de um dos seguintes tipos: *componente de processo* ou *atividade*. Os dois conceitos têm significados similares, sendo que a principal diferença é que um *componente* é definido para reutilização e uma *atividade* é completamente definida. Um *componente de processo* pode ser *abstrato*, quando este possui variabilidades ou *concreto* quando não admite variação, ou seja, precisa ser executado da forma que está descrito. Um outro conceito utilizado é o de *arquitetura de processo*, que é uma organização de *elementos de processo*, que fornece os elementos, ordenação, interfaces e interdependências entre os *elementos de processo*. Um *componente de processo* pode possuir uma *arquitetura interna de processo*, isto significa que para executar o *componente de processo* em questão todos os *elementos de processo* contidos em sua *arquitetura de processo* devem ser executados.

A ferramenta FIE executa um *elemento de processo*, sendo assim, implementa os conceitos de *elemento de processo executável*, onde este pode ser um *componente de processo executável* ou uma *atividade de processo executável*. Um *elemento de processo executável* está relacionado a um *elemento de processo*. A Figura 37 apresenta o modelo definido para a ferramenta FIE que contempla esses conceitos.

Um *elemento de processo executável* por sua vez precisa armazenar os resultados de sua execução, para isso a entidade *resultado execução* foi definida no modelo. Um *elemento de processo executável* possui um ou mais registros de *resultados de execução* e um *resultado de execução* pode estar relacionado a um artefato definido para o *elemento de processo* associado. Para exemplificar, supondo que a *atividade* “Registrar a categorização dos dados da medida” estivesse sendo executada, teríamos um objeto *atividade processo executável* que faria referência à atividade. Esta atividade possui um artefato de saída chamado “Registro da categorização dos dados da medida”. No cadastro deste *resultado de execução* o usuário da ferramenta FIE poderá associá-lo a este artefato. Um *resultado de execução* pode ser dos seguintes tipos: Imagem, Script, Texto ou Planilha.

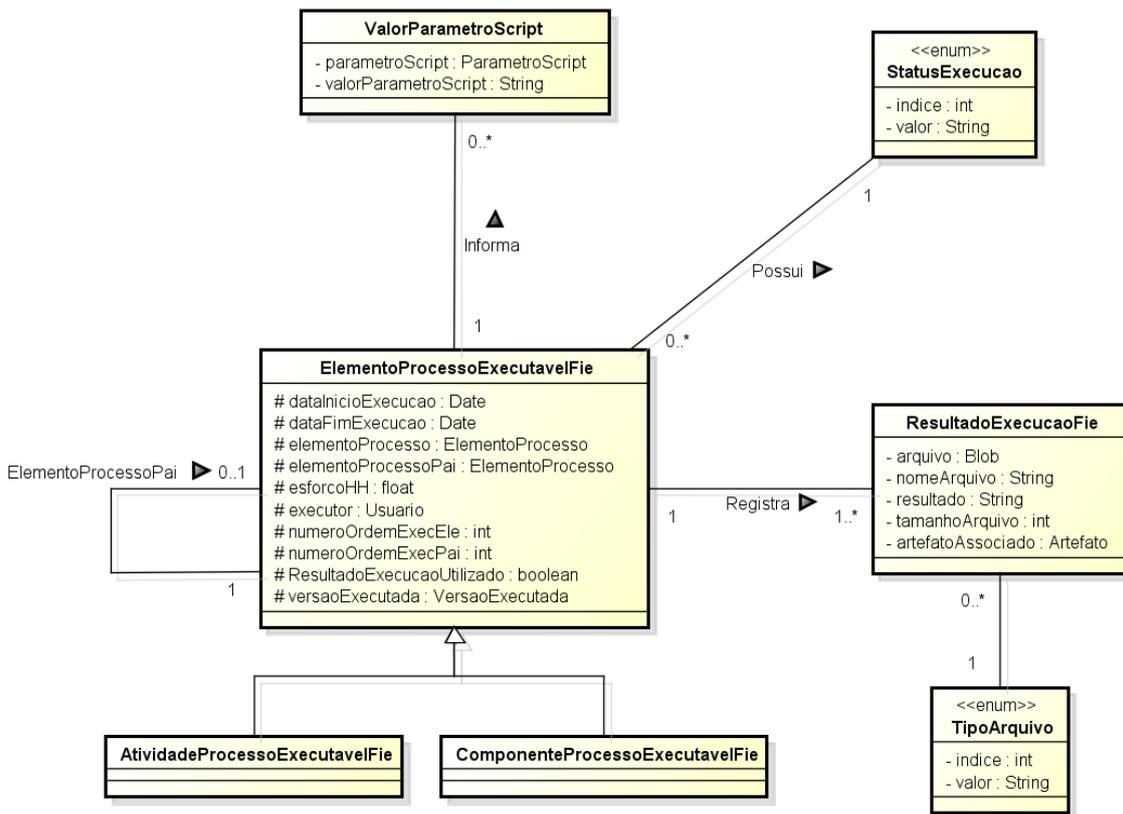


Figura 37 - Elemento de Processo Executável, Resultado Execução, Valor Parâmetro Script

O *elemento de processo executável* possui um *status execução* associado, que informa ao usuário em qual estado se encontra a execução deste elemento. Os possíveis estados de um *elemento de processo executável* são: Em execução, Finalizado, Cancelado, Parado e Aguardando e foram baseados em REIS (2003). A Figura 38 apresenta o diagrama de estados para um *elemento de processo executável*. Inicialmente, um *elemento de processo executável* possui o estado de *Aguardando*, a partir deste estado, o mesmo só poderá ser alterado para *Em execução*, quando o usuário iniciar a execução do *elemento de processo executável*. Após esta alteração os próximos estados possíveis são: (i) *Finalizado* quando a execução for finalizada; (ii) *Parado* quando o usuário por algum motivo deseja parar a execução do *elemento de processo executável*, ou; (iii) *Cancelado* quando o usuário deseja cancelar a execução. E, por último, do estado *Parado* o *elemento de processo executável* pode ir para *Finalizado* ou para *Cancelado*.

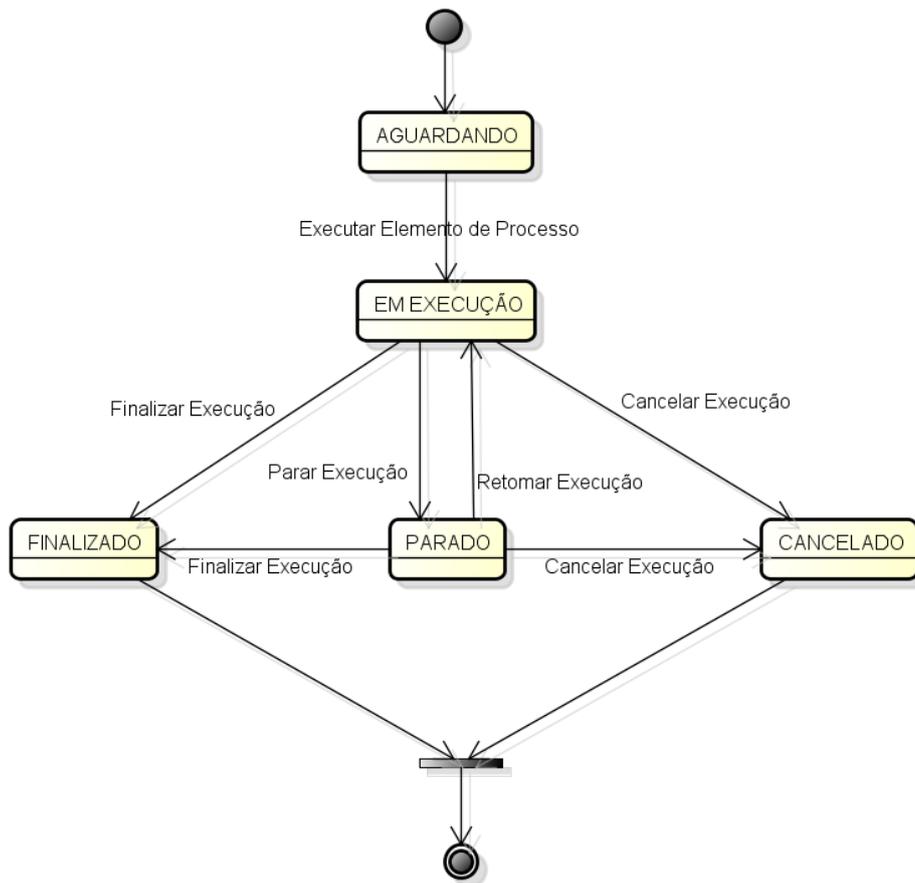


Figura 38 - Estados de Execução de um Elemento de Processo Executável

O conceito de *valor do parâmetro de script* foi criado para que no momento da execução de um *elemento de processo executável* que possua script associado, o usuário possa informar os valores dos parâmetros definidos no momento do cadastro do *elemento de processo*. O capítulo 5 descreve um exemplo de uso da ferramenta onde essas informações foram mais detalhadas e exemplificadas.

4.3.1 Principais Funcionalidades Implementadas

Para o correto funcionamento da ferramenta FIE as seguintes funcionalidades foram definidas:

- Acompanhar Execução do Processo – esta funcionalidade permite acompanhar todas as execuções de análise de desempenho solicitadas pelo SBC que se encontram em execução. Através desta funcionalidade as informações de execução são preenchidas para cada *elemento de processo* executado.

- Consultar Análises Realizadas – esta funcionalidade permite a consulta de todas as análises de desempenho realizadas, independente do seu *status*. É possível consultar as informações sobre as *versões* geradas para cada análise e os *resultados de execução* armazenados para cada *elemento de processo* executado dentro da versão. Caso a análise esteja finalizada, obrigatoriamente possuirá uma *versão* final executada também disponível para consulta.
- Consultar Versão Final Executada – esta funcionalidade permite a consulta das informações apenas das análise finalizadas. Somente a *versão* final encontra-se disponível para consulta nesta funcionalidade.

Para as funcionalidades “Acompanhar Execução do Processo” e “Consultar Análises Realizadas”, é possível verificar as informações das Análises de Desempenho Realizadas (Organização e Subprocesso), das Medidas Analisadas (Medida), das Linhas de Processo Instanciadas (Linha de Processo) e das Versões Executadas (Número da Versão, Indicador de Versão Final). Para cada *versão* executada, é possível verificar as informações dos *elementos de processo* executados (Tipo Elemento, Nome, Data Início, Data Fim e Status) além dos *resultados de execução* armazenados para cada *elemento de processo* executado. A funcionalidade “Consultar Versão Final Executada” não permite a consulta de todas as versões executadas para uma linha de processo instanciada, somente a versão final será exibida.

O Capítulo 5 demonstra as funcionalidades e validações listadas nesta seção através da execução de um cenário de teste para avaliar a ferramenta FIE. As validações também são apresentadas na subseção 4.4, na descrição das atividades do processo para Definição de Processo de Análise de Desempenho.

4.3.2 Interface com o Sistema Baseado em Conhecimento – SBC

Conforme explicado no início deste capítulo, a execução do processo de análise de desempenho é guiada pela necessidade de informações do SBC. Para implementar esta comunicação, foi criado o componente (*managed bean*) *GerenciaInterfaceSBC*. Esta classe da aplicação possui métodos que permitirão ao SBC enviar as informações necessárias para a instanciação e a execução correta do processo de análise de

desempenho e receber as informações sobre os resultados obtidos através da execução dos elementos de processo. Os métodos definidos para esta comunicação são:

- **incluirElementoExecucao**: Método responsável por gerar toda a estrutura necessária para controle da execução do processo (*análise subprocesso, análise medida, análise linha de processo, versão e elemento de processo executável*). Este método retorna o Elemento de Processo (*elemento de processo executável*) que será executado pela ferramenta FIE. Os requisitos REQ01, REQ02, REQ04, REQ05 e REQ06 são implementados. A Tabela 13 apresenta os parâmetros necessários para a chamada deste método.

Tabela 13 - Parâmetros do método incluir elemento execução

| Parâmetro | Descrição | Obrigatório |
|----------------------|---|-------------|
| Organização | Organização responsável pela definição do subprocesso analisado. Deverá ser um objeto do tipo <code>OrganizacaoSimples</code> . | Sim |
| Subprocesso | Subprocesso que será analisado. Deverá ser um objeto do tipo <code>ElementoProcessoSimples</code> . | Sim |
| Medida | Medida que será analisada. Deverá ser um objeto do tipo <code>MedidaSimples</code> . | Sim |
| Linha de Processo | Linha de processo que será executada. Deverá ser um objeto do tipo <code>ArquiteturaProcessoIndependente</code> . | Sim |
| Elemento de Processo | Elemento de Processo da Linha de processo em questão que será executado. Deverá ser um objeto do tipo <code>ElementoProcesso</code> . | Sim |
| Arquitetura Interna | Arquitetura interna, no caso do elemento de processo ser um componente de processo. A arquitetura de processo é solicitada, pois somente serão aceitos atividades e componentes de processo concretos. Deverá ser uma lista de objetos do tipo <code>ItemArquiteturaProcesso</code> . | Não |

A Figura 39 apresenta o diagrama de sequência com as interações entre as entidades da ferramenta FIE com o SBC. Esta interação é controlada pela interface *GerenciaInterfaceSBC*. O fluxo exibido na Figura 39 demonstra as mensagens enviadas e recebidas utilizadas na implementação do método *incluirElementoExecucao*.

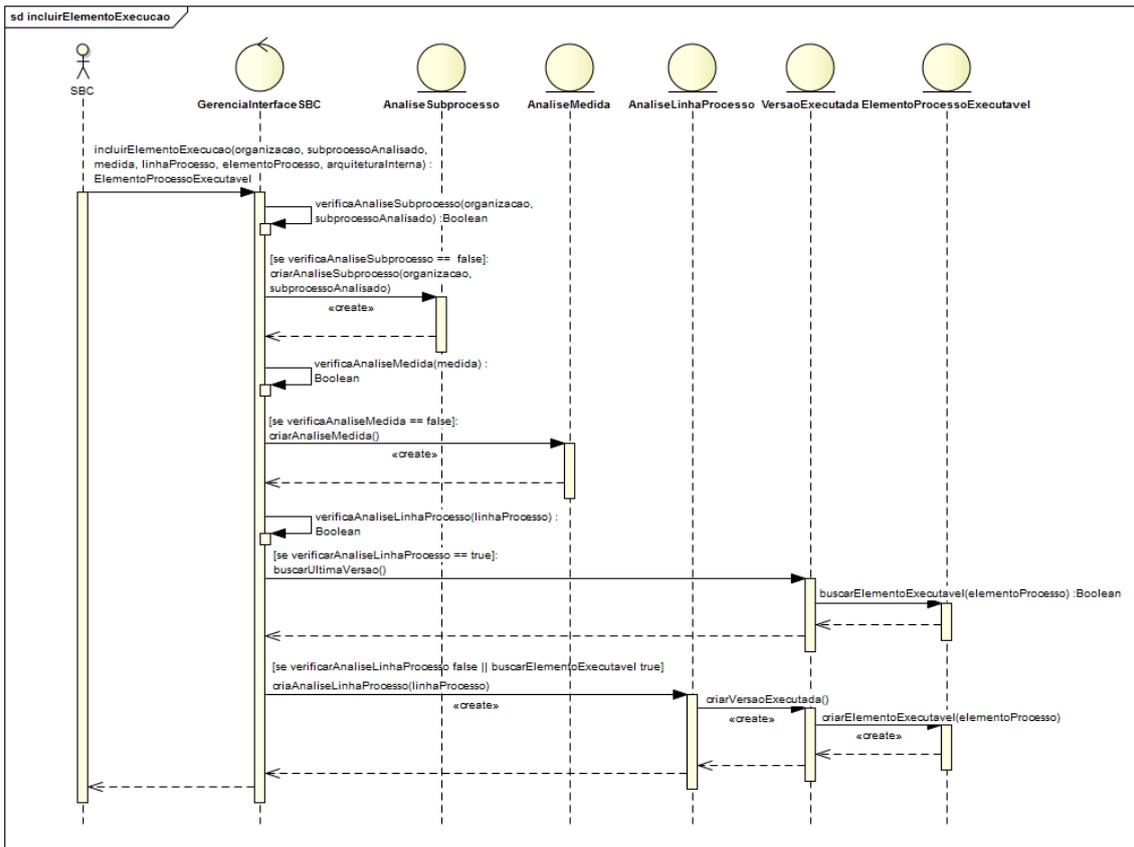


Figura 39 - Diagrama de Sequência - *incluirElementoExecucao*

- atualizarResultadoExecucao: Método responsável por atualizar o atributo do *elemento de processo executável* que faz referência ao uso do resultado da execução na definição do resultado da análise. Este método faz referência ao requisito REQ07. A Tabela 14 apresenta os parâmetros necessários para a chamada deste método.

Tabela 14 - Parâmetros do método atualizar resultado execução

| Parâmetro | Descrição | Obrigatório |
|---------------------------------|---|-------------|
| Elemento de Processo Executado | O elemento de processo que foi executado e será atualizado. Retornado da chamada ao método <i>incluirElementoExecucao</i> . | Sim |
| Resultado da Execução Utilizado | Um indicador de que o resultado gerado pela ferramenta FIE foi satisfatório para a execução da análise. | Sim |

A Figura 40 apresenta o diagrama de sequência com o fluxo de mensagens enviadas e recebidas para a implementação do método *atualizarResultadoExecucao*.

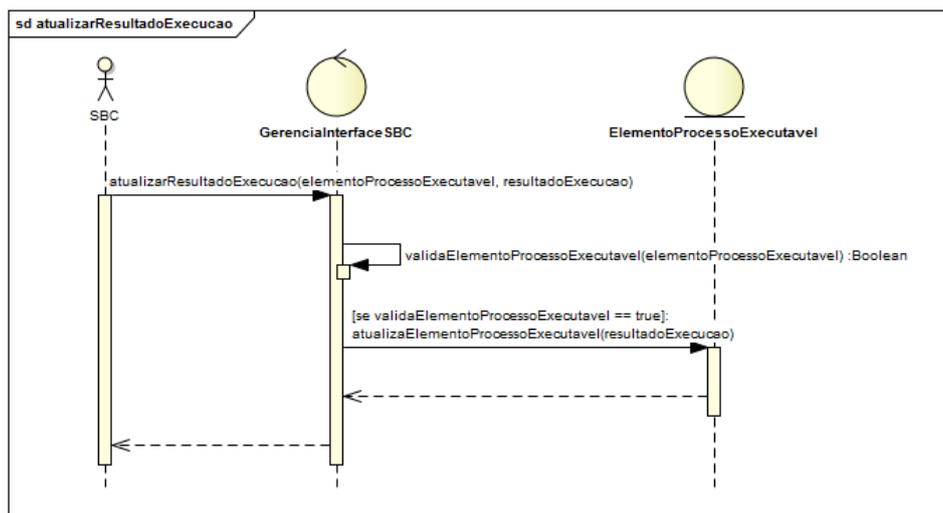


Figura 40 - Diagrama de Sequência - *alterarResultadoExecucao*

- consultarResultadosExecucao: Método responsável por retornar uma lista contendo os resultados da execução de um *elemento de processo executável*. A Tabela 15 apresenta os parâmetros necessários para a chamada deste método. Este método faz referência ao requisito REQ03.

Tabela 15 - Parâmetros do método consultar resultados execução

| Parâmetro | Descrição | Obrigatório |
|--------------------------------|---|-------------|
| Elemento de Processo Executado | O elemento de processo que foi executado. Retornado da chamada ao método <code>incluirElementoExecucao</code> . | Sim |

A Figura 41 apresenta o diagrama de sequência com o fluxo de mensagens enviadas e recebidas para a implementação do método *consultarResultadosExecucao*.

Além dos diagramas de sequência para cada método da interface de comunicação com o SBC, foi criada uma tabela com a matriz de rastreabilidade para demonstrar o relacionamento entre os requisitos da ferramenta FIE e os métodos implementados para a interface de comunicação com o SBC. Estes relacionamentos são apresentados na Tabela 16.

O REQ08 não possui relacionamento com os métodos pois é realizado diretamente pelo usuário da ferramenta FIE.

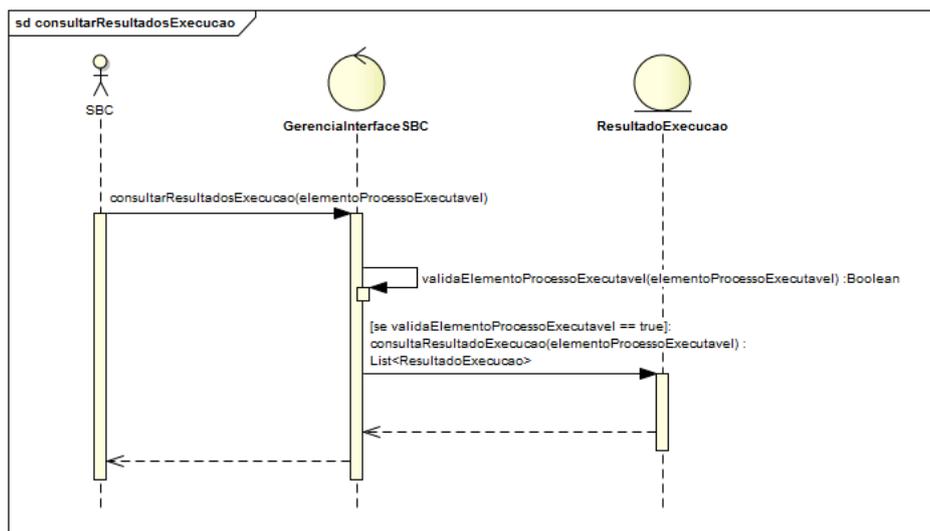


Figura 41 - Diagrama de Sequência - *consultarResultadosExecucao*

Tabela 16 - Matriz de Rastreabilidade dos requisitos da ferramenta FIE com os métodos de comunicação com o SBC

| Requisito | Método |
|-----------|-----------------------------|
| REQ01 | incluirElementoExecucao |
| REQ02 | incluirElementoExecucao |
| REQ03 | consultarResultadosExecucao |
| REQ04 | incluirElementoExecucao |
| REQ05 | incluirElementoExecucao |
| REQ06 | incluirElementoExecucao |
| REQ07 | atualizarResultadoExecucao |
| REQ08 | - |

A partir dos métodos definidos para comunicação com o SBC, a ferramenta FIE é capaz de gerenciar toda a execução do processo de análise de desempenho de acordo com os requisitos definidos para o ambiente e para a ferramenta.

4.4 Definição do Processo de Análise de Desempenho

As atividades da análise de desempenho podem ser repetidas mais de uma vez, variando de acordo com as características do subprocesso a ser analisado. Além da repetição das atividades para cada subprocesso que precisa ser analisado de acordo com as necessidades da organização, a execução do processo de análise de desempenho para um único subprocesso não é linear e sim iterativa, permitindo que uma determinada atividade seja executada mais de uma vez em uma mesma análise.

Para executar a análise de desempenho de processos, é necessário ter um processo definido. Por este motivo, neste trabalho, um processo foi definido para instanciar e executar o processo de análise de desempenho, a partir das linhas de processos definidas por GONÇALVES (2014).

A execução do processo de análise de desempenho é apoiada pela ferramenta FIE, que recebe demandas do SBC quanto à técnica a ser utilizada para análise de desempenho, seleciona o elemento de processo adequado da Biblioteca de Elementos de Processos Reutilizáveis para composição da linha de processo e executa o processo definido.

Para apoiar a execução do processo de Análise de Desempenho, a ferramenta FIE possui um processo que é apresentado na Figura 42 e será descrito nas próximas seções.

A execução deste processo com a ferramenta FIE, permite que o processo para a análise de desempenho seja definido dinamicamente de acordo com o contexto organizacional e com as características do processo analisado. As atividades do processo serão executadas em sua maioria por funcionalidades internas da ferramenta FIE. Por este motivo, não possuem responsáveis, participantes, artefatos requeridos ou produzidos.

A execução do processo é iniciada através de uma solicitação de execução enviada pelo SBC através do método *incluirElementoExecução* desenvolvido na interface com o SBC. Após essa solicitação, a ferramenta FIE avalia se as informações enviadas do subprocesso (organização, subprocesso, medida, linha de processo) já se encontram cadastradas em sua base. A partir destas informações, a ferramenta FIE verifica a necessidade de instanciar uma nova execução ou continuar na versão executada. Após a validação, a ferramenta FIE seleciona o elemento de processo na Base de Elementos de Processos Reutilizáveis e o disponibiliza para execução, apresentando as suas informações e permitindo o cadastro dos resultados. O próximo passo a ser realizado é armazenar os resultados e disponibilizá-los para o SBC. Após a disponibilização dos resultados através do método *consultarResultadosExecução*, a ferramenta FIE aguarda a resposta do SBC quanto à utilização dos resultados da execução do elemento de processo.

O SBC informa o resultado da utilização através do método *atualizarResultadoExecução*, esta informação é atualizada na base da ferramenta FIE.

Este resultado será utilizado na geração final da versão executada do processo de análise de desempenho. O processo apresentado na Figura 42 será exemplificado no Capítulo 5 através de um exemplo de execução deste processo em um cenário de estudo.

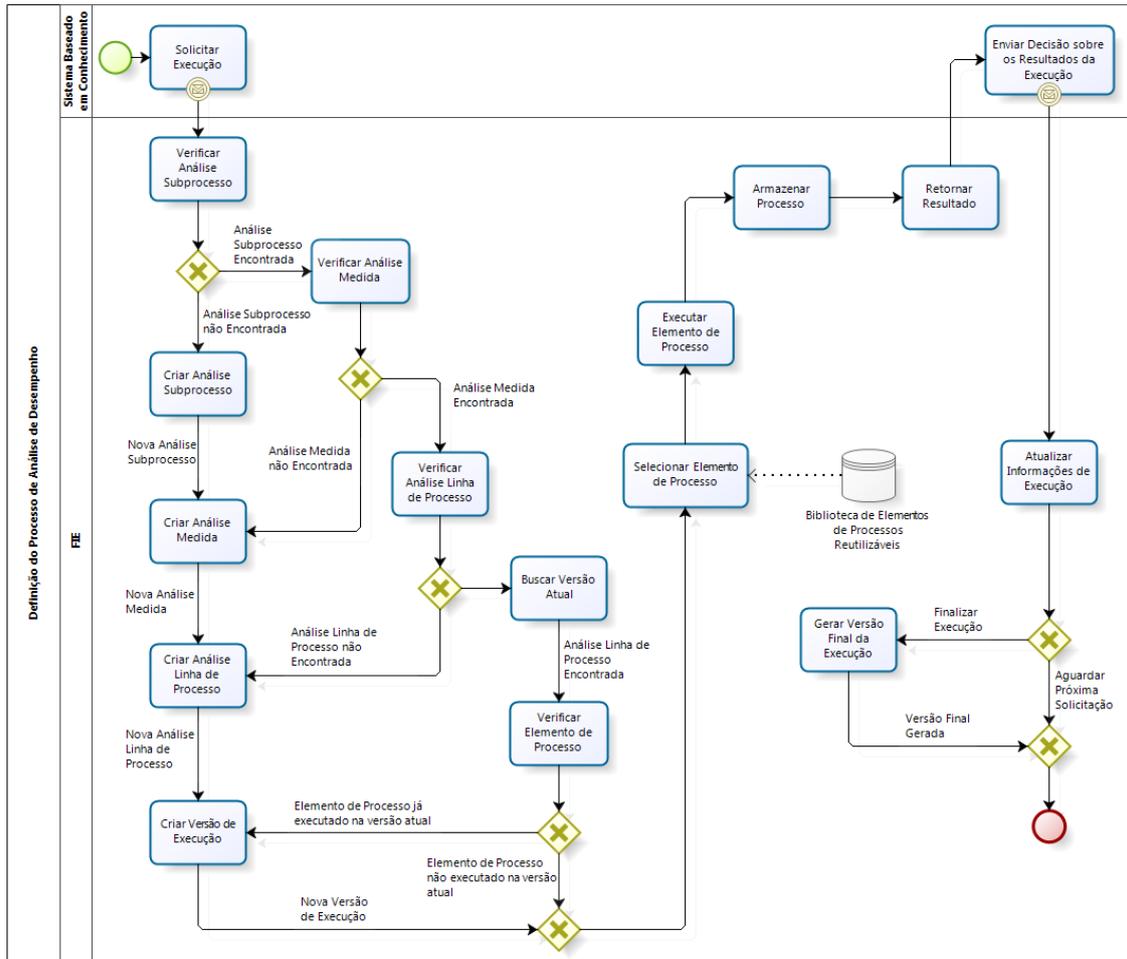


Figura 42 - Processo para Definição do Processo de Análise de Desempenho

O Processo para Definição do Processo de Análise de Desempenho possui 17 atividades, sendo 2 atividades externas à ferramenta FIE. As atividades do processo serão descritas a seguir:

| | |
|------------|--|
| Atividade: | Solicitar Execução |
| Descrição: | Solicitar a execução da análise de desempenho enviando as informações do subprocesso analisado. Esta é uma atividade executada pelo SBC. As informações que serão enviadas pelo SBC são: (i) <i>organização</i> que definiu o subprocesso de software que será analisado; (ii) <i>medida</i> que será analisada; (iii) <i>subprocesso</i> analisado; (iv) <i>linha de processo</i> a ser instanciada; (v) <i>elemento de processo</i> que será executado pela ferramenta FIE, e; (vi) <i>arquitetura interna</i> do |

| | |
|-----------------------|--|
| | <i>elemento de processo</i> caso o <i>elemento de processo</i> a ser executado seja um <i>componente de processo</i> . |
| Pré-atividade: | - |
| Critérios de entrada: | Necessidade de execução de uma atividade da análise de desempenho pelo SBC. |
| Critérios de saída: | Disponibilização das informações para a ferramenta FIE estruturar a execução da atividade da análise de desempenho solicitada. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | SBC e FIE |
| Pós-atividade: | Verificar Análise Subprocesso |

| | |
|-----------------------|--|
| Atividade: | Verificar Análise Subprocesso |
| Descrição: | Verificar se a informação do subprocesso analisado e da organização já se encontram na base de dados da ferramenta FIE, por já existir uma análise realizada. Existem dois caminhos possíveis após a execução desta atividade: (i) Quando o supprocesso e organização informados pelo SBC não possuírem nenhuma análise já realizada, uma nova <i>análise subprocesso</i> será criada, ou; (ii) Quando o subprocesso e a organização estiverem cadastrados, a próxima etapa será verificar a medida que será analisada. |
| Pré-atividade: | Solicitar Execução |
| Critérios de entrada: | Solicitação do SBC para execução da Análise de Desempenho com as informações sobre o subprocesso analisado. |
| Critérios de saída: | Ter sido verificada a existência ou não do cadastro do subprocesso e organização na base de dados da ferramenta FIE. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Verificar Análise Medida (caso a <i>análise subprocesso</i> esteja cadastrada na base da ferramenta FIE) ou Criar Análise Subprocesso (caso a <i>análise subprocesso</i> não esteja cadastrada na base da ferramenta FIE). |

| | |
|------------|---|
| Atividade: | Verificar Análise Medida |
| Descrição: | Verificar se existe alguma <i>análise medida</i> para a medida e subprocesso analisado. Existem dois caminhos possíveis após a execução desta atividade: (i) Quando não existir, uma nova <i>análise medida</i> será criada, ou; (ii) Quando a <i>análise medida</i> existir na base da ferramenta FIE, o próximo passo é validar o <i>elemento de processo</i> . |

| | |
|-----------------------|---|
| Pré-atividade: | Verificar Análise Subprocesso |
| CrITÉRIOS de entrada: | <i>Análise subprocesso</i> já se encontra cadastrada na base da ferramenta FIE. |
| CrITÉRIOS de saída: | Ter sido verificada a existência ou não do cadastro da <i>análise medida</i> na base ferramenta FIE. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Verificar Análise Linha de Processo (caso a <i>análise medida</i> esteja cadastrada na base ferramenta FIE) ou Criar Análise Medida (caso a <i>análise medida</i> não esteja cadastrada na base da ferramenta FIE). |

| | |
|-----------------------|---|
| Atividade: | Verificar Análise Linha Processo |
| Descrição: | Verificar se existe alguma <i>análise linha de processo</i> em aberto (<i>status análise</i> igual a “Em execução”) para a linha de processo informada como parâmetro pelo SBC. Existem dois caminhos possíveis após a execução desta atividade: (i) Quando não existir, uma nova <i>análise linha de processo</i> será instanciada, ou; (ii) Quando a <i>análise linha de processo</i> existir na base da ferramenta FIE, o próximo passo é buscar a <i>versão</i> atual para continuar a execução. |
| Pré-atividade: | Verificar Análise Medida |
| CrITÉRIOS de entrada: | <i>Análise medida</i> já se encontra cadastrada na base da ferramenta FIE. |
| CrITÉRIOS de saída: | Ter sido verificada a existência ou não do cadastro da <i>análise linha de processo</i> na base da ferramenta FIE. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Buscar Versão Atual (caso a <i>análise linha de processo</i> esteja cadastrada na base da ferramenta FIE) ou Criar Análise Linha de Processo (caso a <i>análise linha de processo</i> não esteja cadastrada na base da ferramenta FIE). |

| | |
|-----------------------|---|
| Atividade: | Buscar Versão Atual |
| Descrição: | Buscar a última <i>versão</i> da execução do processo de análise de desempenho para uma <i>análise linha de processo</i> associada. A última <i>versão</i> será utilizada para verificação do <i>elemento de processo</i> que será executado. |
| Pré-atividade: | Verificar Análise Linha Processo |
| CrITÉRIOS de entrada: | <i>Análise linha de processo</i> já se encontra cadastrada na base da ferramenta FIE. |
| CrITÉRIOS de saída: | Ter sido identificada a última <i>versão</i> executada para uma |

| | |
|-----------------------|-----------------------------------|
| | <i>análise linha de processo.</i> |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Verificar Elemento de Processo |

| | |
|-----------------------|---|
| Atividade: | Verificar Elemento de Processo |
| Descrição: | Verificar se o <i>elemento de processo</i> informado pelo SBC já foi executado na <i>versão</i> corrente da execução do processo. Existem dois caminhos possíveis após a execução desta atividade: (i) Caso o <i>elemento de processo</i> já tenha sido executado na <i>versão</i> atual, o passo seguinte é criar uma nova <i>versão</i> para incluir o novo <i>elemento de processo</i> , ou; (ii) caso o <i>elemento de processo</i> não tenha sido executado na <i>versão</i> atual, o próximo passo é selecionar as informações do <i>elemento de processo</i> na Base de Elementos de Processo Reutilizáveis. |
| Pré-atividade: | Buscar Versão Atual |
| CrITÉRIOS de entrada: | <i>Versão</i> atual de execução encontrada na base da ferramenta FIE. |
| CrITÉRIOS de saída: | Ter sido verificado se o <i>elemento de processo</i> foi executado na <i>versão</i> do processo em execução. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Criar Versão de Execução (caso o <i>elemento de processo</i> já tenha sido executado na <i>versão</i> atual) ou Selecionar Elemento de Processo (caso não tenha). |

| | |
|-----------------------|---|
| Atividade: | Criar Análise Subprocesso |
| Descrição: | Caso as informações da organização e subprocesso analisado não se encontrem na base de dados da ferramenta FIE, uma nova <i>análise subprocesso</i> é criada a partir das informações do fornecidas pelo SBC. |
| Pré-atividade: | Verificar Análise Subprocesso |
| CrITÉRIOS de entrada: | Não existir a <i>análise subprocesso</i> na base de dados da ferramenta FIE. |
| CrITÉRIOS de saída: | Ter sido criada uma nova <i>análise subprocesso</i> . |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Criar Análise Medida. |

| | |
|-----------------------|---|
| Atividade: | Criar Análise Medida |
| Descrição: | Caso a <i>análise medida</i> não se encontre na base de dados da ferramenta FIE, uma nova <i>análise medida</i> é criada para uma <i>análise subprocesso</i> existente, a partir das informações de subprocesso e medida fornecidas pelo SBC. |
| Pré-atividade: | Criar Análise Subprocesso (caso seja uma nova execução) ou Verificar Análise Medida (caso a medida informada pelo SBC ainda não tenha sido analisada). |
| Crítérios de entrada: | Não existir a <i>análise medida</i> na base de dados da ferramenta FIE. |
| Crítérios de saída: | Ter sido criada uma nova <i>análise medida</i> . |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Criar Análise Linha de Processo |

| | |
|-----------------------|---|
| Atividade: | Criar Análise Linha de Processo |
| Descrição: | Caso a <i>análise medida</i> não se encontre na base de dados da ferramenta FIE, uma nova <i>análise linha de processo</i> é criada para uma <i>análise medida</i> existente, a partir das informações de subprocesso, medida e linha de processo fornecidas pelo SBC. A nova <i>análise linha de processo</i> é criada com o status “Em Execução” e com data início preenchida com a data atual. |
| Pré-atividade: | Criar Análise Medida (caso seja uma nova execução) ou Verificar Análise Linha de Processo (caso a linha de processo fornecida pelo SBC ainda não tenha sido executada para uma <i>análise medida</i> e <i>análise subprocesso</i> específica). |
| Crítérios de entrada: | Não existir a <i>análise linha de processo</i> em aberto para a <i>análise medida</i> e <i>análise subprocesso</i> em execução. |
| Crítérios de saída: | Ter sido criada uma nova <i>análise linha de processo</i> . |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Criar Versão de Execução |

| | |
|----------------|--|
| Atividade: | Criar Versão de Execução |
| Descrição: | Criar uma nova <i>versão</i> da execução do processo de análise de desempenho para uma <i>análise linha de processo</i> específica. Uma <i>versão</i> de execução é criada quando uma nova <i>análise linha de processo</i> é instanciada ou quando um <i>elemento de processo</i> já tenha sido executado na <i>versão</i> atual do processo. |
| Pré-atividade: | Criar Análise Linha de Processo (caso seja uma nova |

| | |
|-----------------------|--|
| | execução) ou Verificar Elemento de Processo (caso o <i>elemento de processo</i> já tenha sido executado na <i>versão</i> atual). |
| Critérios de entrada: | Uma nova <i>análise linha de processo</i> foi criada ou o elemento de processo verificado tenha sido executado na <i>versão</i> atual em aberto. |
| Critérios de saída: | Ter sido criada uma nova <i>versão</i> de execução. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Selecionar Elemento de Processo |

| | |
|-----------------------|---|
| Atividade: | Selecionar Elemento de Processo |
| Descrição: | Selecionar da Biblioteca de Elementos de Processos Reutilizáveis o <i>elemento de processo</i> de acordo com a informação fornecidas pelo SBC. Neste momento, verifica-se se o <i>elemento de processo</i> é um componente e se possui arquitetura interna, ou seja, se possui outros <i>elementos de processo</i> associados. Caso existam, estes <i>elementos de processo</i> também serão disponibilizados na <i>versão</i> de execução em aberto para que o usuário da ferramenta seja capaz de executá-los. |
| Pré-atividade: | Verificar Elemento de Processo (caso o <i>elemento de processo</i> não tenha sido executado na <i>versão</i> de execução em aberto) ou Criar Versão de Execução (caso uma nova <i>versão</i> de execução tenha sido criada). |
| Critérios de entrada: | Uma nova <i>versão</i> de execução tenha sido criada ou o <i>elemento de processo</i> não tenha sido executado na <i>versão</i> de execução em aberto. |
| Critérios de saída: | Ter selecionado as informações do <i>elemento de processo</i> na Biblioteca de Elementos de Processos Reutilizáveis. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Executar Elemento de Processo |

| | |
|------------|--|
| Atividade: | Executar Elemento de Processo |
| Descrição: | Disponibilizar as informações do <i>elemento de processo</i> selecionado da Biblioteca de Elementos de Processos Reutilizáveis para auxiliar a execução do <i>elemento de processo</i> , tais como: Nome, descrição, critérios de entrada e saída, parâmetros de entrada e saída e o script necessário para a geração do <i>resultado de execução</i> , caso este componente possua um script associado. Esta atividade gera um <i>elemento de processo executável</i> na |

| | |
|-----------------------|--|
| | base da ferramenta FIE a partir do <i>elemento de processo</i> selecionado. |
| Pré-atividade: | Selecionar Elemento de Processo |
| Critérios de entrada: | Possuir as informações do <i>elemento de processo</i> selecionado na Biblioteca de Elementos de Processos Reutilizáveis. |
| Critérios de saída: | Ter executado o <i>elemento de processo</i> e gerado o <i>elemento de processo executável</i> na base da ferramenta FIE. |
| Responsáveis: | Usuário da Ferramenta FIE. |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Armazenar Processo |

| | |
|-----------------------|--|
| Atividade: | Armazenar Processo |
| Descrição: | Armazenar os <i>resultados de execução</i> do <i>elemento processo executável</i> e disponibilizá-los na versão em aberto da execução da linha de processo. Isto permite que o usuário do ambiente SPEAKER tenha uma visão gráfica sobre o processo executado. |
| Pré-atividade: | Executar Elemento de Processo |
| Critérios de entrada: | O <i>elemento de processo</i> tenha sido executado na ferramenta FIE. |
| Critérios de saída: | Ter armazenado os <i>resultados de execução</i> do <i>elemento de processo executável</i> e disponibilizá-los na <i>versão</i> em aberto da execução da linha de processo. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Retornar Resultado |

| | |
|-----------------------|---|
| Atividade: | Retornar Resultado |
| Descrição: | Prover os <i>resultados de execução</i> do <i>elemento de processo executável</i> para o SBC, para que este possa continuar a execução de seu processo. Todos os <i>resultados de execução</i> são disponibilizados no formato de texto e arquivos que são incluídos na ferramenta e associado aos parâmetros de saída definidos para o <i>elemento de processo executável</i> em questão. Arquivos que não estão associados a um parâmetro de saída definido também podem ser armazenados. |
| Pré-atividade: | Armazenar Processo |
| Critérios de entrada: | O <i>elemento de processo</i> ter sido executado e os <i>resultados de execução</i> tenham sido armazenados. O <i>elemento de processo executável</i> tenha sido gerado na base da ferramenta FIE. |
| Critérios de saída: | Terem enviadas as informações dos <i>resultados de execução</i> ao |

| | |
|-----------------------|--|
| | SBC. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE, SBC |
| Pós-atividade: | Enviar Decisão sobre os Resultados da Execução |

| | |
|-----------------------|--|
| Atividade: | Enviar Decisão sobre os Resultados da Execução |
| Descrição: | Esta é uma atividade de responsabilidade do SBC que informa a ferramenta FIE sobre a utilização dos <i>resultados de execução</i> do <i>elemento de processo executável</i> . O SBC informa a ferramenta FIE se os <i>resultados de execução</i> fornecidos foram utilizados na obtenção do resultado da análise de desempenho do subprocesso analisado. Esta informação é necessária para a ferramenta FIE gerar a versão final do processo executado. |
| Pré-atividade: | Retornar Resultado |
| CrITÉRIOS de entrada: | Terem sido enviadas as informações dos <i>resultados de execução</i> ao SBC. |
| CrITÉRIOS de saída: | Ter recebido as informações sobre a utilização dos <i>resultados de execução</i> do SBC. |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE, SBC |
| Pós-atividade: | Atualizar Informações de Execução |

| | |
|-----------------------|--|
| Atividade: | Atualizar Informações de Execução |
| Descrição: | Atualizar a informação sobre a utilização do <i>resultado de processo o elemento de processo executável</i> enviado ao SBC. Esta informação é utilizada para a geração da <i>versão</i> final da execução do processo de Análise de Desempenho. O SBC fornece esta informação para cada <i>versão</i> do <i>elemento de processo executável</i> através de uma interface definida pela ferramenta FIE. |
| Pré-atividade: | Enviar Decisão sobre os Resultados da Execução (a ferramenta FIE aguarda a execução desta atividade do SBC para continuar a execução do processo). |
| CrITÉRIOS de entrada: | O <i>resultado de execução</i> do <i>elemento de processo executável</i> tenha sido enviado para o SBC e armazenado na <i>versão</i> em aberto da execução do processo. |
| CrITÉRIOS de saída: | Terem sido atualizadas as informações sobre a utilização do <i>resultado de execução</i> do <i>elemento de processo executável</i> . |
| Responsáveis: | - |
| Participantes: | - |
| Artefatos requeridos: | - |

| | |
|-----------------------|--|
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | Gerar Versão Final da Execução (caso seja a última solicitação de execução para a <i>análise linha de processo</i> em execução). |

| | |
|-----------------------|---|
| Atividade: | Gerar Versão Final da Execução |
| Descrição: | <p>Gerar a <i>versão</i> final do processo de análise de desempenho executado. Para realizar esta atividade, utiliza-se todas as <i>versões</i> executadas pela análise de desempenho.</p> <p>A <i>versão</i> final executada é composta pelos <i>elementos de processo executável</i> que foram realmente utilizados para a definição do resultado da análise de desempenho, isso porque, um mesmo <i>elemento de processo</i> pode ter sido executado diversas vezes.</p> <p>O SBC é responsável por informar a ferramenta FIE qual a <i>versão</i> do <i>elemento de processo executável</i> foi utilizada para se obter o resultado da análise.</p> <p>Para a geração da <i>versão</i> final executada, são percorridas todas as <i>versões</i> criadas para aquela análise de desempenho na ordem de execução, e para cada <i>versão</i> é verificado o indicador de utilização de cada <i>elemento de processo executável</i> para definir se fará parte ou não da <i>versão</i> final do processo. Após essa verificação uma nova <i>versão</i>, denominada <i>versão</i> final da execução é criada contendo o Processo de Análise de Desempenho executado.</p> <p>No momento da geração da <i>versão</i> final da execução do processo, caso um mesmo <i>elemento de processo</i> tenha sido executado mais de uma vez, e por algum erro na comunicação com o SBC o atributo que indica sua utilização estiverem marcados SIM, como padrão a ferramenta FIE utilizará o último resultado executado (Regra de Validação VAL04).</p> |
| Pré-atividade: | Atualizar Informações de Execução |
| CrITÉRIOS de entrada: | Ter sido finalizada a execução do processo de análise de desempenho. |
| CrITÉRIOS de saída: | Ter sido gerada a <i>versão</i> final da execução do processo somente com os <i>elementos de processo executável</i> que foram utilizados na geração do resultado final da análise de desempenho realizada pelo SBC. |
| Responsáveis: | Usuário da ferramenta FIE |
| Participantes: | - |
| Artefatos requeridos: | - |
| Artefatos produzidos: | - |
| Ferramentas: | FIE |
| Pós-atividade: | - |

4.5 Considerações Finais

Neste capítulo foram apresentados os requisitos definidos para a construção da ferramenta FIE, além de suas funcionalidades e do processo definido para a definição dinâmica da Análise de Desempenho. Também foram apresentados os diagramas de classe e de sequência das entidades e dos métodos utilizados na interface com o SBC.

Além disso, também foram apresentados os conceitos criados e adotados na modelagem da ferramenta, as regras de validação e a interface de comunicação com o SBC. O próximo capítulo apresenta um exemplo de uso para avaliar a utilização da ferramenta FIE.

CAPÍTULO 5 - EXEMPLO DE USO

Este capítulo apresenta um exemplo de uso realizado para avaliar a utilização da ferramenta FIE juntamente com as informações dos elementos de processo cadastradas na ferramenta A2MComponenteProcesso.

5.1 Introdução

Os cenários de testes definidos em GONÇALVES (2014) foram adaptados para demonstrar a utilização da ferramenta FIE. Os cenários foram criados para avaliar os scripts definidos para a execução de componentes concretos de maior nível. Os cenários foram criados a partir de exemplos da literatura (ROCHA *et al.*, 2012) que foram adaptados de (FLORAC e CARLETON, 1999) com dados hipotéticos. Até o momento não foi possível a utilização de dados reais para avaliar esses cenários. É importante ressaltar que apesar dos exemplos serem fictícios, eles se assemelham com situações reais que podem ocorrer em projetos de software.

Os cenários definidos demonstram a execução dos elementos de processo respeitando as informações definidas para os elementos de processo e cadastrando os resultados para cada execução. Como até o momento o desenvolvimento do SBC não foi concluído, as interações com a ferramenta FIE serão simuladas da maneira com a qual estão sendo desenvolvidas. A execução dos scripts definidos por GONÇALVES (2014) será realizada pela ferramenta Minitab (Minitab Inc., 2013).

5.2 Execução do Processo

Esta seção apresenta os cenários adaptados para a avaliação das funcionalidades descritas nesta dissertação para a ferramenta FIE.

5.2.1 Cenário 1

A Figura 43 apresenta a linha de processos definida por GONÇALVES (2014) para verificar a estabilidade do subprocesso. Esta linha será utilizada no cenário descrito a seguir.

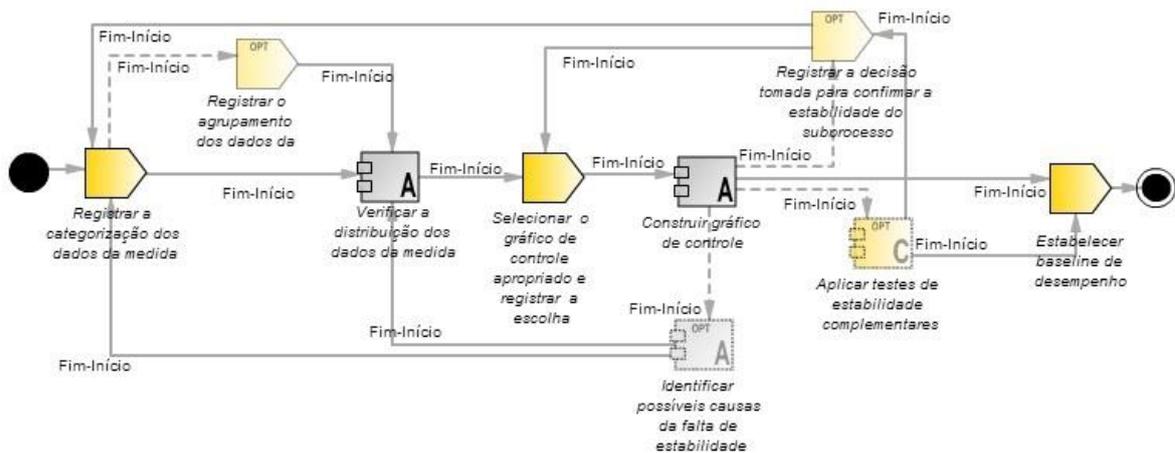


Figura 43 - Linha de processos para verificar a estabilidade do subprocesso (GONÇALVES, 2014)

Este exemplo se baseia na necessidade do gerente de escritório de projetos da Empresa XYZ analisar a estabilidade do subprocesso de manutenção de projetos com relação a medida quantidade despendidas em atividades do.

Para instanciar a execução do processo na ferramenta FIE, o SBC necessita enviar uma mensagem informando os seguintes parâmetros: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Manutenção de Projetos; (iii) Medida analisada: Quantidade de horas gastas; (iv) Linha de Processo Executada: Verificar Estabilidade; (v) Elemento de Processo: Registrar a categorização dos dados da medida,e; (vi) Arquitetura Interna: Não possui.

A partir dessas informações a ferramenta FIE cria a estrutura necessária para armazenar a análise que será realizada. É importante ressaltar que o elemento de processo da linha de processo executado será “Registrar a categorização dos dados da medida”. A Figura 44, demonstra a estrutura de *análise subprocesso*, *análise medida* e *análise linha de processo* criada para armazenar a execução desta análise. São três telas apresentadas na figura que fazem parte da funcionalidade “Acompanhar Execução do Processo”. A primeira tela exibida é a tela de **Análises Realizadas**, onde o usuário da ferramenta FIE poderá escolher qual subprocesso e organização deseja visualizar. Após a escolha, a tela de **Medidas Analisadas** é exibida listando as medidas que foram analisadas para aquele subprocesso específico. A próxima tela exibe as **Linhas de Processo** em execução para a medida e subprocesso selecionados.

Para este caso o usuário selecionou na tela de **Análises Realizadas** o subprocesso “Manutenção de Projetos”. Na tela de **Medidas Analisadas**, selecionou “Quantidade de horas gastas com manutenção”

The figure consists of three vertically stacked screenshots of the A2M software interface, showing the workflow for analyzing subprocesses, measures, and process lines.

Top Screenshot: Análises Realizadas
 The interface shows a menu on the left with 'Análises Realizadas' selected. The main area displays a table of subprocesses. A callout box points to the table with the text: "Lista dos subprocessos analisados."
 Table data:

| Organização | Subprocesso |
|----------------------------------|------------------------|
| XYZ Consultoria de Sistemas Ltda | Manutenção de Projetos |

Middle Screenshot: Medidas Analisadas
 The interface shows the 'Medidas Analisadas' screen. A callout box points to the table with the text: "Lista das medidas analisadas para o subprocesso selecionado."
 Table data:

| Medida |
|---|
| Quantidade de horas gastas com manutenção |

Bottom Screenshot: Linhas de Processo
 The interface shows the 'Linhas de Processo' screen. A callout box points to the table with the text: "Lista das linhas de processo instanciadas em execução."
 Table data:

| Linha de Processo | Data Inicio | Data Fim | Status |
|------------------------|-------------|----------|-------------|
| Verificar estabilidade | 28/10/2014 | | Em execução |

Figura 44 - Tela da ferramenta FIE - Acompanhar Subprocessos, Medidas e Linhas de Processos Analisadas

A primeira *versão* da execução foi criada e o elemento de processo executado foi associado a ela. Por padrão, as versões são criadas com a seguinte nomenclatura: EXEC_VERSAO_XXX, onde XXX será o número da versão, que neste caso é 001. A Figura 45 apresenta a *versão* criada para a análise que estamos executando neste cenário.

A Figura 46 apresenta a tela que exibe os elementos de processo que foram executados para a versão selecionada, que neste caso foi o componente concreto “Registrar a categorização dos dados da medida”.



Figura 45 - Tela da ferramenta FIE - Acompanhar Versões Executadas

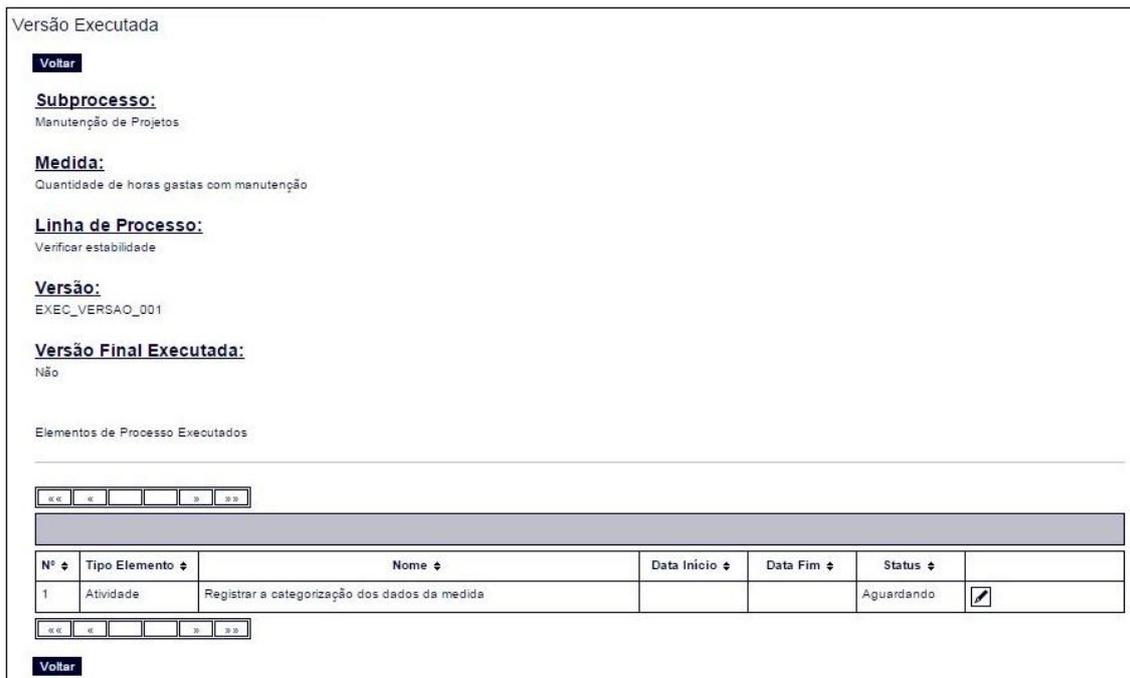


Figura 46 - Tela da ferramenta FIE - Acompanhar Elementos de processo Executados

A partir deste momento, o usuário da ferramenta deverá preencher as informações sobre a execução do elemento, para isso, basta clicar no link⁵ **Editar Elemento**, localizado ao direito de cada elemento de processo executado. A Figura 47 e a Figura 48 demonstram as telas onde o usuário deverá preencher as informações de execução do elemento, tais como: Data de Início e Fim da Execução do Elemento, Esforço em horas gastos, Status e os Resultados encontrados. Para este elemento, os dados da medida analisada deverão estar dispostos na Planilha do ambiente SPEAKER (fornecida pelo ambiente através do SBC).

Como resultado esperado, o componente “Registrar a categorização dos dados da medida” exige um artefato contendo o registro da categorização realizada “Registro da categorização dados da medida”. A Figura 48 demonstra o formato que este resultado é registrado na ferramenta FIE. Os valores coletados que serão utilizados neste exemplo são apresentados na Tabela 17 e são referentes a 20 dias de observação da execução do subprocesso.

É importante ressaltar que a qualquer momento o usuário da ferramenta pode consultar os dados da execução do elemento de processo utilizando a funcionalidade *consultar análise realizada*, selecionando a *análise subprocesso*, *análise medida*, *análise linha de processo* e a *versão* que deseja consultar. A Figura 49 apresenta a tela com todos os elementos executados e os resultados cadastrados.

Para o tipo de dados (dados variáveis) e o tipo de análise desejada pelo gerente de projetos, o gráfico de controle XmR seria o mais adequado, visto que a medida em questão representa uma variável medida frequentemente. O componente definido para o gráfico de controle XmR recomenda que os dados sigam uma distribuição Normal. Para avaliar esta recomendação, o componente abstrato “Verificar a distribuição dos dados da medida” será executado, porém, um componente concreto deverá ser escolhido para que a execução possa ser realizada. A partir dessas informações o SBC envia os seguintes parâmetros para informar à ferramenta FIE sobre a execução do elemento de processo: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Manutenção de Projetos; (iii) Medida analisada: Quantidade de horas gastas; (iv) Linha de Processo Executada: Verificar Estabilidade; (v) Elemento de Processo: Verificar a distribuição Normal dos dados com Minitab, e; (vi) Arquitetura Interna: Não possui.

⁵ Será utilizada notação com os termos em letras minúsculas e em negrito para se referir aos links da FIE.

Editar Elemento de Processo
Executado

Subprocesso:
Manutenção de Projetos

Medida:
Quantidade de horas gastas com manutenção

Linha de Processo:
Verificar estabilidade

Versão:
EXEC_VERSAO_001

Versão Final Executada:
Não

Confirmar **Cancelar**

Principal **Script**

Nº: 1 Tipo: Atividade
Elemento: Registrar a categorização dos dados da medida Descrição: Registrar a categorização dos dados coletados para a medida do subprocesso que foi selecionado para a análise de desempenho. Alguns dos critérios que podem ser utilizados para categorizar os dados da medida são: Tamanho do projeto, Versão do subprocesso, Perfil da equipe, Complexidade, Domínio de aplicação.

Critério de Entrada: Ter-se analisado o conjunto dos dados da medida do subprocesso selecionado para a análise de desempenho. Critério de Saída: Ter-se registrado a categorização dos dados da medida.

Executor: rodrigo Esforço:
 Data Início: Data Fim:
 Status: Aguardando

Parâmetros de Entrada

| Nome | Descrição |
|--|--|
| Identificação do subprocesso selecionado | Identificação do subprocesso selecionado |
| Planilha de medidas do SPEAKER | Planilha de medidas do SPEAKER |

Parâmetros de Saída

| Nome | Descrição |
|---|---|
| Registro da categorização dos dados da medida | Registro da categorização dos dados da medida |

Resultados da Execução

| Nome Arquivo | Tipo do arquivo | Artefato |
|--------------|-----------------|----------|
|--------------|-----------------|----------|

Confirmar **Cancelar**

Figura 47 - Tela da ferramenta FIE - Acompanhar Elemento de Processo

Cadastrar Resultados da Execução

Subprocesso:
Manutenção de Projetos

Medida:
Quantidade de horas gastas com manutenção

Linha de Processo:
Verificar estabilidade

Linha de Processo: Verificar estabilidade

Versão:
EXEC_VERSAO_001

Versão Final Executada:
Não

Atividade:
Registrar a categorização dos dados da medida

Confirmar **Cancelar**

Principal

Nome: Medidas - Manutenção de Projetos - SPEAKER.xls Tamanho(bytes): 421888

Arquivo: Artefato: Registro da categor

Tipo do arquivo: Planilha

Resultado: Planilha do ambiente contendo as medidas coletadas e categorizadas que serão utilizadas para esta execução do processo de análise de desempenho.

Arquivo: 



Confirmar **Cancelar**

Figura 48- Tela da ferramenta FIE - Cadastrar Resultado da Execução

Tabela 17 - Valores coletados

| Dia | Valores coletados |
|-----|-------------------|
| 1 | 50,5 |
| 2 | 43,5 |
| 3 | 45,5 |
| 4 | 39,8 |
| 5 | 42,9 |
| 6 | 44,3 |
| 7 | 44,9 |
| 8 | 42,9 |
| 9 | 39,8 |
| 10 | 39,3 |
| 11 | 48,8 |
| 12 | 51,0 |
| 13 | 44,3 |
| 14 | 43,0 |
| 15 | 51,3 |
| 16 | 46,3 |
| 17 | 45,2 |
| 18 | 48,1 |
| 19 | 45,7 |
| 20 | 44,1 |

Consultar Versão Executada

Voltar

Subprocesso:
Manutenção de Projetos

Medida:
Quantidade de horas gastas com manutenção

Linha de Processo:
Verificar estabilidade

Versão:
EXEC_VERSAO_001

Versão Final Executada:
Não

Diagrama

```

graph LR
    Start(( )) --> Task[Registrar a categorização dos dados da medida]
    Task --> End(( ))
  
```

Resultado da Execução:

Registrar a categorização dos dados da medida

Descrição: Registrar a categorização dos dados coletados para a medida do subprocesso que foi selecionado para a análise de desempenho. Alguns dos critérios que podem ser utilizados para categorizar os dados da medida são: Tamanho do projeto, Versão do subprocesso, Perfil da equipe, Complexidade, Domínio de aplicação.

Parâmetros de Entrada: Identificação do subprocesso selecionado; Planilha de medidas do SPEAKER;

Parâmetros de Saída: Registro da categorização dos dados da medida;

Data Início: 28/10/2014
Data Fim: 28/10/2014
Esforço: 1.0
Executor: rodrigo
Status: Finalizado

Resultados da Execução

Nome Arquivo: Medidas - Manutenção de Projetos - SPEAKER.xls
Tamanho(bytes): 421888
Tipo do arquivo: Planilha
Artefato: Registro da categorização dos dados da medida
Resultado: Planilha do ambiente contendo as medidas coletadas e categorizadas que serão utilizadas para esta execução do processo de análise de desempenho.

Voltar

Figura 49 - Tela da ferramenta FIE - Consultar Versão Executada

Após o recebimento dos dados, o novo elemento de processo executável foi criado com sucesso e está disponível para preenchimento na ferramenta FIE. O procedimento é o mesmo descrito no item anterior. O elemento de processo foi incluído na versão corrente da execução.

Este elemento possui um script associado para geração dos resultados na ferramenta Minitab. A Figura 50, apresenta como a ferramenta FIE disponibiliza o

script para a geração dos resultados. O script está disponível para download na página de edição do elemento em questão. Após a execução do script e análise do usuário, os resultados serão cadastrados na ferramenta FIE. Como resultado, o elemento de processo exige um artefato contendo o resultado da verificação.

Editar Elemento de Processo
Executado

Subprocesso:
Manutenção de Projetos

Medida:
Quantidade de horas gastas com manutenção

Linha de Processo:
Verificar estabilidade

Versão:
EXEC_VERSAO_001

Versão Final Executada:
Não

Confirmar **Cancelar**

Principal Script

Nome Script: DistribuicaoNormal.mtb Tamanho(bytes): 240
Script Original:

Valores dos Parâmetros:

| Parâmetro ↓ | Valor ↓ |
|-------------|---------|
|-------------|---------|

Confirmar **Cancelar**

Figura 50 - Tela da ferramenta FIE - Script para geração dos resultados

Após o download e a execução do script na ferramenta Minitab, os resultados foram cadastrados na ferramenta FIE pelo usuário juntamente com a análise realizada de acordo com a descrição do componente. A Figura 51 exibe o resultado cadastrado pelo usuário na ferramenta FIE. Todos os resultados armazenados na ferramenta FIE (gráficos, planilhas, imagens etc.) durante a execução dos elementos de processo estão disponíveis em GONÇALVES (2014). A Figura 52 apresenta o resultado da execução do script que foi armazenado pelo usuário na ferramenta FIE.

Verificar a distribuição Normal dos dados com Minitab

Identificador: COP.ADP.EST.CON.0002

Descrição:
 Verificar se os dados seguem o padrão de distribuição normal, aplicando o teste de normalidade de Anderson-Darling. O teste de normalidade possui duas hipóteses: i) hipótese nula (H_0) de que os dados seguem uma distribuição Normal e ii) hipótese alternativa (H_1) de que os dados não seguem uma distribuição Normal. A tabela abaixo apresenta os valores dos níveis de significância estatística do teste.

| Confiabilidade do Teste | Nível de significância estatística (α) |
|-------------------------|---|
| 90% | 0,1 |
| 95% | 0,05 |
| 97,5% | 0,025 |
| 99% | 0,01 |

Para a verificação da distribuição Normal os dados devem ser organizados na Planilha de medidas do SPEAKER no formato de coluna, o nível α de significância do teste deve ser escolhido e os gráficos gerados. O nível de significância mais usado é $\alpha = 0,05$.

Para a interpretação do teste deve-se considerar que os dados seguem uma distribuição normal se: i) os pontos no gráfico formam uma linha reta e ii) o valor-P identificado no gráfico não for menor que o nível α escolhido. Se o valor-P identificado no gráfico for menor que o nível α escolhido, a hipótese nula (H_0) é rejeitada e se conclui que os dados não seguem uma distribuição normal.

Parâmetros de Entrada: Planilha de medidas do SPEAKER;
Parâmetros de Saída: Resultado da verificação da distribuição normal dos dados;
Data Início: 28/10/2014
Data Fim: 28/10/2014
Esforço: 2,0
Executor: rodrigo
Status: Finalizado

Resultados da Execução

 **Nome Arquivo:** Gráfico Distribuição Normal e Histograma.jpg
Tamanho(bytes): 71498
Tipo do arquivo: Imagem
Artefato: Resultado da verificação da distribuição normal dos dados
Resultado: De acordo com a análise do teste sugerida pelo componente, os valores coletados seguem uma distribuição normal pois os valores estão distribuídos em torno da reta e o valor de-P 0,339 não é menor que nível escolhido que é 0,05.

Figura 51 – Resultado da Verificação da distribuição dos dados

Após a análise realizada pelo SBC sobre os resultados cadastrados na ferramenta FIE, o SBC envia uma nova mensagem solicitando a execução do próximo elemento da linha de processo que é “Selecionar o gráfico de controle apropriado e registrar a escolha”, para isso, os seguintes parâmetros são informados: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Manutenção de Projetos; (iii) Medida analisada: Quantidade de horas gastas; (iv) Linha de Processo Executada: Verificar Estabilidade; (v) Elemento de Processo: Selecionar o gráfico de controle apropriado e registrar a escolha, e; (vi) Arquitetura Interna: Não possui.

O próximo elemento a ser executado é o “Construir gráfico de controle”. Por ser um componente abstrato, o SBC deverá informar qual o componente concreto deverá ser executado, esta escolha baseia-se na execução da atividade anterior da linha de processo “Selecionar o gráfico de controle apropriado e registrar a escolha, que neste caso será o componente “Construir gráfico de controle XmR com Minitab”. Neste momento o SBC envia os seguintes parâmetros para a ferramenta FIE: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Manutenção de Projetos; (iii) Medida analisada: Quantidade de horas gastas; (iv) Linha

de Processo Executada: Verificar Estabilidade; (v) Elemento de Processo: Construir gráfico de controle XmR com o Minitab, e; (vi) Arquitetura Interna: Gerar Gráfico XmR, Aplicar teste de estabilidade padrões e Aplicar testes de estabilidade.

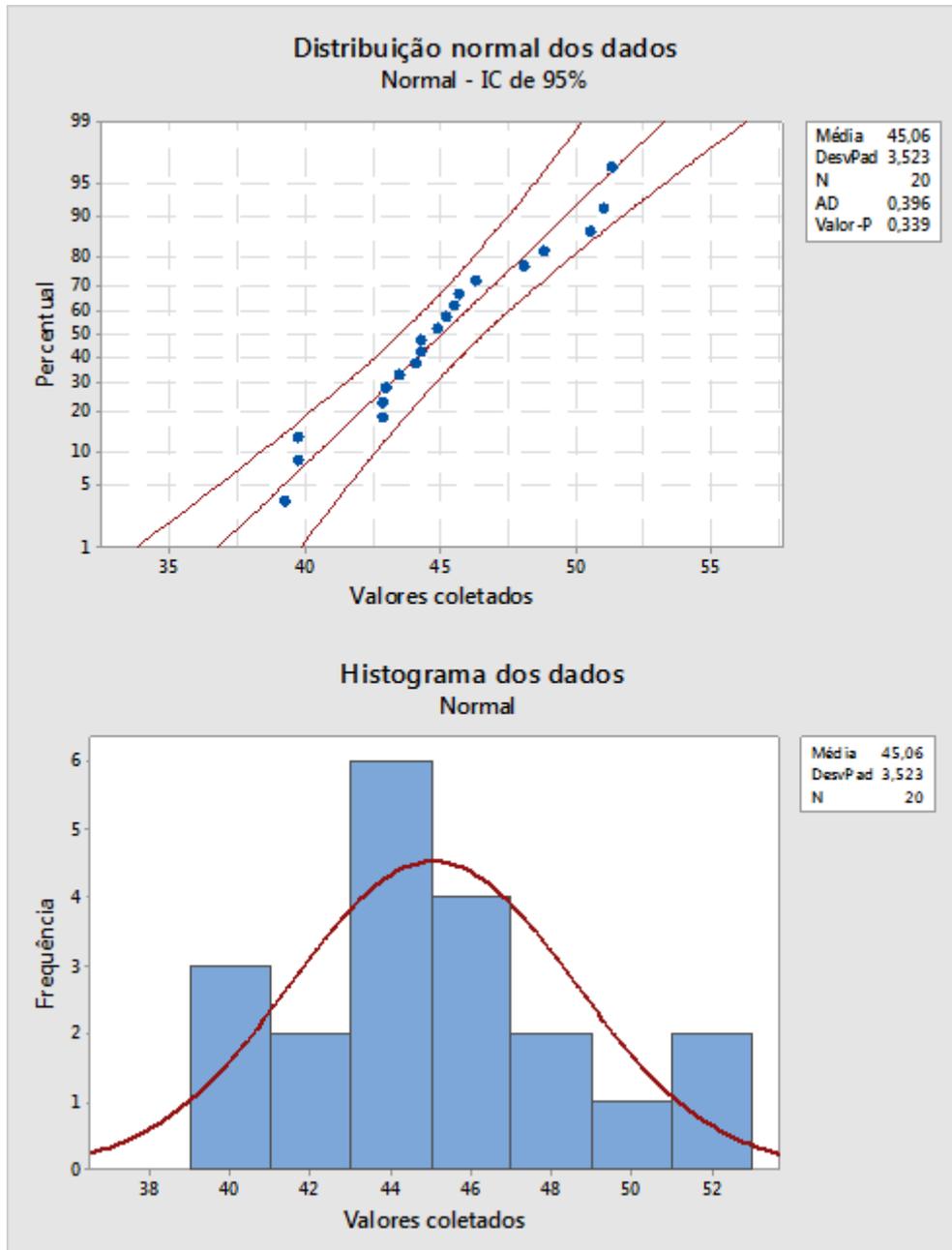


Figura 52 - Gráfico de Distribuição Normal e Histograma

Após a inclusão do componente na versão em execução, a ferramenta FIE disponibiliza o script associado para a geração do gráfico de controle e dos testes de estabilidade. Através da tela de edição de elemento de processo o usuário cadastra o

resultado da execução e registra sua análise de acordo com o solicitado pela descrição do componente. Os resultados são analisados pelo SBC e de acordo com os resultados apresentados, os valores não falharam em nenhum dos testes executados. A Figura 53 apresenta o gráfico de controle gerado para ferramenta Minitab a partir do script disponibilizado pela ferramenta FIE. Sendo assim, o processo é considerado estável o que vai ao encontro do resultado do exemplo da literatura que foi adaptado, onde o processo se apresenta estável com comportamento repetível.

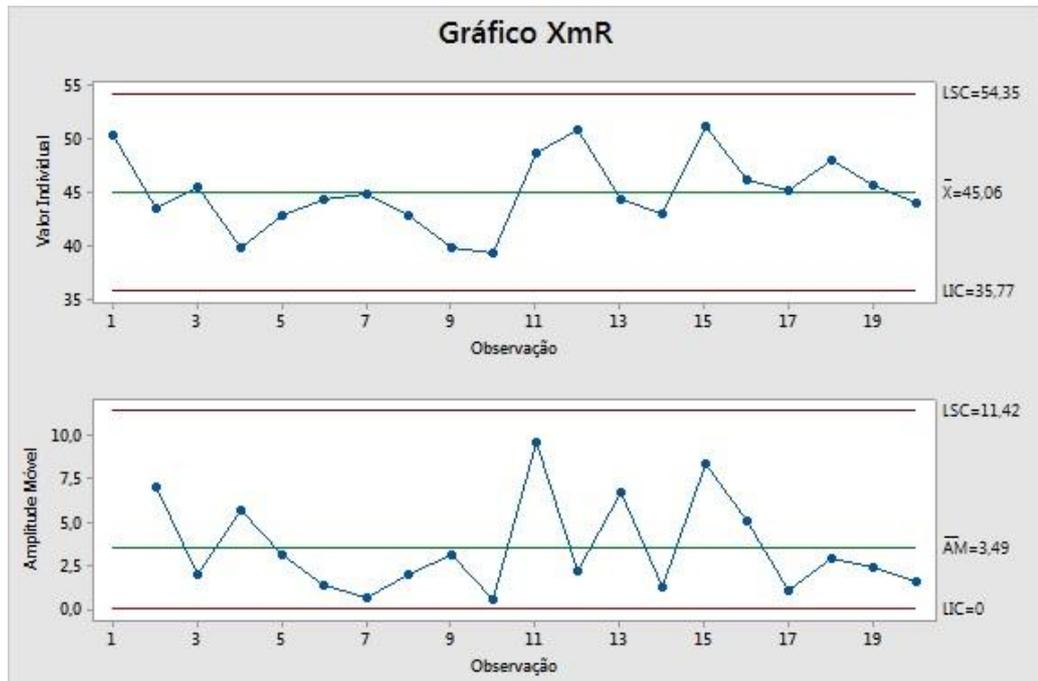


Figura 53 - Gráfico XmR gerado na execução do componente “Construir gráfico de controle”

Por fim, como o processo foi considerado estável e o usuário ficou satisfeito com o resultado encontrado, o próximo elemento de processo executado é o “Estabelecer baseline de desempenho”. Neste momento o SBC envia os seguintes parâmetros para a ferramenta FIE: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Manutenção de Projetos; (iii) Medida analisada: Quantidade de horas gastas; (iv) Linha de Processo Executada: Verificar Estabilidade; (v) Elemento de Processo: Estabelecer baseline de desempenho, e; (vi) Arquitetura Interna: Não possui.

Para finalizar a execução de uma análise específica, o usuário deverá clicar na funcionalidade *acompanhar execução do processo*, selecionando a *análise subprocesso*, a *análise medida*, a *análise linha de processo* e então clicar na funcionalidade *encerrar análise e gerar versão final*, neste momento a ferramenta FIE

percorre todas as versões executadas para a análise específica e gera uma nova versão contendo o processo executado para se obter o resultado da Análise de Desempenho. Para este cenário, somente uma versão foi criada, por isso a versão final será a mesma. A Figura 54 apresenta os elementos de processo que foram executados nesta versão. O usuário poderá visualizar este fluxo a qualquer momento através da funcionalidade *consultar análises realizadas*, ou a funcionalidade *consultar versão final executada*, que exibe apenas a versão final que foi executada para uma análise específica.



Figura 54 - Versão final instanciada e executada do processo.

A Figura 54 apresenta a versão final instanciada e executada da linha de processo para verificar a estabilidade do subprocesso analisado. Se analisarmos a Figura 43, que apresenta toda a linha de processo definida com suas possíveis variações, é possível verificar as escolhas dos componentes concretos executados realizadas pelo SBC.

A próxima seção apresenta um cenário onde será avaliada a linha de processo para determinar a capacidade do subprocesso.

5.2.2 Cenário 2

A Figura 55 apresenta a linha de processos definida para determinar a capacidade do subprocesso. Esta linha será utilizada no cenário descrito a seguir. Todos os artefatos utilizados como resultado da execução estão detalhados em GONÇALVES (2014).

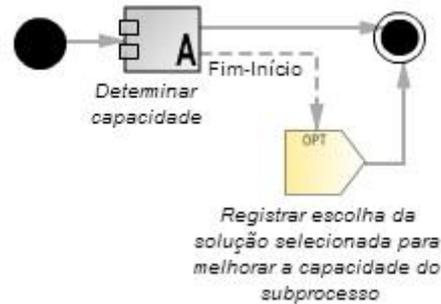


Figura 55 - Linha de processos para determinar a capacidade do subprocesso (GONÇALVES, 2014)

Para realização da análise da capacidade, utilizou-se o subprocesso avaliado como estável no cenário 1, onde o gerente de projetos deseja analisar a quantidade de horas diárias despendidas em atividades de manutenção. De acordo com a linha de processo, o primeiro elemento de processo a ser executado é o componente abstrato “Determinar capacidade”, por isso, o primeiro passo é a escolha do componente concreto que será executado, que neste caso é o “Determinar capacidade de análise XmR com Minitab”. Para instanciar a execução desta linha de processo, o SBC envia os seguintes parâmetros para execução da ferramenta FIE: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Manutenção de Projetos; (iii) Medida analisada: Quantidade de horas gastas; (iv) Linha de Processo Executada: Determinar Capacidade; (v) Elemento de Processo: Determinar capacidade de análise XmR com Minitab, e; (vi) Arquitetura Interna: Construir histograma de frequência, Calcular índice de capacidade (C_p).

Assim como na linha de processo, a estrutura para armazenar esta análise também será criada na ferramenta FIE, desta vez, apenas um registro de *análise linha de processo* será criado, pois a organização, subprocesso e medidas já foram analisados anteriormente, ou seja, já existe um objeto *analise subprocesso* e *analise medida* criados na base da ferramenta. A Figura 56 apresenta a tela que exibe as 2 linhas de processo que já foram analisados para o mesmo subprocesso e medida.

Editar Elemento de Processo
Executado

Subprocesso:
Manutenção de Projetos

Medida:
Quantidade de horas gastas com manutenção

Linha de Processo:
Verificar Capacidade

Versão:
EXEC_VERSAO_001

Versão Final Executada:
Não

Confirmar **Cancelar**

Principal Script

Nº: 1 Tipo: Componente
Elemento: Nome: Determinar capacidade de análise de XmR com Minitab Descrição: Determinar a capacidade do subprocesso, construindo o histograma de frequência, comparando os limites naturais do processo (voz do processo) com os limites de especificação (voz do cliente) e calculando o índice de capacidade (C_p) com o software Statistica.
Este componente para determinar a capacidade do subprocesso aplica-se a subprocessos que tiveram a estabilidade analisada pelo gráfico XmR.

Critério de Entrada: Ter-se a baseline de desempenho do subprocesso estabelecida e os limites de especificação (voz do cliente) determinados. Critério de Saída: Ter-se a capacidade do subprocesso determinada.

Executor: rodrigo Esforço:
Data Início: Data Fim:
Status: Aguardando

Parâmetros de Entrada

| Nome | Descrição |
|--|--|
| Baseline de desempenho do subprocesso. | Baseline de desempenho do subprocesso. |
| Planilha de medidas do SPEAKER | Planilha de medidas do SPEAKER |

Parâmetros de Saída

| Nome | Descrição |
|--|--|
| Histograma de frequência com o índice de capacidade (C_p). | Histograma de frequência com o índice de capacidade (C_p). |

Elementos de Processo Executados

| Nº | Tipo Elemento | Nome | Data Início | Data Fim | Status | |
|----|---------------|---|-------------|----------|------------|--|
| 1 | Componente | Construir histograma de frequência | | | Aguardando | |
| 2 | Componente | Calcular índice de capacidade (C_p) | | | Aguardando | |

Confirmar **Cancelar**

Figura 57 - Tela da ferramenta FIE - Edição de Elemento de Processo Pai

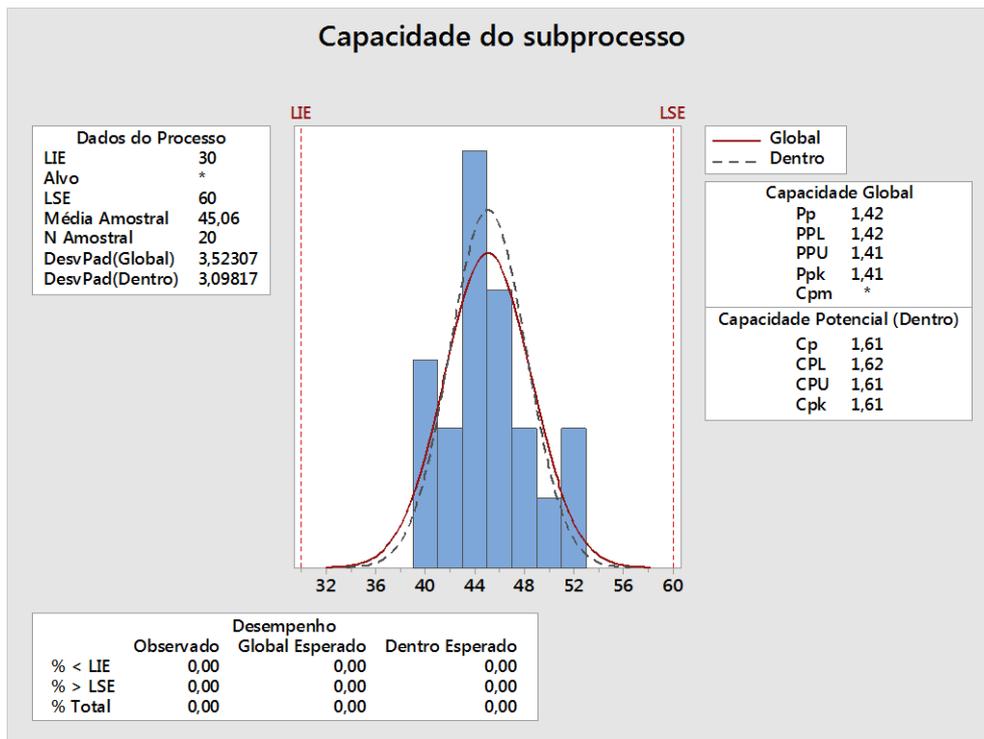


Figura 58 – Resultado da capacidade do processo



Figura 59 - Processo Instanciado e Executado Cenário 2

A Figura 59 apresenta a versão final instanciada e executada da linha de processo para verificar a capacidade do subprocesso analisado. Se analisarmos a Figura 55, que apresenta toda a linha de processo definida com suas possíveis variações, é possível verificar as escolhas dos componentes concretos executados realizadas pelo SBC.

A próxima seção apresenta um cenário onde será avaliada a linha de processo para estabelecer o modelo de desempenho do subprocesso.

5.2.3 Cenário 3

A Figura 60 apresenta a linha de processos definida para estabelecer o modelo de desempenho do subprocesso. Esta linha será utilizada no cenário descrito a seguir. Todos os artefatos utilizados como resultado da execução estão detalhados em GONÇALVES (2014).



Figura 60 - Linha de processo para estabelecer o modelo de desempenho do subprocesso (GONÇALVES, 2014)

Para a realização deste cenário, utilizou-se a mesma Organização dos outros 2 cenários. Porém, desta vez, o subprocesso analisado é o de Desenvolvimento de Requisitos e a medida analisada é a produtividade de especificação de caso de uso, que é composta pelas medidas tamanho e esforço. A finalidade da organização é identificar a relação existente entre o tamanho de casos de uso e o esforço necessário para a especificação. O subprocesso em questão já foi analisado anteriormente e definido como estável, e os valores foram utilizados para estabelecer a *baseline* de desempenho do subprocesso.

A instanciação do processo tem início com o elemento de processo “Registrar escolha das variáveis do modelo de desempenho”, onde as variáveis que farão parte do modelo de desempenho são selecionadas. Neste momento, o SBC enviar os seguintes parâmetros para a ferramenta FIE: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Desenvolvimento de Requisitos; (iii) Medida analisada: Produtividade de especificação de caso de uso; (iv) Linha de Processo Executada: Estabelecer modelo de desempenho; (v) Elemento de Processo: Registrar escolha das variáveis do modelo de desempenho, e; (vi) Arquitetura Interna: Não possui. A partir dessas informações, a ferramenta FIE cria a estrutura necessária para armazenar o resultado da análise. Neste caso, um novo subprocesso foi incluído, conforme apresenta a Figura 61. Sendo assim, a estrutura de *análise*

subprocesso, análise medida, análise linha de processo e versão também foram definidas, conforme apresentados nos cenários anteriores.

| Análises Realizadas | | |
|----------------------------------|-------------------------------|--|
| Subprocessos | | |
| Organização ↕ | Subprocesso ↕ | |
| XYZ Consultoria de Sistemas Ltda | Desenvolvimento de Requisitos | |
| XYZ Consultoria de Sistemas Ltda | Manutenção de Projetos | |

Figura 61 - Tela da ferramenta FIE - Consultar Subprocessos Analisados

Após a execução desta atividade, e análise do resultado pelo usuário do SBC, o próximo elemento a ser executado é “Selecionar método para construção do modelo de desempenho”. Sendo assim, os seguintes parâmetros são enviados pelo SBC: (i) Organização responsável pela definição do subprocesso: Empresa XYZ; (ii) Subprocesso analisado: Desenvolvimento de Requisitos; (iii) Medida analisada: Produtividade de especificação de caso de uso; (iv) Linha de Processo Executada: Estabelecer modelo de desempenho; (v) Elemento de Processo: Selecionar método para construção do modelo de desempenho, e; (vi) Arquitetura Interna: Não possui. Para esta atividade, o usuário deverá selecionar o método com base no tipo de variáveis que foram escolhidas para compor o modelo de desempenho e cadastrar o resultado na ferramenta. A tela para o cadastro do resultado de uma atividade e é idêntica a de um componente que não possui arquitetura interna, conforme apresentado nos outros cenários. A Figura 62 apresenta o resultado cadastrado para esta atividade.

O próximo elemento de processo a ser executado de acordo com a linha é o componente abstrato “Desenvolver modelo de desempenho”, que por sua vez necessita que a escolha de uma variante seja realizada, neste caso, o componente concreto escolhido foi o “Desenvolver modelo de desempenho através da análise de regressão – Minitab”. Por este motivo os seguintes parâmetros são enviados à ferramenta FIE para execução: (i) Empresa XYZ; (ii) Subprocesso analisado: Desenvolvimento de Requisitos; (iii) Medida analisada: Produtividade de especificação de caso de uso; (iv) Linha de Processo Executada: Estabelecer modelo de desempenho; (v) Elemento de Processo: Desenvolver modelo de desempenho através de análise de regressão - Minitab, e; (vi) Arquitetura Interna: Não possui. Após a inclusão deste elemento na versão em execução, e geração do gráfico com o script disponibilizado pela ferramenta

FIE, o usuário deverá cadastrar os resultados, que neste caso será um gráfico gerado pela ferramenta Minitab contendo a relação entre as variáveis e a função matemática (Esforço = 3,573 + 2,321 Tamanho) que representa o modelo de desempenho que quantifica a relação entre as variáveis.

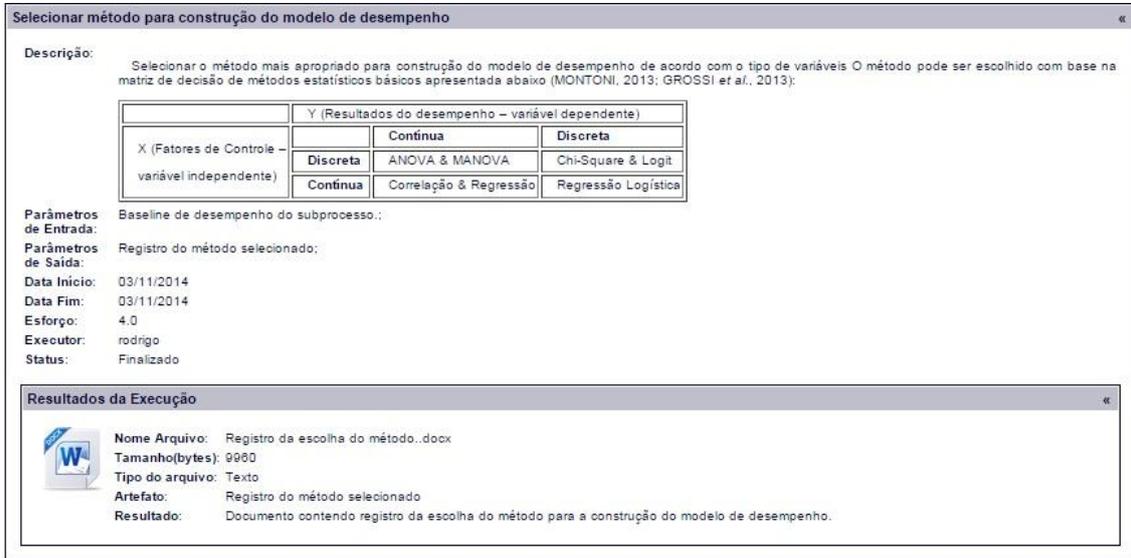


Figura 62 - Resultado da execução da atividade "Selecionar método para construção do modelo de desempenho".

O último elemento de processo a ser executado pela linha é o “Calibrar e testar o modelo de desempenho” cujo objetivo é verificar se o modelo de desempenho está atendendo às necessidades atuais da organização. Para executar este componente, o SBC envia os seguintes parâmetros para a ferramenta FIE: (i) Empresa XYZ; (ii) Subprocesso analisado: Desenvolvimento de Requisitos; (iii) Medida analisada: Produtividade de especificação de caso de uso; (iv) Linha de Processo Executada: Estabelecer modelo de desempenho; (v) Elemento de Processo: Calibrar e testar o modelo de desempenho, e; (vi) Arquitetura Interna: Não possui. Após a inclusão na versão, o usuário deverá cadastrar os resultados e preencher as informações de execução do elemento.

A Figura 63 apresenta o resultado da execução do script associado ao componente “Desenvolver modelo de desempenho através de análise de regressão – Minitab”.

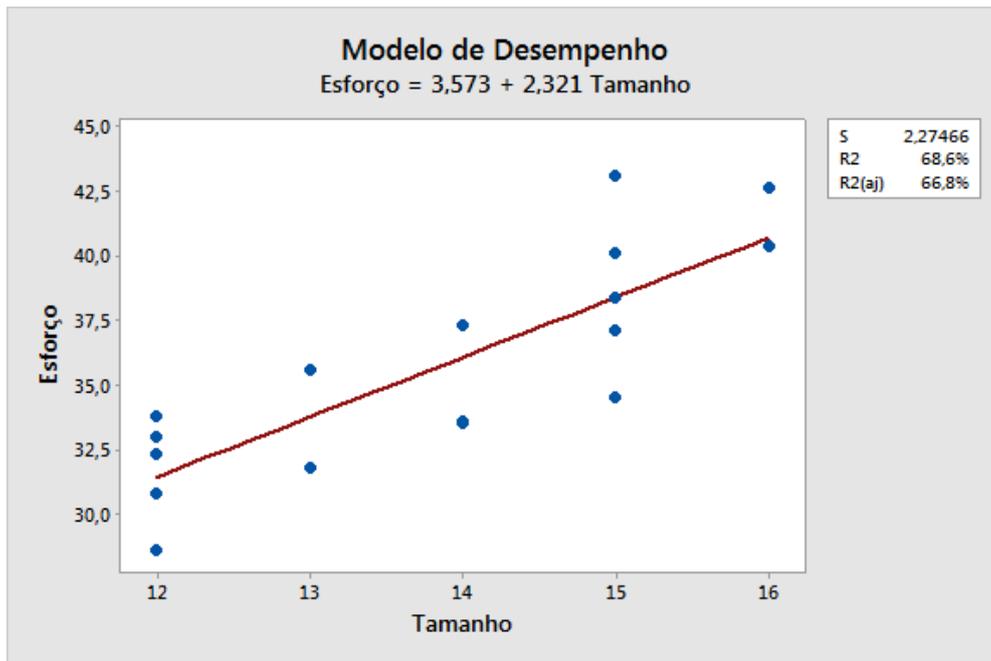


Figura 63 - Modelo de Desempenho

A Figura 64 apresenta a versão final instanciada e executada da linha de processo para estabelecer o modelo de desempenho do subprocesso analisado. Se analisarmos a Figura 60, que apresenta toda a linha de processo definida com suas possíveis variações, é possível verificar as escolhas dos componentes concretos executados realizadas pelo SBC.

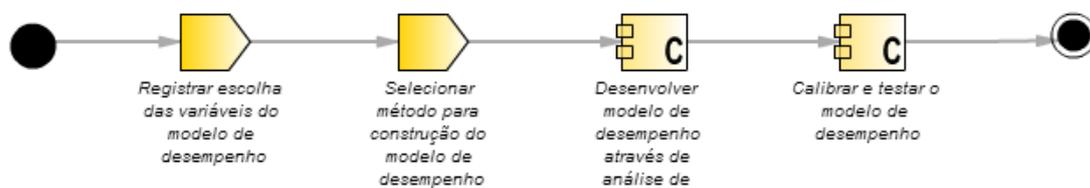


Figura 64 - Processo final instanciada e executado

A Figura 64 apresenta a versão final instanciada e executada da linha de processo para estabelecer o modelo de desempenho do subprocesso analisado. Se analisarmos a Figura 60, que apresenta toda a linha de processo definida com suas possíveis variações, é possível verificar as escolhas dos componentes concretos executados realizadas pelo SBC.

5.3 Considerações Finais

Neste capítulo foi apresentado um exemplo de uso baseado na literatura (GONÇALVES, 2014) para demonstrar a utilização da ferramenta FIE na execução das linhas de processo “Verificar estabilidade”, “Verificar capacidade” e “Estabelecer modelo de desempenho do subprocesso”. O exemplo apresentado, mesmo não sendo um cenário real, possui situações que ocorrem em projetos de software.

Através da execução deste exemplo é possível verificar que a ferramenta FIE atende aos requisitos gerais do ambiente SPEAKER REQ02 e REQ03 pois controle a execução das atividades do processo de análise de desempenho.

É possível determinar também o atendimento aos requisitos definidos para a ferramenta:

- O REQ01 foi atendido através das simulações das chamadas que serão realizadas pelo SBC de solicitação de execução.
- O REQ02 foi atendido através dos resultados armazenados para a execução de cada elemento de processo.
- O REQ03 foi atendido através da disponibilização dos resultados da execução de um elemento de processo através do método de consulta aos resultados disponibilizado pela ferramenta FIE.
- O REQ04 foi atendido através das versões controladas para cada análise de desempenho realizada.
- O REQ05 foi atendido, visto que os componentes executados foram componentes concretos. Alguns dos componentes possuíam variações e somente os componentes concretos foram executados. Durante a execução dos exemplos foram simuladas solicitações de execução para componentes concretos e a ferramenta FIE retornou uma mensagem de validação para este requisito.
- O REQ06 foi atendido exibindo somente os dois primeiros níveis de execução para um componente que possui arquitetura interna.
- O REQ07 foi atendido, visto que as versões finais geradas para cada um dos exemplos executados encontra-se registrada neste capítulo.

- O REQ08 foi atendido através da funcionalidade que gera a versão final gerada, e durante a execução dos exemplos pode se verificar seu funcionamento.

Os resultados obtidos através da execução dos elementos de processo são armazenados e analisados para o conhecimento obtido através dessas execuções sejam incluídos na ferramenta SBC para auxiliar os usuários do ambiente SPEAKER em execuções futuras.

É importante ressaltar, que a ferramenta FIE apesar de ser desenvolvida para executar as linhas de processo definidas por GONÇALVES (2014), pode ser utilizada para executar qualquer outra linha de processo definida na ferramenta A2MComponenteProcesso do ambiente A2M. Isto significa que, mesmo que a linha de processo seja alterada, a ferramenta FIE não necessitará de alteração.

CAPÍTULO 6 - CONCLUSÃO

Este capítulo apresenta as considerações finais desta dissertação, suas contribuições, suas limitações e os trabalhos futuros.

6.1 Considerações Finais

A execução do processo de Análise de Desempenho não é uma atividade trivial, pois necessita do conhecimento na utilização e interpretação dos diferentes tipos de gráficos e técnicas utilizadas.

A execução de algumas atividades deste processo na maioria das vezes necessita ser realizada mais de uma vez, isto porque, a técnica escolhida pode não ser a mais indicada para o cenário que está sendo avaliado. Estas execuções devem ser controladas e armazenadas para que o conhecimento obtido através de sua execução possa ser utilizado em outras análises.

O ambiente SPEAKER encontra-se em desenvolvimento e tem como objetivo auxiliar a execução do processo de Análise de Desempenho para Processos de Software através da captura, armazenamento e divulgação do conhecimento necessário para sua realização (Capítulo 3). Este ambiente contém: (i) um sistema baseado em conhecimento (SBC); (ii) linhas de processo que apoiam a execução das atividades de análise de desempenho, e; (iii) uma ferramenta que instancia e executa as linhas de processo através da comunicação com o SBC (FIE).

Dentro do contexto do ambiente SPEAKER, esta dissertação teve como objetivo a criação de um apoio ferramental para executar as atividades do processo de Análise de Desempenho definidas por GONÇALVES (2014) baseado na necessidade de informações do SBC.

Para embasamento desta proposta, foi realizada uma revisão ad hoc da literatura técnica (Capítulo 2) com foco em análise de desempenho e execução de processos. A partir desta revisão, os requisitos e funcionalidades da ferramenta FIE foram definidos.

Um exemplo de uso foi realizado para avaliar o uso da ferramenta. Este exemplo foi adaptado de GONÇALVES (2014), visto que não foi possível avaliar o seu funcionamento em um cenário real.

6.2 Contribuições

A principal contribuição deste trabalho é a criação da Ferramenta de Instanciação e Execução do Processo de Análise de Desempenho. A partir desta ferramenta, é possível instanciar e executar as atividades do processo de Análise de Desempenho definidas no ambiente SPEAKER e armazenar os seus resultados.

Ao longo do desenvolvimento desta dissertação alguns resultados foram registrados nas publicações listadas abaixo:

- SCHOTS, N. C. L., GONÇALVES, T. G., MAGALHAES, R. F., ROCHA, A. R. C., SANTOS, G., OLIVEIRA, K. M., 2013, “Componentes e Requisitos de um Ambiente Baseado em Conhecimento para Análise de Desempenho de Processos de Software”. In: IX Workshop Anual do MPS - WAMPS 2013, v. 1, pp. 84-94, Campinas, Outubro.
- SCHOTS, N. C. L.; GONCALVES, T. G.; MAGALHAES, R. F.; ROCHA, A. R. C.; SANTOS, G.; OLIVEIRA, K. M. Supporting Software Process Performance Analysis through a Knowledge-based Environment. In: XL Conferencia Latinoamericana en Informática, 2014, Montevideo. Proceedings of the Latin American Computing Conference (CLEI), 2014, pp. 286-297.

6.3 Limitações

A principal limitação deste trabalho foi a impossibilidade da utilização da ferramenta FIE em um cenário real. A ferramenta FIE necessita do conhecimento em análise de desempenho de processos e das solicitações realizadas pelo SBC para que então possa instanciar e executar as atividades do processo de análise de desempenho. O SBC ainda se encontra em desenvolvimento.

Outra limitação no desenvolvimento da ferramenta FIE foi a não execução automática dos scripts associados aos componentes de processo. A execução dos scripts depende da ação do usuário da ferramenta FIE. Os softwares estatísticos utilizados para a geração dos gráficos (Minitab e Statistica) não possuem integração com o framework utilizado no desenvolvimento da ferramenta FIE.

6.4 Trabalhos Futuros

Como trabalho futuro, uma oportunidade seria a inclusão da geração automática dos gráficos de controle e a aplicação dos testes de estabilidade dentro da própria ferramenta. Para que isso seja possível, será necessário melhorar a integração da ferramenta com a base de métricas coletadas pela organização, pois atualmente, a base de medidas é baseada em um planilha controlada pelo SBC. A ferramenta FIE atualmente armazena somente as medidas utilizadas na geração do gráfico de controle.

Outra possibilidade, seria a integração com outro software estatístico para realização das atividades que geram os gráficos de controle. A ferramenta Statistica (StatSoft Inc., 2013) possui uma interface que permite esta integração, porém não está disponível para o framework utilizado no desenvolvimento da ferramenta FIE.

Como a ferramenta FIE foi desenvolvida antes da criação do SBC, outra oportunidade de trabalho futuro seria a realização de uma avaliação do ambiente SPEAKER como um todo em um cenário real e a partir desta avaliação propor melhorias para a ferramenta.

REFERÊNCIAS

- ALEIXO *et al.*, 2011, “A Model-Driven Approach to Managing and Customizing Software Process Variabilities”, in *Enterprise Information Systems*, pp., 372-387.
- BALDASSARRE, M. T., BOFFOLI, N., CAIVANO, D., 2010, “Statistical Process Control for Software: Fill de Gap”, in COSKUN, A., "Quality Management and Six Sigma", pp., 135-153.
- BALDASSARRE T., BOFFOLI N., CAIVANO D., VISAGGIO G., 2004, “Managing Software Process Improvement (SPI) through Statistical Process Control (SPC)”, In: *Proceedings of 5th International Conference on Product Focused Software Process Improvement (PROFES'04)*, pp. 30-46.
- BARCELLOS, M. P., 2009, “Uma Estratégia para Medição de Software e Avaliação de Bases de Medidas para Controle Estatístico de Processos de Software em Organizações de Alta Maturidade”. Tese de Doutorado, Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- BARRETO, A., 2011, “Uma Abordagem para Definição de Processos Baseada em Reutilização Visando à Alta Maturidade em Processos”, Tese de Doutorado, Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- BORIA, J. L., 2007, “What’s Wrong With My Level 4?”, *Comunicação Pessoal*.
- CARD, D., 2007, “Challenges in Applying SPC to Software”, in: Ebert, C. e Dumke, R., *Software Measurement*, Springer, pp. 413-418.
- CARD, D., DOMZALSKI, K., DAVIES, G., 2008, "Making Statistics Part of Decision Making in an Engineering Organization", *IEEE Software*, v. 25, n. 3, pp. 37-47.

- CARDOSO, F. S., 2012, “Definição de Processos Reutilizáveis para Projetos com Aquisição”, Dissertação de Mestrado, Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- CERDEIRAL, C., *et al.*, 2007 “Uma Abordagem para Controle Estatístico do Processo e Gerência Quantitativa de Projetos”. *In*: VI Simpósio Brasileiro de Qualidade de Software, pp. 293-307, Porto de Galinhas - PE.
- COSTA, T. M., 2012, “Melhoria Contínua de Processo de Software Utilizando a Teoria das Restrições”, Dissertação de Mestrado Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- CMMI Product Team, 2010, “CMMI for Development, Version 1.3 (CMU/SEI-2010-TR-033)”. Software Engineering Institute, Carnegie Mellon University. Disponível em: <http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm>. Acesso em: janeiro de 2014.
- FLORAC, W. A., CARLETON, A. D., BARNARD, J. R., 2000, "Statistical Process Control: Analyzing a Space Shuttle Onboard Software Process", IEEE Software, v. 17(4), pp. 97- 106.
- FLORAC, W. A., CARLETON, A. D., 1999, “Measuring the Software Process: Statistical Process Control for Software Process Improvement”, Addison Wesley.
- FLORENCE, A., 2001, “Lessons Learned in Attempting to Achieve Software CMM Level 4, CrossTalk, v. 14 (8), pp. 29–30.
- FONSECA, P. C., “Modelo para Controle Estatístico de Processos de Desenvolvimento de Software (CEP-S)”, Dissertação de Mestrado em Ciência da Computação, Belo Horizonte, Minas Gerais, Brasil.

- FUGGETTA, A., 2000, “Software Process: A Roadmap”. In: Proceedings of the Conference on the Future of Software Engineering (ICSE '00), New York, USA, pp. 25-34.
- GARCIA, C. S., 2012, “Proposta de uma linha de processo de software para desenvolvimento de aplicações usando SOA e BPM”, Dissertação de Mestrado em Informática, Pontifícia Universidade Católica do Paraná – PUC-PR, Curitiba.
- GONÇALVES, T. G., 2014, “Componentes de Processo para Análise de Desempenho de Processos de Software”, Dissertação de Mestrado, Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- GONÇALVES, T. G.; ROCHA, A. R.; OLIVEIRA, K. M., 2013, “Definição de Componentes de Processo para Apoiar a Análise de Desempenho de Processos de Software”, In: XI Workshop de Teses e Dissertações em Qualidade de Software (WTDQS), Salvador.
- GONÇALVES, L. P., 2012, “Apoio ao Controle Estatístico de Processos de Software integrado a um ADS”, Dissertação de Mestrado em Ciência da Computação, Universidade Federal do Pará – UFPA, Pará.
- ISO/IEC, 2008a, “ISO/IEC 15504: Information Technology – Process Assessment – Part 7: Assessment of Organizational Maturity”, The International Organization for the Standardization and the International Electrotechnical Commission, Genebra, Suíça.
- ISO/IEC, 2008b, “ISO/IEC 12207: Systems and software engineering — Software life cycle processes”, The International Organization for the Standardization and the International Electrotechnical Commission, Genebra, Suíça.
- ISO/IEC, 2003, “ISO/IEC 15504: Software Engineering – Process Assessment – Part 2: Performing an Assessment”, International Organization for the Standardization and International Electrotechnical Commission, Genebra, Suíça.

- MAHANTI, R., EVANS, J. R., 2012, “Critical Success Factors for Implementing Statistical Process Control in the Software Industry”, *Benchmarking*, v. 19(4), pp. 374-394.
- MINITAB INC, 2013, “Minitab 17”, disponível em: <http://www.minitab.com/>. Acesso em Janeiro de 2014.
- MURTA, *et al.*, 2002, “Charon: uma Ferramenta para a Modelagem, Simulação, Execução e Acompanhamento de Processos de Software”, XVI Simpósio Brasileiro de Engenharia de Software, Gramado, pp. 366-371.
- NUNES, E. D., 2011, “Definição de Processos de Aquisição de Software para Reutilização”, Dissertação de Mestrado Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- PAULK, M. C., HYDER, E. B., 2007, “Common Pitfalls in Statistical Thinking”, *SQP*, v. 9 (3), pp. 12-19.
- REIS, C. A. L., 2003, “Uma Abordagem Flexível para Execução de Processos de Software Evolutivos”, Tese de Doutorado, Programa de Pós-Graduação em Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre.
- ROCHA, A. R. C., SOUZA, G. S., BARCELLOS, M. P. , 2012, “Medição de Software e Controle Estatístico de Processos”, PBQP Software, Brasília.
- SCHOTS, N. C. L., ROCHA, A. R. C., SANTOS, G., 2014a, “A Body of Knowledge for Executing Performance Analysis of Software Processes”. In: The 26th International Conference on Software Engineering and Knowledge Engineering, Vancouver, Canada.
- SCHOTS, N. C. L. ; GONCALVES, T. G. ; MAGALHAES, R. F. ; ROCHA, A. R. C. ; SANTOS, G. ; OLIVEIRA, K. M. . Supporting Software Process Performance Analysis

- through a Knowledge-based Environment. In: XL Conferencia Latinoamericana en Informática, 2014b, Montevideo. Proceedings of the Latin American Computing Conference (CLEI), 2014, pp. 286-297.
- SCHOTS, N. C. L., 2013a, “Um Ambiente Baseado em Conhecimento para Análise de Desempenho de Processos de Software”, Exame de Qualificação, Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- SCHOTS, N. C. L., *et al.*, 2013b, “Componentes e Requisitos de um Ambiente Baseado em Conhecimento para Análise de Desempenho de Processos de Software”. In: IX Workshop Anual do MPS, 2013, Campinas, pp. 84-94.
- SCHOTS, N. C. L., ROCHA, A. R., 2012, “Um Workflow para Controle Estatístico de Processos em Software”, In: VIII Workshop Anual do MPS, Campinas.
- SCHOTS, N. C. L., 2010, “Uma abordagem para a identificação de causas de problemas utilizando Grounded Theory”, Dissertação de Mestrado Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.
- SIMÕES, C.A., 2011, Repositório de Medidas para Organização de Alta Maturidade em Processos de Software, Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro.
- SIMÕES, C., *et al.*, 2013, “Aplicando Controle Estatístico de Processo em Projetos Evolutivos de Pequeno Tamanho: Resultados e Lições Aprendidas na Implementação do Nível 5 do CMMI-DEV na Synapsis”, In: XII Simpósio Brasileiro de Qualidade de Software (SBQS), pp. 286-293, 2013.
- SOARES BARRETO, A.O., 2011, Definição e Gerência de Objetivos de Software Alinhados ao Planejamento Estratégico, Tese de Doutorado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, Brasil.

SOFTEX, 2013, “MPS.BR – Melhoria de Processo do Software Brasileiro – Guia de Implementação – Parte 6: Fundamentação para Implementação do Nível B do MRMPS”. Disponível em: <http://www.softex.br/mpsbr>. Acesso em: janeiro de 2014.

SOFTEX, 2012, “MPS.BR – Melhoria de Processo do Software Brasileiro – Guia Geral”. Disponível em: <http://www.softex.br/mpsbr>. Acesso em: janeiro de 2014.

SOFTEX, 2012a, “MPS.BR – Melhoria de Processo do Software Brasileiro – Guia de Implementação – Parte 11: Implementação e Avaliação do MR-MPS-SW: 2012 em Conjunto com o CMMI-DEV v1.3”. Disponível em: <http://www.softex.br/mpsbr>. Acesso em: janeiro de 2014.

STATSOFT, 2013, “Statistica 12”, disponível em: <http://www.statsoft.com>. Acesso em Janeiro de 2014.

TARHAN, A., DEMIRÖRS, O., 2006, “Investigating Suitability of Software Process and Metrics for Statistical Process Control”, Software Process Improvement, Lecture Notes in Computer Science, vol. 4257, pp. 88-99.

TEIXEIRA, E. N., 2011, “Odysseyprocess-Fex: Uma Abordagem para Modelagem de Variabilidades de Linha de Processos de Software”, Dissertação de Mestrado do Programa de Pós-graduação em Engenharia de Sistemas e Computação – PESC/COPPE, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro.

WAZLAWICK, R. S., 2009, “Metodologia de Pesquisa para Ciência da Computação”, 1ª edição, Elsevier, Rio de Janeiro.

WELLER, E. F., 2000, "Practical Applications of Statistical Process Control", IEEE Software, v. 17, n. 3, pp. 48-55.

WHEELER, D. J., Chambers, D. S., 1992, “Understanding Statistical Process Control”, 2nd Edition, SPC Press, Inc.

WHEELER, D. J., 2008, “Entendendo a Variação: A Chave para Administrar o Caos”,
QualityMark Ed., Rio de Janeiro.