Silvia Inês Dallavalle de Pádua

**MÉTODO DE AVALIAÇÃO DO
MODELO DE PROCESSOS DE
NEGÓCIO DO EKD**

Tese apresentada a Escola de
Engenharia de São Carlos da
Universidade de São Paulo, como
parte dos requisitos para obtenção
do Título de Doutor em
Engenharia Mecânica.

Orientador: Prof. Dr. Ricardo Yassushi Inamasu

São Carlos

2004

Dedicatória

*Aos meus pais, que sempre me incentivaram,
acreditaram nos meus sonhos e me deram
forças para continuar minha luta.
Sou eternamente agradecida.*

*Ao Jubert, pelo carinho, amor,
companheirismo e por incentivar-me sempre.
Agradeço também por me estimular todos os
dias com energias positivas e com sua forma
simples e objetiva de ver a VIDA.
Muito Obrigada!*

*A minha querida irmã Sandra, que embora não
esteja mais nesse plano, com certeza
acompanhou minha luta e está compartilhando
com alegria essa vitória.*

*“DE TUDO, FICARAM TRÊS COISAS:
a certeza de que estamos sempre começando...
a certeza de que é preciso continuar...
a certeza de que seremos interrompidos antes de
terminar...*

*PORTANTO DEVEMOS
fazer da interrupção um caminho novo...
da queda um passo de dança...
do medo, uma escada...
do sonho, uma ponte...
da procura... um encontro”*

Fernando Sabino

Agradecimentos

A Deus, pela saúde, disposição e vontade de trabalhar que tenho todos os dias.

À CAPES (Coordenação de Aperfeiçoamento de Pessoal de Ensino Superior), pelo apoio financeiro, fundamental para o desenvolvimento e conclusão deste trabalho.

Ao professor Dr. Ricardo Yassuchi Inamasu, pela valiosa orientação neste trabalho, meus sinceros agradecimentos.

Ao professor Dr. Edson Walimir Cazarini, meu orientador de mestrado, pela confiança e apoio de sempre.

Ao professor Dr. Arthur José Vieira Porto, reservo minha gratidão pela atenção e apoio nos vários momentos de dúvidas e dificuldades.

A professora Dr. Rosely Sanches (ICMC), uma grande professora e amiga, reservo minha gratidão pelas conversas e orientações nos momentos de dúvidas e dificuldades.

A professora Dra. Ethel Cristina Chiari da Silva, pela importante contribuição na banca de qualificação e pelo incentivo e apoio que recebo desde a graduação.

A professora Dra. Creusa Sayuri, pelas orientações que me foram passadas na banca de qualificação.

Ao professor Adenilso da Silva Simão (ICMC), pela importante contribuição no desenvolvimento deste trabalho.

À amiga Andrea Ribari Yoshizawa, por me ajudar em vários momentos de dúvidas.

À professora Rosana, da Biblioteca Central da Escola de Engenharia de São Carlos EESC-USP, pela orientação na parte de metodologia científica deste trabalho.

Ao professor Dr. Henrique Rozenfeld, pelas várias conversas e orientações que contribuíram muito para minha formação.

Ao professor Dr. João do Espírito Santo Batista (ICMC), pelo incentivo e amizade.

Aos professores Villa, Rosely Sanches (ICMC) Agostinho, Edmundo, Frederico Bremer, Cazarini, Renato Vairo e Renata Fortes (ICMC), pelos conhecimentos que me

foram passados, pela dedicação que fora dada, pelas orientações que me ajudaram a chegar até aqui e que seguirão comigo tanto na minha vida pessoal quanto profissional.

Aos funcionários do Departamento de Engenharia Mecânica da Escola de Engenharia de São Carlos da USP.

Aos amigos Alessandra, Mayb, Fabiana, Fabrícia, Andrea, Beth, Bispo, Íris, Rosana, João e Fernando pela convivência, amizade e troca de experiências que com certeza, ajudaram-me a evoluir um pouco mais como ser humano e também como profissional.

A bibliotecária Eleninha, da Biblioteca Central da Escola de Engenharia de São Carlos EESC-USP, pela grande ajuda na correção das referências bibliográficas.

Ao Tales pelo carinho, pela ótima convivência e amizade.

Ao Joubert, meu marido, por me incentivar e me apoiar em todos os momentos desta pesquisa. Seu apoio foi essencial para o desenvolvimento deste trabalho.

Aos meus pais, pela vida e ensinamentos que me deram.

SUMÁRIO

LISTA DE FIGURAS	XI
LISTA DE QUADROS	XIV
LISTA DE ABREVIATURAS	XVI
LISTA DE ABREVIATURAS	XVI
RESUMO	XVIII
ABSTRACT	XIX
CAPÍTULO 1 INTRODUÇÃO	1
1.1. CONTEXTO	1
1.2. OBJETIVOS	9
1.3. ORGANIZAÇÃO DO TRABALHO	9
CAPÍTULO 2 METODOLOGIA DE PESQUISA	11
2.1. CONSIDERAÇÕES INICIAIS	11
2.2. CARACTERIZAÇÃO DAS METODOLOGIAS DE PESQUISA	11
2.1.1. CLASSIFICAÇÃO DOS MÉTODOS DE PROCEDIMENTOS DE PESQUISA	13
2.3. SELEÇÃO DA METODOLOGIA DE PESQUISA	14
2.3.1. SELEÇÃO DOS PROCEDIMENTOS DE PESQUISA	15
2.4. PLANO METODOLÓGICO ADOTADO	17
2.5. LIMITAÇÕES DA PESQUISA	19
2.6. CONSIDERAÇÕES FINAIS	19
CAPÍTULO 3 MODELAGEM ORGANIZACIONAL	21
3.1. CONSIDERAÇÕES INICIAIS	21
3.2. CONTEXTO	21
3.3. ABORDAGENS DE MODELAGEM ORGANIZACIONAL	22
3.4. CONSIDERAÇÕES SOBRE AS TÉCNICAS	24
3.5. EKD	31
3.5.1. CONHECIMENTO ORGANIZACIONAL	32
3.5.2. COMPONENTES DO EKD	35
3.5.3. COMO USAR O EKD	35
3.5.4. REFINAMENTO E OPERACIONALIZAÇÃO DOS OBJETIVOS	39
3.5.5. RELACIONAMENTO ENTRE MODELOS	40
3.5.6. CONSIDERAÇÕES SOBRE O EKD	43
3.6. CONSIDERAÇÕES FINAIS	43

CAPÍTULO 4 REDES DE PETRI 45

4.1.	CONSIDERAÇÕES INICIAIS	45
4.2.	REDES DE PETRI	45
4.3.	SUBCLASSES DE REDES DE PETRI	49
4.4.	PROPRIEDADES DAS REDES DE PETRI	50
4.5.	MÉTODOS DE ANÁLISE DAS REDES DE PETRI	53
4.6.	APLICABILIDADE DE REDES DE PETRI	54
4.7.	CONSIDERAÇÕES FINAIS	56

CAPÍTULO 5 WORKFLOW E REDES DE PETRI 57

5.1.	CONSIDERAÇÕES INICIAIS	57
5.2.	DEFINIÇÃO	58
5.3.	GERENCIAMENTO DE WORKFLOW	62
5.4.	POR QUE USAR REDES DE PETRI EM MODELAGEM DE WORKFLOW	64
5.5.	MODELANDO OS CONCEITOS DE PROCESSOS DE NEGÓCIO PARA REDES DE PETRI	72
5.6.	VERIFICAÇÃO DO MODELO DE PROCESSOS DE NEGÓCIO	81
5.7.	ANÁLISE QUALITATIVA	86
5.8.	ANÁLISE QUANTITATIVA	90
5.9.	EXEMPLO DE MODELO DE PROCESSOS DE NEGÓCIO UTILIZANDO REDES DE PETRI (COM PARÂMETROS DE TEMPO E CUSTOS)	92
5.10.	ÁRVORE DE ALCANÇABILIDADE	94
5.11.	CONSIDERAÇÕES FINAIS	95

CAPÍTULO 6 DESENVOLVIMENTO DO MÉTODO DE AVALIAÇÃO DO MODELO DE PROCESSOS DE NEGÓCIO 97

6.1.	CONSIDERAÇÕES INICIAIS	97
6.2.	MODELO DE PROCESSO DE NEGÓCIO	98
6.3.	MAPEANDO OS CONCEITOS DE PROCESSOS DE NEGÓCIO PARA REDES DE PETRI	100
6.4.	FORMALIZAÇÃO DOS CONCEITOS DO MODELO DE PROCESSOS DE NEGÓCIO	102
6.5.	FORMALIZAÇÃO DO MAPEAMENTO DO MODELO DE PROCESSOS DE NEGÓCIO EM REDES DE PETRI	108
6.6.	MÉTODO DE AVALIAÇÃO DO MODELO DE PROCESSOS DE NEGÓCIO	119
6.7.	CONSIDERAÇÕES FINAIS	124

CAPÍTULO 7 TESTES DO MÉTODO DE AVALIAÇÃO DO MODELO DE PROCESSOS DE NEGÓCIO 125

7.1.	CONSIDERAÇÕES INICIAIS	125
7.2.	CONSIDERAÇÕES SOBRE OS TESTES	125
7.3.	CASO 1 – MODELO DE SELEÇÃO DE ALUNOS	127
7.4.	CASO 2 – MODELO DE EMPRÉSTIMO	133
7.4.1.	PROPOSTA DE MODIFICAÇÃO DO MODELO DE EMPRÉSTIMO DE LIVROS	139
7.5.	CASO 3 – ELEKTRA	146
7.5.1.	MODELO DE PLANEJAMENTO ESTRATÉGICO DE RECURSOS HUMANOS	146
7.5.2.	PROPOSTA DE MODIFICAÇÃO DO MODELO DE PLANEJAMENTO ESTRATÉGICO DE RECURSOS HUMANOS	153
7.5.3.	MODELO DE AUDITORIA	160
7.5.3.	MODELO DE PLANEJAMENTO DE COMPETÊNCIAS	166
7.5.4.	MODELO DE FINALIZAÇÃO DE SCORECARD	173
7.5.5.	MODELO DE PLANEJAMENTO ESTRATÉGICO	179

7.6.	CASO 4 – DELEGACIA REGIONAL TRIBUTÁRIA DE ARARAQUARA	186
7.6.1.	MODELO DE ELABORAÇÃO DE ESCALAS	188
7.6.2.	MODELO DE ELABORAÇÃO DE ESCALAS MODIFICADO (PARTE 1)	190
7.6.3.	MODELO DE ELABORAÇÃO DE ESCALAS MODIFICADO (PARTE 2)	197
7.6.4.	MODELO DE PLANEJAMENTO DE ESCALAS DA UNIDADE	201
7.6.5.	PLANEJAMENTO DAS ESCALAS DAS EQUIPES	208
7.6.6.	MODELO DE PLANEJAMENTO DAS SUBSTITUIÇÕES DE FUNÇÕES INTERNAS	214
7.6.7.	MODELO DE ANÁLISE DE PEDIDO DE ALTERAÇÃO DE FÉRIAS	220
7.6.8.	MODELO DE ANÁLISE DE PEDIDO DE ALTERAÇÃO DE EQUIPE	226
7.7.	CONSIDERAÇÕES FINAIS	233

CAPÍTULO 8 CONCLUSÃO	234
-----------------------------	------------

8.1.	CONSIDERAÇÕES INICIAIS	234
8.2.	CONCLUSÃO	235
8.3.	TRABALHOS FUTUROS	239

REFERÊNCIAS BIBLIOGRÁFICAS	242
-----------------------------------	------------

Lista de Figuras

<i>Figura 1- Plano metodológico adotado.....</i>	<i>19</i>
<i>Figura 2 - Relações entre submodelos</i>	<i>42</i>
<i>Figura 3 - Elementos básicos das Redes de Petri.....</i>	<i>46</i>
<i>Figura 4 - Conceitos e Terminologia Básica de Workflow</i>	<i>59</i>
<i>Figura 5 - Fases e componentes de um sistema de negócios.....</i>	<i>63</i>
<i>Figura 6 - Primitivas de Processos de Negócio</i>	<i>68</i>
<i>Figura 7 - Uma descrição baseada em evento de um procedimento de workflow</i>	<i>69</i>
<i>Figura 8 - Uma descrição baseada em estados de um procedimento de processos de negócio.....</i>	<i>69</i>
<i>Figura 9 - Tarefa B e C são competitivas.....</i>	<i>71</i>
<i>Figura 10 - O processo gerenciar reclamação modelado em Redes de Petri.....</i>	<i>74</i>
<i>Figura 11 - Exemplo de modelo em que cada caso usa uma ou mais marcas.....</i>	<i>75</i>
<i>Figura 12 - Rota sequencial</i>	<i>75</i>
<i>Figura 13- Rota paralela.....</i>	<i>76</i>
<i>Figura 14 - Rota seletiva 1</i>	<i>77</i>
<i>Figura 15 - Rota seletiva 2</i>	<i>77</i>
<i>Figura 16 - Rota seletiva 3</i>	<i>78</i>
<i>Figura 17 - Rota seletiva 4</i>	<i>79</i>
<i>Figura 18 - Rota repetitiva (iterativa).....</i>	<i>80</i>
<i>Figura 19 - Rota repetitiva (iterativa) 2.....</i>	<i>80</i>
<i>Figura 20 - Uma definição de modelo incorreta.....</i>	<i>83</i>
<i>Figura 21 - Processo de uma companhia de seguros incluindo as informações tempo e custo.....</i>	<i>93</i>
<i>Figura 22 - Exemplo de um processo verificação do endereço do cliente</i>	<i>99</i>
<i>Figura 23- O processo verificação de endereço é decomposto em quatro sub-processos</i>	<i>100</i>
<i>Figura 24 - Construções que não são permitidas em um MPN-EKD formal</i>	<i>103</i>
<i>Figura 25 - Estratégia para tratar as construções que não são permitidas.....</i>	<i>104</i>
<i>Figura 26 - Construções que não são permitidas de acordo com as definições de conectores.....</i>	<i>105</i>
<i>Figura 27 - Estratégia para tratar conectores ligados a conectores</i>	<i>105</i>
<i>Figura 28- Exemplo de Mapeamento do C_{AND} entre dois ou mais inf-sets para um processo</i>	<i>110</i>
<i>Figura 29- Exemplo de Mapeamento do C_{AND} entre dois ou mais processos para um inf-set</i>	<i>111</i>
<i>Figura 30- Exemplo de Mapeamento C_{OR} entre dois ou mais inf-sets para um processo.....</i>	<i>112</i>
<i>Figura 31- Exemplo de Mapeamento C_{OR} entre dois ou mais processos para um inf-set.....</i>	<i>113</i>
<i>Figura 32- Exemplo de Mapeamento do C_{AND} um inf-set para dois ou mais processos.....</i>	<i>114</i>
<i>Figura 33- Exemplo de Mapeamento do C_{AND} de um processo para dois ou mais inf-sets.....</i>	<i>115</i>
<i>Figura 34- Exemplo de Mapeamento do C_{OR} de um inf-set para dois ou mais processos.....</i>	<i>116</i>
<i>Figura 35- Exemplo de Mapeamento do C_{OR} de um processo para dois ou mais inf-sets.....</i>	<i>117</i>
<i>Figura 36 - Esquema do método de avaliação do modelo de processos de negócio.....</i>	<i>123</i>
<i>Figura 37 - Modelo de Processo de Seleção de alunos.....</i>	<i>128</i>
<i>Figura 38- Modelo de Seleção de Alunos Mapeado em Redes de Petri.....</i>	<i>129</i>
<i>Figura 39 - Simulação do Modelo de Seleção de Alunos na ferramenta Petri Net Tools.....</i>	<i>131</i>
<i>Figura 40 - Árvore de Alcançabilidade do Modelo de Seleção de Alunos.....</i>	<i>132</i>
<i>Figura 41 - Modelo de Processos de empréstimos de biblioteca original.....</i>	<i>134</i>
<i>Figura 42 - Modelo de Processos de empréstimos de biblioteca</i>	<i>135</i>
<i>Figura 43 - Modelo de Processo de biblioteca mapeado em redes de Petri</i>	<i>136</i>
<i>Figura 44 - Simulação do Modelo de Empréstimo de Biblioteca na ferramenta Petri Net Tools.....</i>	<i>138</i>
<i>Figura 45 - Árvore de Alcançabilidade do Modelo de Empréstimo de Biblioteca.....</i>	<i>138</i>
<i>Figura 46 - Modelo de Processos de empréstimos de biblioteca com modificações na estrutura do modelo</i>	<i>141</i>
<i>Figura 47 - Modelo de Processo de biblioteca com modificações na estrutura mapeado em redes de Petri</i>	<i>142</i>
<i>Figura 48 - Simulação do Modelo com modificações na estrutura de Empréstimo de Biblioteca na ferramenta Petri Net Tools.....</i>	<i>144</i>
<i>Figura 49 - Árvore de Alcançabilidade do Modelo de Empréstimo de Biblioteca.....</i>	<i>145</i>
<i>Figura 50 - Modelo de Processo de planejamento estratégico de recursos humanos original.....</i>	<i>148</i>
<i>Figura 51 - Modelo de Processo de planejamento estratégico de recursos humanos modificado.....</i>	<i>149</i>
<i>Figura 52 - Modelo de Planejamento estratégico mapeado em redes de Petri.....</i>	<i>150</i>

<i>Figura 53 - Simulação do Modelo de Planejamento estratégico na ferramenta Petri Net Tools.....</i>	<i>152</i>
<i>Figura 54 – Árvore de alcançabilidade do Modelo de Planejamento estratégico mapeado em redes de Petri.....</i>	<i>152</i>
<i>Figura 55 - Modelo de Processo de planejamento estratégico de recursos humanos com proposta de modificação</i>	<i>155</i>
<i>Figura 56 - Modelo de Planejamento estratégico com a proposta de modificação mapeado em redes de Petri.....</i>	<i>156</i>
<i>Figura 57 - Simulação do Modelo com a proposta de modificação de Planejamento estratégico na ferramenta Petri Net Tools</i>	<i>158</i>
<i>Figura 58 - Árvore de Alcançabilidade do Modelo de Planejamento de Recursos humanos com a proposta de modificação</i>	<i>159</i>
<i>Figura 59 - Modelo de Processos de Negócio de Auditoria (Vatterfall) original</i>	<i>161</i>
<i>Figura 60 - Modelo de Processos de Negócio de Auditoria (Vatterfall) alterado.....</i>	<i>162</i>
<i>Figura 61 - Modelo de Processos de Negócio de Auditoria mapeado em redes de Petri</i>	<i>163</i>
<i>Figura 62 - Simulação do Modelo de Auditoria na ferramenta Petri Net Tools</i>	<i>165</i>
<i>Figura 63 - Árvore de alcançabilidade do modelo de Auditoria.....</i>	<i>165</i>
<i>Figura 64 - Modelo de Processos de Negócio Planejamento de Competências (Vatterfall) original.....</i>	<i>168</i>
<i>Figura 65 - Modelo de Processos de Negócio Planejamento de Competências (Vatterfall) alterado</i>	<i>169</i>
<i>Figura 66 - Modelo de Processos de Negócio de Planejamento de Competências mapeado em redes de Petri.....</i>	<i>170</i>
<i>Figura 67 - Simulação do Modelo de Planejamento de Competências na ferramenta Petri Net Tools ...</i>	<i>172</i>
<i>Figura 68 - Árvore de Alcançabilidade do Modelo de Processos de Negócio de Planejamento de Competências</i>	<i>173</i>
<i>Figura 69 - Modelo de Processos de Negócio Finalizar scorecard (Vatterfall) original com acréscimo dos conectores.....</i>	<i>175</i>
<i>Figura 70 - Modelo de Processos de Negócio Finalizar scorecard (Vatterfall) alterado.....</i>	<i>176</i>
<i>Figura 71 - Modelo de Processos de Negócio Finalizar scorecard mapeado em redes de Petri.....</i>	<i>177</i>
<i>Figura 72 - Simulação do Modelo Finalizar scorecard na ferramenta Petri Net Tools</i>	<i>178</i>
<i>Figura 73 - Árvore do Modelo de Processos de Negócio Finalizar scorecard</i>	<i>179</i>
<i>Figura 74 - Modelo de Processo de Negócio de Plano Estratégico (Vattenfall) original.....</i>	<i>181</i>
<i>Figura 75 - Modelo de Processos de Negócio Projeto de Plano Estratégico (Vattenfall) alterado.....</i>	<i>182</i>
<i>Figura 76 - Mapeamento do Modelo de Processos de Negócio Projeto de Plano Estratégico em redes de Petri.....</i>	<i>183</i>
<i>Figura 77 - Simulação do Modelo Projeto de Plano estratégico na ferramenta Petri Net Tools</i>	<i>185</i>
<i>Figura 78 - Árvore de alcançabilidade Modelo de Processos de Negócio Projeto de Plano Estratégico</i>	<i>186</i>
<i>Figura 79- Modelo de Processos de Negócio de Elaboração de Escalas original</i>	<i>191</i>
<i>Figura 80 -Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 1</i>	<i>192</i>
<i>Figura 81 -Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 1 mapeado em redes de Petri.....</i>	<i>193</i>
<i>Figura 82 - Simulação do Modelo de Processos de Negócio de Elaboração de Escalas na ferramenta Petri Net Tools.....</i>	<i>195</i>
<i>Figura 83 - Árvore de Alcançabilidade do Modelo de Elaboração de Escalas parte 1</i>	<i>196</i>
<i>Figura 84 -Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 2</i>	<i>198</i>
<i>Figura 85 - Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 2 mapeado em redes de Petri.....</i>	<i>199</i>
<i>Figura 86 - Simulação do Modelo de Processos de Negócio de Elaboração de Escalas na ferramenta Petri Net Tools.....</i>	<i>201</i>
<i>Figura 87 - Árvore de Alcançabilidade do Modelo de Escalas modificado parte 2.....</i>	<i>201</i>
<i>Figura 88 - Modelo de Processos de Negócio Planejamento das Escalas da Unidade original</i>	<i>203</i>
<i>Figura 89 - Modelo de Processos de Negócio Planejamento das Escalas da Unidade</i>	<i>204</i>
<i>Figura 90 - Modelo de Planejamento de Escalas da unidade mapeado em redes de Petri</i>	<i>205</i>
<i>Figura 91 - Simulação do Modelo de Planejamento de Escalas da Unidade na ferramenta Petri Net Tools</i>	<i>207</i>
<i>Figura 92 - Árvore de Alcançabilidade do modelo Planejamento de Escalas da Unidade.....</i>	<i>207</i>
<i>Figura 93 - Modelo de Processos de Negócio de Planejamento das Escalas das equipes original.....</i>	<i>209</i>
<i>Figura 94 - Modelo de Processos de Negócio de Planejamento das Escalas das equipes alterado.....</i>	<i>210</i>
<i>Figura 95 - Modelo de Planejamento das Escalas das equipes mapeado em redes de Petri.....</i>	<i>211</i>
<i>Figura 96 – Simulação do Modelo de Processos de Negócio de Planejamento das Escalas das equipes na ferramenta Petri Net Tools.....</i>	<i>213</i>

<i>Figura 97 - Arvore do Modelo de Planejamento das Escalas das equipes</i>	213
<i>Figura 98 - Modelo de Processos de Negócio de Planejamento das substituições de funções internas original</i>	215
<i>Figura 99 - Modelo de Processos de Negócio de Planejamento das substituições de funções internas alterado</i>	216
<i>Figura 100 - Modelo de Processos de Negócio de Planejamento das substituições de funções internas mapeado em redes de Petri</i>	217
<i>Figura 101 - Simulação do Modelo de Processos de Negócio de Planejamento das substituições de funções internas na ferramenta Petri Net Tools</i>	219
<i>Figura 102 - Árvore de Alcançabilidade do Modelo de Planejamento das substituições de funções internas</i>	219
<i>Figura 103 - Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna original</i>	221
<i>Figura 104 - Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna alterado</i>	222
<i>Figura 105 - Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna mapeado em redes de Petri</i>	223
<i>Figura 106 - Simulação do Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna</i>	225
<i>Figura 107 - Árvore de alcançabilidade do Modelo de Análise de pedido/ necessidade interna</i>	226
<i>Figura 108 - Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe original</i>	228
<i>Figura 109 - Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe alterado</i>	229
<i>Figura 110 - Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe mapeado em redes de Petri</i>	230
<i>Figura 111 - Simulação do Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe na ferramenta Petri Net Tools</i>	232
<i>Figura 112 - Árvore de Alcançabilidade do Modelo de Análise de Pedido de alteração de equipe</i>	233

Lista de Quadros

<i>Quadro 1 - Caracterização do Trabalho de pesquisa</i>	<i>16</i>
<i>Quadro 2 - Comparação entre as técnicas.....</i>	<i>30</i>
<i>Quadro 3 - Características das técnicas</i>	<i>31</i>
<i>Quadro 4 - Benefícios do EKD.....</i>	<i>33</i>
<i>Quadro 5 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de seleção de alunos</i>	<i>130</i>
<i>Quadro 6 - Conectores do Modelo de Seleção de Alunos e a regra de mapeamento correspondente</i>	<i>130</i>
<i>Quadro 7 - Elementos de Redes de Petri correspondentes aos elementos do MPN- EKD de Empréstimos de livros</i>	<i>137</i>
<i>Quadro 8 - Conectores utilizados no Modelo de Empréstimo de Biblioteca e a regra de mapeamento correspondente</i>	<i>137</i>
<i>Quadro 9 - Elementos de Redes de Petri correspondentes aos elementos do MPN- EKD de Empréstimos de livros com a modificações na estrutura.....</i>	<i>143</i>
<i>Quadro 10 - Conectores utilizados no Modelo de Empréstimo de Biblioteca com modificações na estrutura e a regra de mapeamento correspondente</i>	<i>143</i>
<i>Quadro 11 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Planejamento estratégico</i>	<i>151</i>
<i>Quadro 12 - Conectores utilizados no Modelo de Planejamento estratégico de recursos humanos e a regra de mapeamento correspondente</i>	<i>151</i>
<i>Quadro 13 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Planejamento estratégico com proposta de modificação</i>	<i>157</i>
<i>Quadro 14 - Conectores utilizados no Modelo de Planejamento estratégico com proposta de modificação de recursos humanos e a regra de mapeamento correspondente</i>	<i>157</i>
<i>Quadro 15 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Auditoria.</i>	<i>164</i>
<i>Quadro 16 - Conectores do Modelo de Auditoria e as regras de mapeamento correspondente.....</i>	<i>164</i>
<i>Quadro 17 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Planejamento de Competências.....</i>	<i>171</i>
<i>Quadro 18 - Conectores do Modelo Planejamento de Competências e a regra de mapeamento correspondente</i>	<i>171</i>
<i>Quadro 19 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Finalizar scorecard.....</i>	<i>178</i>
<i>Quadro 20 - Conectores do modelo Finalizar scorecard e a regra de mapeamento correspondente.....</i>	<i>178</i>
<i>Quadro 21 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Projeto de Plano Estratégico.....</i>	<i>184</i>
<i>Quadro 22 - Conectores do Modelo Projeto Plano Estratégico e a regra de mapeamento correspondente</i>	<i>184</i>
<i>Quadro 23 - Elementos de redes de Petri correspondentes aos elementos do MPN_EKD de Elaboração de Escalas.....</i>	<i>194</i>
<i>Quadro 24 - Conectores do Modelo de Elaboração de Escalas e as regras de mapeamento correspondente</i>	<i>194</i>
<i>Quadro 25 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Elaboração de Escalas modificado parte 2.....</i>	<i>200</i>
<i>Quadro 26 - Conectores do Modelo de Elaboração de Escalas modificado parte 2 e as regras de mapeamento correspondentes</i>	<i>200</i>
<i>Quadro 27 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Planejamento das Escalas da Unidade</i>	<i>206</i>
<i>Quadro 28 - Conectores do Modelo de Elaboração de Escalas da Unidade e as regras de mapeamento correspondente</i>	<i>206</i>
<i>Quadro 29 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Planejamento das Escalas das equipes</i>	<i>212</i>
<i>Quadro 30 - Conectores do Modelo de Planejamento das Escalas das equipes e as regras de mapeamento correspondente</i>	<i>212</i>
<i>Quadro 31 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Planejamento das substituições de funções internas</i>	<i>218</i>

<i>Quadro 32 - Conectores do Modelo de Planejamento das substituições de funções internas e as regras de mapeamento correspondente</i>	<i>218</i>
<i>Quadro 33 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Análise de pedido/necessidade de alteração interna</i>	<i>224</i>
<i>Quadro 34 - Conectores do Modelo de Análise de pedido/necessidade de alteração interna e as regras de mapeamento correspondente</i>	<i>224</i>
<i>Quadro 35 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Análise de Pedido de alteração de férias de equipe</i>	<i>231</i>
<i>Quadro 36 - Conectores do Modelo de Análise de Pedido de alteração de férias de equipe e as regras de mapeamento correspondente</i>	<i>231</i>

Lista de Abreviaturas

BP-nets	Business Procedure net (BP-net)
BPP-nets	Business Process Petri net
CGSPN	Coloured Generalised stochastic Petri Net
CPN	Coloured Petri Net
CTMC	Continuous Time Markov chain
ECOM	European Electricity Company
EKD	Enterprise Knowledge Development
ELEKTRA	ELectrical Enterprise Knowledge for TRansforming Applications.
EU	European Union
GSPN	Generalized stochastic Petri Net
HOON	Higher-Order Object Net
ICN	Information Control Net
LEA	Load Equivalence Aggregation
MA	Modelo de Atores
MAU	Modelo de Atividades e Uso
MC	Modelo de Conceitos
MO	Modelo de Objetivos
MPN	Modelo de Processos de Negócio
MPN-EKD	Modelo de Processos de Negócio do EKD
MRSI	Modelo de Requisitos do Sistema de Informação
PPC	Public Power Corporation
Relação ISA	Generalização
SD	Modelo de Dependências Estratégicas
SGML	Standard Generalized Markup Language

SR	Modelo de Razões Estratégicas
TCPN	Temporal Constraint Petri Net
WAN	Workflow Authorization model
WfMS	Workflow management System

Resumo

PÁDUA, S.I.D. (2004). Método de avaliação do Modelo de Processos de Negócio do EKD. São Carlos. Tese (Doutorado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2004.

Atualmente as empresas precisam de sistemas ágeis a mudanças constantes do ambiente do negócio e para garantir que os sistemas cumpram com sua finalidade, os desenvolvedores devem possuir uma compreensão mais aprofundada sobre a organização, seus objetivos, metas e estratégias de mercado. O principal problema para o desenvolvimento de sistemas de software tem sido a dificuldade em se obter informações sobre o domínio da aplicação. Essa dificuldade levou ao surgimento de técnicas de *modelagem organizacional*, sendo uma atividade valiosa para a compreensão do ambiente empresarial. O *EKD* – Enterprise Knowledge Development – é uma metodologia que fornece uma forma sistemática e controlada de analisar, entender, desenvolver e documentar uma organização. Infelizmente não possui uma sintaxe e semântica bem definidas, dificultando análises mais complexas dos modelos. Como resultado, o modelo de processos de negócio do *EKD* pode ser ambíguo e de difícil análise, principalmente em sistemas mais complexos, não sendo possível verificar a consistência e completude do modelo. Neste trabalho, esses problemas serão estudados sob uma abordagem baseada em *redes de Petri*. O formalismo de redes de Petri a torna uma importante técnica de modelagem para a representação de processos. Além disso, *redes de Petri* permitem rastrear cada etapa da operação sem ambigüidade e possuem métodos eficientes de análise que garantem que o modelo está livre de erros. Assim, este trabalho tem como objetivo desenvolver um método de avaliação do *modelo de processos de negócio* do *EKD* (MPN-EKD). Por meio desse método é possível verificar se o modelo tem erros de construção e travamentos. Este método pode ser aplicado em modelos direcionados para o desenvolvimento de sistema de informação ou de controle do fluxo de trabalho, e pode ser utilizado também para o estudo de estratégias de trabalho e simulação do fluxo de trabalho.

Palavras-chaves: Redes de Petri, Modelagem Organizacional, Regras do Negócio, Desenvolvimento de Software, Modelo de Processos de Negócio, *Workflow*, EKD, Avaliação do Modelo de Processos de Negócio.

Abstract

PÁDUA, S.I.D. (2004). Assessment Method of Business Process Model of EKD. São Carlos. Ph.D. Thesis – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2004.

Nowadays all companies need fast systems and frequent changes on the business environment and to guarantee that the systems are reaching their goals, the developers must have a deeper comprehension of the enterprise, its goals and market strategies. The main problem to the development of software systems has been the difficulty to obtain information about the application domain. This difficulty led to the creation of Enterprise Modeling techniques, which is a valuable activity for the comprehension of business environment. The EKD – Enterprise Knowledge Development - is a methodology that gives a systematic and controlled way to analyze, understand, develop, and document an Enterprise. Unfortunately it doesn't have syntax neither a semantic well defined, which doesn't help on more complex analyses of the models. As a result, the Enterprise Process model of EKD can be ambiguous and hard to analyze, especially on more complex systems, and also it is not possible to verify the consistency and entirety of the model. On this paper, these problems will be studied under an approach based on Petri Nets. Because of the Petri Nets formalism this is an important modeling technique to process representation. Furthermore, Petri Nets allow the tracking of each step of the operation without ambiguity and also they have efficient methodology for analyses, which guarantee the accuracy of the model. Therefore, this work has the objective to develop an evaluation methodology of the business process model of EKD (MPN-EKD). Such methodology will make possible the verification of possible building and locking model errors. This methodology can be applied to information systems or workflow, and also can be used to study the strategies of work and workflow simulations.

Key words: Petri Nets, Enterprise Modeling, Business Rules, Software development, Business Process Model, Workflow, EKD, Assessment of Business Process Model

CAPÍTULO 1

Introdução

1.1. Contexto

Em um mercado competitivo, as organizações precisam encontrar soluções melhores, com estruturas flexíveis e confiáveis. Muito da infra-estrutura da empresa é sustentada pela tecnologia da informação, que diretamente ou indiretamente apóia os processos de negócio com importância crucial para o seu crescimento e sobrevivência. Por meio do gerenciamento eficiente desses processos, muitas vantagens competitivas podem ser obtidas, como redução de custo, aumento de produção e melhoria dos serviços ao cliente.

As mudanças do mercado têm-se tornado uma constante, o que revela uma necessidade clara e imperativa de um melhoramento contínuo dos processos de negócio. Uma vez que os processos de negócio são blocos fundamentais da construção de uma organização de sucesso, a tecnologia da informação direcionada para o gerenciamento e melhoria dos processos de negócio tem ajudado a organização a completar sua visão da empresa e a melhorar sua posição competitiva. De acordo com Kruchten (2000), Kozlenkov & Zisman (2002) e Alencar (1999), a tecnologia da informação não deveria ser utilizada apenas para automação dos processos de negócio existentes, mas também ser uma base para a reformulação desses processos visando a atingir objetivos essenciais como concorrência, competitividade e estratégias.

Um sistema de informação é uma solução organizacional e administrativa baseada na tecnologia da informação para administrar, controlar e auxiliar os executivos na gestão empresarial (Laudon & Laudon, 2003). Existem muitos sistemas que, embora

tecnicamente corretos, não satisfazem as reais necessidades do negócio, atrapalhando o seu desenvolvimento. As organizações precisam de sistemas ágeis à mudanças constantes do ambiente do negócio e para garantir que os sistemas cumpram com sua finalidade, os desenvolvedores de software precisam possuir uma compreensão mais aprofundada sobre a organização, seus objetivos, metas e estratégias do mercado.

A necessidade de automatizar processos manuais buscando o aumento da produtividade, qualidade e eficiência da organização é, na maioria das vezes, o que impulsiona o processo de engenharia de software. Ao iniciar um processo de engenharia de software é importante definir os requisitos do ambiente, fase essa considerada a mais trabalhosa de todos os estágios de desenvolvimento (Alencar, 1999). Muitos dos problemas oriundos de sistemas mal desenvolvidos são causados por falhas na fase de análise, resultando na produção de um sistema que não satisfaz às necessidades do cliente. De acordo com Kozlenkov & Zisman (2002), o que se observa são usuários insatisfeitos com softwares que não se adequam às suas reais necessidades. Como consequência, diversas tarefas do ciclo de desenvolvimento de software devem ser refeitas, elevando o seu custo.

Segundo Achour *et al.* (1999), para desenvolver um sistema com qualidade é necessário requisitos de qualidade. Para isso, é necessário o entendimento dos aspectos sociais, organizacionais, técnicos, econômicos e jurídicos envolvidos no ambiente onde o sistema será inserido. Em geral, a captura e entendimento desses requisitos não são tarefas simples, por se tratar, muitas vezes, de problemas ainda abstratos para o cliente (Alencar, 1999). Nesse contexto, a Engenharia de Requisitos surge como uma forma de contribuir para a melhoria da qualidade dos sistemas.

A Engenharia de Requisitos corresponde ao processo de elicitação, especificação, modelagem, validação, gerenciamento e documentação das necessidades do cliente para um sistema de software (Sommerville, 2001). Como um processo cooperativo, a aquisição requisitos envolve múltiplos *stakeholders*¹ os quais visam à comunicação e entendimento entre usuários, clientes, especialistas no domínio, gerentes de projeto, entre outros. Muitos desses *stakeholders* têm pouca habilidade de Engenharia de Requisitos e apresentam dificuldades para realizar a análise dos requisitos (Achour *et al.*, 1999). Para Chen-Burguer *et al.* (2003), existe um vazio entre o usuário, que tem as necessidades mas não sabe como especificá-las como requisitos; e o engenheiro de

¹ O termo *Stakeholder* foi introduzido para ter um termo comum para todos os envolvidos no projeto, direta ou indiretamente, ou que tenha interesse no resultado do projeto.

requisitos que sabe especificar requisitos, mas não conhece as necessidades. Em reação a essa questão, a Engenharia de Requisitos vem deslocando o foco de interesse da tecnologia para o usuário (Achour *et al.*, 1999).

Segundo Alencar (1999), Rolland *et al.* (1999) e Santander (2002), a preocupação da Ciência da Computação esteve por muito tempo extremamente relacionada com aspectos da funcionalidade do sistema, ou seja, com "o que" e "como" fazer e não com o "por que" fazer, envolvidos nos processos decisórios existentes.

As técnicas tradicionalmente aplicadas no desenvolvimento de software tratam aspectos relacionados à funcionalidade do sistema, à descrição de atividades e entidades, às entradas que deverão ser transformadas e às saídas que deverão ser produzidas, porém sem considerarem aspectos mais amplos como: os objetivos da organização, regras do negócio, restrições, aspectos não funcionais relacionados à qualidade, confiabilidade e usabilidade. De acordo com Jacobson *et al.* (1999) e Kruchten (2000), essas técnicas não ajudam, portanto, a buscar soluções alternativas aos problemas da organização, não adicionam valor ao negócio e, na maioria das vezes, os processos manuais são automatizados sem nenhuma modificação.

Neste mesmo sentido, Grover *et al.* (1998) afirmam que investimentos em informática, sem mudanças nos processos organizacionais, podem afetar negativamente os efeitos da informatização, e que a mesma tecnologia pode apresentar resultados diferentes em empresas diferentes, dependendo da forma como ela é implementada.

Elicitar requisitos organizacionais é ainda uma das grandes dificuldades dos engenheiros de requisitos. Isso ocorre por questões sociais e organizacionais, as quais devem ser adequadamente tratadas para possibilitar o desenvolvimento de sistemas que satisfaçam realmente as metas organizacionais (Alencar, 1999).

Os requisitos organizacionais não devem ser considerados como uma simples descrição da funcionalidade do sistema, pois tratam do domínio onde o sistema está inserido e das restrições que podem existir no ambiente, no sistema e no desenvolvimento, diminuindo ambigüidades e incertezas. Nesse contexto, a modelagem organizacional facilita a compreensão do ambiente empresarial e é reconhecida como uma atividade valiosa pela Engenharia de Requisitos.

Nurcan & Barrios (2003), Persson (2000), Nurcan & Rolland (2003), Alencar (1999), Jonkers *et al* (2003a) e Santander (2002) afirmam que a Modelagem Organizacional é muito importante para o desenvolvimento de sistemas de informação.

De acordo com Alencar (1999), o modelo organizacional é uma representação da estrutura, das atividades, dos processos, das informações, dos recursos, de pessoal, do comportamento, dos aspectos jurídicos e econômicos, dos objetivos e restrições das empresas comerciais, governamentais ou de outra natureza, a fim de ajudar a compreender as complexas interações entre organizações e pessoas. Cada organização tem sua missão, seus objetivos e seus processos próprios e é importante dar atenção à modelagem desses processos. Os objetivos da modelagem organizacional são apresentados a seguir:

- Fornecer um objeto que seja uma representação compartilhável e reusável da cadeia de fornecimento de informação e conhecimento;
- Suportar tarefas da cadeia de fornecimento por meio da habilitação de respostas a questionamentos que não estão explicitamente representados no modelo;
- Definir os objetos de maneira precisa, de forma que sejam consistentemente aplicados, por meio dos domínios e interpretados pelos usuários;
- Suportar visualização do modelo de forma intuitiva, simples e consistente.

De acordo com Koubarakis & Plexousakis (2002), Delen & Benjamin (2003) e Jonkens *et al.* (2003), o problema de representar, analisar e gerenciar conhecimento sobre a organização e seus processos sempre foi muito importante, e ultimamente a comunidade científica está interessada no entendimento da organização e seus processos, facilitando o processo de análise e projeto para apoio do gerenciamento de processos. O assunto é de grande importância para a indústria como um benefício para projeto de estruturas organizacionais, processos e infra-estrutura de tecnologia da informação que atende aos objetivos do negócio em uma forma eficiente e flexível.

O processo de modelagem organizacional deve trazer respostas a essas questões: por que, o que, quem, qual, quando, onde e como. Para tanto, existem diversas técnicas de modelagem na literatura com uma significativa variedade de notações. A abordagem EKD – *Enterprise Knowledge Development*- é uma metodologia que fornece uma forma sistemática e controlada de analisar, entender, desenvolver e documentar uma

organização e seus componentes, usando a Modelagem Organizacional (Rolland *et al.*, (2000) e Nurcan (1999)).

O EKD pode ser utilizado na Engenharia de Requisitos, na análise do negócio para detecção do problema, na engenharia de processos de negócio para a definição de novos sistemas de negócio e no gerenciamento do conhecimento organizacional ou aprendizagem organizacional para formar a base de propagação e ampliação do conhecimento. Nurcan & Barrios (2003) afirmam que a modelagem do conhecimento organizacional torna-se um pré-requisito para a especificação de requisitos e desenvolvimento de sistemas. O EKD também contribui para a tomada de decisão em modernas organizações que são altamente dependentes de tecnologia de informação, conforme (Nurcan & Barrios (2003), Nurcan & Rolland (2003)).

De acordo com Bubenko *et al.* (1998), os tipos de submodelos do método EKD são: Modelo de Objetivos, Modelo de Regras do Negócio, Modelo de Conceitos, Modelo de Processos de Negócio, Modelo de Atores e Recursos e Modelo de Requisitos e Componentes Técnicos.

O modelo organizacional representa o "mundo" no qual as regras do negócio se aplicam. Muitas organizações têm reconhecido que para ser flexível em um ambiente de crescente competitividade, é importante ter um claro entendimento das regras do negócio da organização. Para Dallavalle & Cazarini (2000), entender as regras do negócio é um fator chave de sucesso para o processo de desenvolvimento de software. Regras do negócio, segundo Rosca *et al.* (1997), são uma nova categoria de requisitos do sistema que representam decisões sobre como executar o negócio, e são caracterizadas pela orientação do negócio e sua tendência às mudanças. As regras do negócio estão relacionadas aos processos de negócio, porque condições diferentes vão requerer a invocação de processos diferentes. São as regras do negócio que determinam os estados permissíveis dos objetos do negócio e a interação entre tarefas diferentes. O EKD foi a única abordagem dentre as encontradas na literatura a tratar explicitamente o Modelo de Regras do Negócio e suas ligações com outros modelos.

A importância de regras do negócio na modelagem organizacional e na especificação de requisitos tem sido discutida por diversos autores (Bubenko (1993), Bubenko & Wangler (1993), Stergiou & Johnson (1998), Rosca *et al.* (1997), Gottesdiener (1999), Herbst (1996), Pádua (2001), Kilov & Simmonds (1997), Leite & Leonarde (1998), Dallavalle & Cazarini (2000), Pádua & Cazarini (2001a), Pádua & Cazarini (2001b) e

Pádua & Cazarini (2001c)). Neste sentido, a abordagem que será utilizada neste trabalho é o EKD – *Enterprise Knowledge Development*.

Antigamente, os sistemas de informação eram desenvolvidos para apoiar a execução de tarefas individuais, já os sistemas de informação atuais precisam apoiar os processos de negócio de perto: o foco somente nas tarefas não satisfaz. O sistema de informação também precisa controlar, monitorar e apoiar os aspectos lógicos dos processos de negócio. Em outras palavras, o sistema de informação também tem que gerenciar o fluxo de trabalho por meio da organização (Aalst & Hee, 2002). Muitas organizações com processos de negócio complexos têm identificado a necessidade de conceitos, técnicas e ferramentas para apoiar o gerenciamento de *workflow*. Baseado nessa necessidade surgiu o termo gerenciamento de *workflow*. O termo *workflow* é utilizado na literatura representando processos de negócio. Um *workflow* é definido como um conjunto de atividades que podem ou não ser executadas simultaneamente, com alguma especificação de controle e fluxo de dados entre essas atividades (Leymann & Roller, 1997). Dentre os sub-modelos do método EKD, que são de abrangência organizacional, existe o modelo de processos de negócio similar aos modelos de *workflows* encontrados na literatura.

O principal problema das abordagens de Modelagem Organizacional, incluindo-se o EKD, é a ausência de técnicas de análise mais complexas. O problema da estrutura informal das técnicas de modelagem organizacional e de processos de negócio tem sido discutido por diversos autores. Entre eles, pode-se mencionar: Chen-Burger *et al.* (2002); Chen-Burger (2001); Koubarakis & Plexousakis (2002), Jonkens *et al.* (2003), Taylor & Tofts (2003), Junginger *et al.* (2001), Jonkers *et al.* (2003), Berio & Petit (2003), Chen-Burger & Stader (2003), Dehnert (2003) e Aalst (1999).

As técnicas de análise com rigor matemático não são atraentes aos engenheiros de software, contudo, com um pouco de esforço seria possível que esses profissionais os compreendessem. Porém, em se tratando dos profissionais não especificamente ligados à área de desenvolvimento de software, tais técnicas seriam de difícil compreensão (Alencar, 1999). Nesse caso, redes de Petri têm um excelente potencial para resolver esse problema, uma vez que elas possuem representação gráfica, são de fácil aprendizado, funcionam como linguagem de comunicação entre especialistas de diversas áreas, permitem a descrição dos aspectos estáticos e dinâmicos do sistema a ser representado, e ainda possuem o formalismo matemático que permite a utilização de

métodos de análise. As diversas aplicações das redes de Petri na Engenharia são apresentadas em Pádua *et al.* (2003b). Em Aalst & Hee (2002), Neiger & Churilov (2003), Chrzastowski-Wachtel *et al.* (2003), McDermid (2003), Rinderle *et al.* (2003), Verbeck *et al.* (2002), Pádua *et al.* (2002), Pádua *et al.* (2004), Janssens *et al.* (2000), Holt (2000), Desel (2000), Dellarocas & Kleins (2000), Hee & Reijers (2000), Knolmayer *et al.* (2000) e em Salimifard & Wright (2001), são apresentados o potencial das redes de Petri em modelagem e análise de processos de negócio.

Desde que Zisman (1977) usou redes de Petri para modelar *workflow* pela primeira vez, muitos autores publicaram trabalhos que procuravam, também, integrar os dois assuntos. Entre eles, pode-se mencionar: Chrzastowski-Wachtel *et al.* (2003), Rinderle *et al.* (2003), Chen-Burger & Stader (2003), Cindio *et al.* (1988), Li *et al.* (1993), Ferscha (1994), Merz *et al.* (1995), Schömig & Rau (1995), Ellis & Nutt (1996), Nutt (1996), Wikarski (1996), Oberweis (1997), Aalst (1998), Adam *et al.* (1998), Badouel & Oliver (1998), Dehnert (2003), Aalst & Hee (2002), Eshuis & Dehnert (2003), Grigova (2003) e Verbeck *et al.* (2002). Aalst & Hee (2002) afirmam que existem diversas razões para usar redes de Petri para modelagem de processos de negócio:

- 1 – Semântica formal: um processo de negócio especificado em uma rede de Petri tem definição clara e precisa, por serem a semântica da rede de Petri clássica e suas extensões definidas formalmente.
- 2 – Natureza gráfica: rede de Petri é uma linguagem gráfica intuitiva e de fácil aprendizado. A natureza gráfica também apóia a comunicação com usuários finais.
- 3 – Expressividade: redes de Petri suportam todas as primitivas necessárias para modelar um processo de negócio. Todas as construções das rotas presentes nos atuais sistemas de gerenciamento de processos de negócio podem ser modeladas. Além disso, o fato dos estados serem representados explicitamente permite escolhas explícitas na modelagem de processos.
- 4 – Propriedades: nas últimas três décadas, muitas pessoas têm investigado as propriedades básicas de redes de Petri. A fundamentação matemática firme permite algumas conclusões e verificações importantes. Por isso existem muitas publicações científicas sobre essas propriedades.
- 5 – Análise: as redes de Petri são caracterizadas pela disponibilidade de muitas técnicas de análise. Claramente, esse é um ponto positivo a favor do uso de redes de Petri

para modelagem de *workflow*. Essas técnicas podem ser usadas para provar propriedades (segurança, *deadlock*, entre outras) e para calcular métricas de *performance* (tempo de resposta, tempo de espera, taxa de ocupação). Dessa forma, é possível avaliar *workflows* alternativos usando ferramentas de análise baseadas em redes de Petri.

6 – Não dependente de fornecedor: rede de Petri é uma ferramenta de modelagem com estrutura para modelagem e análise independente de um vendedor.

Nos trabalhos de Aalst & Hee (2002), Verbeek *et al.* (2002), Aalst (2000), Salimifard & Wright (2001) Aalst (1999), Aalst & Hofstede (2000), Voorhoeve (2000), Oberweis (1997), Oberweis (1996), Mold & Valk (2000) e Merz (1995) os processos de negócio são modelados diretamente em redes de Petri. No capítulo 5 deste trabalho são apresentados trabalhos que modelam *workflow* diretamente em redes de Petri.

Neste trabalho, a construção do modelo de processos de negócio buscará seguir o método de modelagem organizacional EKD e não diretamente em redes de Petri, espera-se que o desenvolvimento do modelo organizacional utilizando o método EKD proporcione o gerenciamento de mudanças, captura das melhores práticas do negócio, gerenciamento das regras do negócio e o gerenciamento do conhecimento organizacional. O método apresenta: um conjunto de diretrizes que orienta todo o processo de modelagem, um conjunto de questões que apóia o desenvolvimento de todos os modelos e um outro conjunto de questões para apoio na verificação das ligações entre componentes de todos os modelos depois de desenvolvidos. O modelo organizacional desenvolvido utilizando o método EKD ajuda a compreender as complexas interações entre as organizações e as pessoas. Este método é orientado a objetivos e trata também os atores envolvidos nos processos, nas regras e nos objetivos da organização.

O EKD é composto de seis modelos: Modelo de Objetivos (MO), Modelo de Regras do Negócio (MRN), Modelo de Conceitos (MC), Modelo de Processos de Negócio (MPN), Modelo de Atores e Recursos (MAR) e Modelo de Requisitos e componentes técnicos (MRCT). Todos os modelos são inter-relacionados. Dessa forma, o modelo de processos de negócio desenvolvido seguindo a método EKD está relacionado aos objetivos, às regras do negócio, aos conceitos, aos atores e recursos e aos requisitos e componentes técnicos.

1.2. Objetivos

A sintaxe e a semântica do modelo de processos de negócio do EKD não são definidas formalmente. Como resultado, o modelo de processos de negócio do EKD pode ser ambíguo e de difícil análise, principalmente em sistemas mais complexos, não sendo possível verificar a consistência e completude do modelo. Por isso, o EKD não possui um método de análise. A ausência de semântica formal dificulta, também, o uso de técnicas mais eficientes de análise. Neste trabalho, esses problemas serão estudados sob uma abordagem baseada em redes de Petri. O formalismo de redes de Petri a torna uma importante técnica de modelagem para a representação de processos, permitindo a exibição de: concorrência, paralelismo, sincronização, não-determinismo e exclusão mútua. Além disso, redes de Petri permitem rastrear cada etapa da operação sem ambigüidade e possuem métodos eficientes de análise que garantem que o modelo está livre de erros.

Assim, este trabalho tem como objetivo *desenvolver um método de avaliação do modelo de processos de negócio do EKD (MPN-EKD)*. Por meio desse método será possível verificar se o modelo tem erros de construção e travamentos. Este método poderá ser aplicado em modelos direcionados para o desenvolvimento de sistemas de informação ou de controle do fluxo de trabalho, e poderá ser utilizado também para o estudo de estratégias de trabalho e simulação do fluxo de trabalho.

Para alcançar esse objetivo deverão ser desenvolvidos uma semântica formal para o Modelo de Processos de Negócio do EKD e criado um procedimento de mapeamento do Modelo de Processos de Negócio do EKD em redes de Petri.

1.3. Organização do Trabalho

Neste capítulo foi apresentado o contexto em que este trabalho insere-se, o objetivo a ser atingido e as justificativas para o seu desenvolvimento.

No capítulo 2 é apresentada a metodologia científica utilizada neste trabalho.

Nos capítulos 3, 4 e 5 são apresentados, respectivamente, a pesquisa bibliográfica sobre modelagem organizacional, redes de Petri e *workflow* e redes de Petri.

No capítulo 6 são apresentados a formalização do modelo de processos de negócio do EKD, mapeamento do modelo de processos de negócio do EKD em redes de Petri, a primeira versão do método de avaliação do modelo de processos de negócio.

No capítulo 7 são apresentados os testes da primeira versão do método de avaliação do modelo de processos de negócio.

No capítulo 8 é apresentada a conclusão deste trabalho e sugestões para trabalhos futuros nesta área de pesquisa.

CAPÍTULO 2

Metodologia de Pesquisa

2.1. Considerações Iniciais

Neste capítulo são discutidos os aspectos metodológicos considerados na realização desta pesquisa. Para isso, inicia-se com os conceitos intrínsecos ao contexto metodológico e com a caracterização das diferentes metodologias de pesquisa, em seguida, os métodos de procedimentos de pesquisas e por fim, são definidas as fases da pesquisas e as etapas do trabalho.

2.2. Caracterização das Metodologias de Pesquisa

A pesquisa é a atividade básica da ciência pela qual se desvenda a realidade. Ela parte de uma dúvida ou problema, e com o uso do método científico, busca uma resposta ou solução (Cervo & Bervian, 1983).

A escolha de um problema relevante e a identificação de suas contingências determinarão em boa parte o tipo de pesquisa a ser desenvolvida, restando escolher qual será o método de abordagem do problema para o qual é proposto uma solução científica (Salomon, 1991).

De acordo com Demo (1987), a metodologia é uma preocupação instrumental, isto é, a forma de se fazer ciência, cuidando dos caminhos, dos procedimentos, e das ferramentas. A finalidade da ciência é tratar a realidade teórica e prática, propondo-se a captar e manipular a realidade como ela é. A metodologia possibilita desenvolver o mecanismo de como atingir essa finalidade. A idéia que se faz da realidade precede a idéia de como tratá-la, tornando clara a posição instrumental da metodologia, ou seja, a serviço da captação da realidade.

Lakatos & Marconi (1995) definem método de pesquisa como: o conjunto de atividades sistemáticas e racionais que, com maior segurança e economia, permite alcançar um objetivo – conhecimentos válidos e verdadeiros – traçando o caminho a ser seguido, detectando erros e auxiliando as decisões do cientista. Os métodos de pesquisa são métodos mais amplos que tratam de questões de forma mais abstrata e genérica, como se fossem métodos básicos, ou seja é a linha de raciocínio adotada no processo de pesquisa. Já os métodos de procedimento são etapas mais concretas de busca de solução, sendo restritos a certos tipos de fenômenos, isto é, eles proporcionam os meios técnicos para a investigação.

D’Onofrio (1999) define método como o caminho a percorrer para alcançar objetivos específicos. A escolha do caminho para atingir a verdade implica na utilização de meios adequados para cada tipo de conhecimento, variando estes conforme o assunto e a finalidade.

Demo (1987) ressalta a relevância de se dedicar à discussão sobre os caminhos seguidos pelos autores para construir suas teorias, contrastando com outros caminhos e construindo um caminho próprio.

Apesar da importância reconhecida do método científicos, atualmente, percebe-se que a ciência não é fruto de um roteiro de criação totalmente previsível. Portanto, não há apenas uma maneira de raciocínio capaz de dar conta do complexo mundo das investigações científicas (Feyerabend, 1989).

Dentre os métodos de pesquisa, o que mais se aproxima ao objetivo deste trabalho é o método hipotético-dedutivo. De acordo com Gil (1999), o método hipotético-dedutivo consiste da seguinte linha de raciocínio: quando os conhecimentos disponíveis sobre determinado assunto são insuficientes para explicação de um fenômeno, surge o problema. Para tentar explicar as dificuldades expressas no problema, são formuladas conjecturas ou hipóteses. Das hipóteses formuladas, deduzem-se consequências que deverão ser testadas ou falseadas.

Esse método foi escolhido neste trabalho uma vez que o objetivos é desenvolver um método de avaliação do modelo de processos de negócio do EKD. Esse método é desenvolvido por meio da construção e verificação de hipóteses de mapeamento do modelo de processos de negócio EKD em redes de Petri. O mapeamento deverá ser testado e não falseado.

Este método permite ao pesquisador levantar um conjunto de postulados que governam os fenômenos pelos quais está interessado e, por meio de experimentos, deduzir as conseqüências e talvez os postulados, substituindo-os por outros.

Existem diversas abordagens possíveis para o estudo do problema de pesquisa e sua aplicabilidade. Para Salomon (1991), o problema detectado é que determina o tipo de pesquisa científica a ser empreendida, a qual pode ser classificada quanto a sua ênfase de duas formas:

- Pesquisa Fundamental: para desenvolver teorias mediante o descobrimento de amplas generalizações ou princípios, para o avanço da ciência sem aplicação prática prevista.
- Pesquisa Aplicada: para modificar instituições por meio da ação conjugada se seus responsáveis que devem encontrar as práticas a serem mudadas, satisfazendo necessidades e demandas da vida moderna, e recolhendo metódica e sistematicamente os resultados que comprovam seu valor.

Essa pesquisa pode ser caracterizada como pesquisa aplicada, uma vez que objetiva gerar conhecimentos novos úteis para o avanço da ciência e apresenta testes que são aplicados com a intenção de solucionar problemas específicos.

Do ponto de vista da forma da abordagem do problema pode ser pesquisa quantitativa ou qualitativa. A pesquisa quantitativa considera que tudo pode ser quantificável, o que significa traduzir em números opiniões e informações para classificá-las e analisá-las. Requer o uso de recursos e de técnicas estatísticas. A pesquisa qualitativa não procura enumerar ou medir os eventos estudados, nem emprega instrumental estatístico na análise dos dados; ela parte de questões e focos de interesse mais amplos, que vão se definindo à medida que o estudo se desenvolve, procurando compreender os fenômenos segundo a perspectiva dos participantes da situação em estudo (Bryman, 1989).

Este trabalho se caracteriza como uma abordagem qualitativa de acordo com essa classificação, uma vez que este trabalho não procura provar estatisticamente o sucesso ou o fracasso do método proposto.

2.1.1. Classificação dos Métodos de Procedimentos de Pesquisa

Para se confrontar a visão teórica do problema com os dados da realidade é necessário o delineamento da pesquisa. O delineamento refere-se ao planejamento da pesquisa com a indicação preliminar dos recursos que serão utilizados para a coleta de dados (Pádua,

1996). As principais classificações dos métodos de procedimentos de pesquisa são (Cervo & Bervian, 1983; Gil, 1991; Pádua, 1996):

- **Pesquisa Bibliográfica.** Pesquisa desenvolvida a partir de materiais já elaborados, principalmente, livros e artigos científicos. A finalidade é colocar o pesquisador em contato com o que já se produziu e registrou sobre o tema de pesquisa.
- **Pesquisa Documental.** Quando elaborada a partir de materiais que não receberam tratamento analítico.
- **Pesquisa Experimental.** Pesquisa que consiste na busca de relações entre fatos sociais ou fenômenos físicos pela identificação e pela manipulação das variáveis que determinam a relação causa-efeito. Consiste em determinar um objeto de estudo, selecionar as variáveis que seriam capazes de influenciá-lo, definir as formas de controle e a observação dos efeitos que a variável produz.
- **Pesquisa de Levantamento (Survey).** Quando a pesquisa envolve a interrogação direta das pessoas cujo comportamento se deseja conhecer.
- **Pesquisa Estudo de Caso.** Quando envolve o estudo profundo e exaustivo de um ou poucos objetos de maneira que se permita o seu amplo e detalhado conhecimento.
- **Pesquisa-Ação.** Thiollent (1998) caracteriza pesquisa-ação como “pesquisa com base empírica que é concebida e realizada em estreita associação com uma ação ou com a resolução de um problema coletivo, no qual os procedimentos e participantes representativos da situação ou do problema estão envolvidos de modo cooperativo ou participativo”. É uma abordagem que exige o envolvimento ativo do pesquisador e a ação por parte das pessoas ou dos grupos envolvidos no problema.
- **Pesquisa Participante.** Quando se desenvolve a partir da interação entre pesquisadores e membros das situações investigadas.

2.3. Seleção da Metodologia de Pesquisa

A adoção de um ou outro método depende de muitos fatores, tais como: da natureza do objeto que se pretende pesquisar, dos recursos disponíveis, do nível de abrangência do estudo, da adequação do método aos conceitos envolvidos e em relação aos objetivos da pesquisa e confiabilidade (Yin, 1989; Gil, 1999).

Segundo Chizzotti (1995), a delimitação do problema não resulta de uma afirmação prévia e individual, formulada pelo pesquisador e para a qual recolhe dados comprobatórios. O problema decorre, antes de tudo, de um processo indutivo que se vai definindo e se delimitando na exploração dos contextos ecológico e social, onde se realiza a pesquisa; da observação reiterada e participante do objeto pesquisado, e dos contatos duradouros com informantes que conhecem esse objeto e emitem juízos sobre ele.

Ainda segundo Chizzotti (1995), o problema afigura-se como um obstáculo, percebido pelos sujeitos de modo parcial e fragmentado, e analisado assistematicamente. A identificação do problema e sua delimitação pressupõem uma imersão do pesquisador na vida e no contexto, no passado e nas circunstâncias presentes que condicionam o problema. Pressupõem, também, uma partilha prática nas experiências e percepções que os sujeitos possuem desses problemas, para descobrir os fenômenos além de suas aparências imediatas. A delimitação é feita, pois, em campo onde a questão inicial é explicitada, revista e reorientada a partir do contexto e das informações das pessoas ou grupos envolvidos na pesquisa.

Inicialmente, se define a *questão da pesquisa* e, em seguida, o tipo de metodologia para solucioná-lo.

Ao estudar as técnicas de modelagem organizacional, observou-se que existiam várias técnicas com carência de mecanismos que verificam se o modelo de processos de negócio está correto. Assim, a questão da pesquisa definida neste trabalho foi:

Como garantir que o modelo de processos de negócio do EKD esteja correto, livre de deadlock (travamentos), para que seja base para construção do sistema atendendo as reais necessidades da organização ou para avaliação dos processos de negócio?

2.3.1. Seleção dos Procedimentos de Pesquisa

Os procedimentos técnicos de pesquisa na busca da solução da questão da pesquisa são apresentados a seguir.

Na primeira fase da pesquisa, o objetivo foi explorar algumas técnicas de modelagem organizacional, em seguida, buscar se existem formas de verificar se o modelo está correto, livre de problemas que possam comprometer atividades baseadas nesse modelo,

como o desenvolvimento de um sistema de informação. A partir desse estudo, percebe-se que as técnicas de modelagem organizacional precisam ter uma linguagem natural e simples para que as pessoas da organização participem ativamente do processo de modelagem. O método EKD mostrou-se mais abrangente, de linguagem simples e com um conjunto importante de diretrizes para a construção do modelo. No capítulo 3 é demonstrado mais detalhadamente por que o modelo de processos de negócio do EKD foi escolhido para este trabalho. Inicialmente, o procedimento adotado foi pesquisa bibliográfica como atividade primária.

Na segunda fase da pesquisa, o objetivo foi explorar trabalhos que apresentam relação entre as redes de Petri e modelos de processos de negócio, um sub-modelo do modelo organizacional.

O potencial da área foi confirmado e iniciou-se o desenvolvimento do método de avaliação do modelo de processos de negócio. O primeiro passo foi o desenvolvimento da formalização do modelo de processos de negócio do EKD. A seguir foi feito o mapeamento matemático do modelo de processos de negócio em redes de Petri. Em seguida, a execução do procedimento *pesquisa experimental* por meio de testes, buscando confirmar e enriquecer as considerações apresentadas no mapeamento por meio de testes de mapeamento em um editor de redes de Petri. Foi desenvolvida a árvore de alcançabilidade para todos os modelos testados.

Em resumo este trabalho pode ser caracterizado pelas classificações no quadro 1.

Aspectos	Classificação
Método Científico	Hipotético-dedutivo
Ênfase	Pesquisa Aplicada
Abordagem do Problema	Pesquisa Qualitativa
Métodos de Procedimento	Pesquisa Bibliográfica Pesquisa Experimental

Quadro 1 - Caracterização do Trabalho de pesquisa

2.4. Plano Metodológico Adotado

A partir do método de trabalho e da determinação das fases que configuram a pesquisa são definidas as etapas de trabalho.

Nesta pesquisa, o esquema adotado compreende as fases: Planejamento; Pesquisa Bibliográfica; Análise e Interpretação das Informações, e Redação do Resultado. A partir da metodologia de pesquisa adotada e da determinação das fases que a configuram, foram definidas as etapas do trabalho.

As etapas definidas a cada uma das fases da pesquisa são apresentadas na figura 1. Os retângulos maiores, no fundo da figura, representam as fases da pesquisa, discutidas anteriormente. A seqüência lógica entre as fases é representada por flechas que interligam as figuras. A fase caracterizada pela pesquisa bibliográfica está representada pelo retângulo à esquerda. Essa fase é constituída por outro retângulo relacionado aos procedimentos adotados. Os retângulos menores representam os procedimentos de pesquisa, que são especificados como tarefas, por um verbo no infinitivo e um complemento.

As etapas do trabalho são discutidas a seguir:

- **Planejamento.** Nessa fase foi definida a questão da pesquisa foi selecionado o tipo de metodologia. Nesta etapa inicial compreendeu a determinação da pergunta de pesquisa, a escolha do tipo de metodologia de pesquisa, e a definição das fases da pesquisa e das etapas do trabalho, com o planejamento dos procedimentos de pesquisa e das técnicas de coletas de dados a serem executados.
- **Pesquisa Bibliográfica.** Foi realizada praticamente ao longo de todo o trabalho. A pesquisa bibliográfica abrange redes de Petri, modelagem organizacional e processos de negócio relacionados a redes de Petri. Baseado na pesquisa bibliográfica foi feita a formalização do Modelo de Processos de Negócio, o mapeamento do modelo organizacional em redes de Petri. A seguir foi desenvolvido o método de avaliação do modelo de processos de negócio.
- **Pesquisa Experimental.** Foi aplicado o método de avaliação do modelo de processos de negócio criado neste trabalho no modelo de seleção de aluno (desenvolvido neste trabalho) e em alguns MPN-EKD retirados de outros trabalhos. As árvores de alcançabilidade dos modelos são construídas. Todos os modelos mapeados em redes de Petri foram simulados na ferramenta *Petri Net Tools*

desenvolvida e implementada no Laboratório de Simulação e Controle de Sistemas Discretos da USP de São Carlos (Soares, 2001). O editor possui seis módulos. Quatro módulos estão em funcionamento: rede de Petri L/T (lugar-transição), MFG (*Mark Flow Graph*), SFC (*Sequential Flow Chart*) e PN Estocásticas. Dois módulos estão em fase de implantação: PLC, Virtual Petri Net (Redes de Petri Virtual). O editor permite os seguintes tipos de análises: árvore de alcançabilidade; matriz de incidência; limitação; vivacidade; verificação do estado final e transições e lugares invariantes.

- **Análise dos resultados.** Com a execução dos procedimentos de pesquisa e das técnicas adotados, os dados coletados na fase anterior foram analisados e interpretados, para auxiliar na identificação das considerações do método de avaliação de processos de negócio. Ao final dessa fase, fez-se uma análise e a comparação dos resultados obtidos nos testes com as considerações identificadas, no intuito de enriquecê-las.

Figura 1- Plano metodológico adotado

2.5. Limitações da Pesquisa

Os testes que são apresentados neste trabalho são feitos manualmente por não existir ainda uma ferramenta que apóie a primeira versão do método proposto.

Os modelos que serão utilizados nos testes neste trabalho foram retirados de trabalhos que utilizaram o método EKD e por isso não foram desenvolvidos de acordo com a primeira versão do método de avaliação do modelo de processos de negócio.

2.6. Considerações Finais

Neste capítulo, foi feita uma caracterização das diferentes metodologias de pesquisa e, em seguida, foi apresentada a análise de adequação e de viabilidade da adoção de uma

metodologia de pesquisa condizente aos objetivos propostos. Foi mostrado, ainda, o plano metodológico com a descrição de suas fases e de suas etapas e as limitações de pesquisa.

CAPÍTULO 3

Modelagem Organizacional

3.1. Considerações iniciais

A modelagem organizacional ajuda a entender o ambiente de negócio e isso é conhecido como valiosa atividade pela Engenharia de Requisitos porque considera aspectos abrangentes como: objetivos organizacionais, regras do negócio, restrições, aspectos de qualidade não funcionais. Em um ambiente instável, os engenheiros de software são desafiados a desenvolver sistemas de modo a atender aos requisitos das organizações modernas que passam por sucessivas mudanças. A mudança não é um processo simples, para realizá-lo, a organização deve entender, conhecer a situação atual e a situação futura que se espera alcançar e como será alcançado esse estado. Nesse sentido, neste capítulo apresenta-se a modelagem organizacional como forma de adquirir o conhecimento da estrutura organizacional e estratégica; capturar requisitos organizacionais na tentativa de melhorar a compreensão do domínio e interagir com usuários para que eles entendam o que o sistema pode fazer para melhorar o negócio.

3.2. Contexto

Muitas técnicas tradicionalmente aplicadas no desenvolvimento de software tratam de aspectos relacionados à funcionalidade do sistema, à descrição de atividades e entidades, às entradas que deverão ser transformadas e às saídas que deverão ser produzidas, porém sem considerar aspectos mais amplos como: os objetivos da organização, as regras do negócio, as restrições, os aspectos não funcionais relacionados à qualidade, a confiabilidade e a usabilidade. De acordo com Alencar (1999), essas técnicas não ajudam a buscar soluções alternativas para problemas da organização, não

adicionam valor ao negócio e, na maioria das vezes, os processos manuais são automatizados sem nenhuma modificação. Os requisitos organizacionais não devem ser considerados como uma simples descrição da funcionalidade do sistema, pois tratam do domínio no qual o sistema está inserido e das restrições que podem existir no ambiente, no sistema e no desenvolvimento, diminuindo ambigüidades e incertezas.

Bubenko (1993) faz importante observação com relação ao grande número de ferramentas de métodos CASE. Para o autor esses produtos são direcionados para a metade ou para o final do processo de desenvolvimento de software. Praticamente nenhum deles é direcionado de forma estruturada: o início do processo, os objetivos do negócio, os estágios de geração de requisitos, e não resolvem o problema de mover do domínio informal para o formal. Os métodos existentes não são designados para captura explícita e representação de forma estruturada do “conhecimento organizacional e do negócio” para ser subsequente usado na fase de projeto do sistema de informação. Não são mantidas ligações entre modelo organizacional e especificação do sistema. Assim, não pode ser explicitamente feito o gerenciamento de mudança e de evolução da organização, e nem o mapeamento de mudanças nos requisitos e nos componentes do sistema de informação.

Nesse contexto, a modelagem organizacional facilita a compreensão do ambiente empresarial e é reconhecida como uma atividade valiosa pela Engenharia de Requisitos. O modelo organizacional é uma representação da estrutura, das atividades, dos processos, das informações, dos recursos, do pessoal, do comportamento, dos objetivos e das restrições das empresas comerciais, governamentais ou de outra natureza. Esse modelo ajuda a compreender as complexas interações entre as organizações e as pessoas.

3.3. Abordagens de Modelagem Organizacional

Muitas técnicas de Modelagem de Organizacional são propostas, algumas com o foco principal nos aspectos sociais como em Dobson (1994) que descreve os objetivos, política e estrutura da organização. Na linha de Bubenko (1994), Yu (1993) e Rolland *et al.* (2000), é realizada a modelagem organizacional com múltiplas visões com análise de metas e objetivos da organização. A organização, segundo esses autores, é representada por meio de modelos que facilitam a realização de especificações de requisitos mais próximas à realidade da organização. Os modelos de requisitos existentes descrevem o

ambiente organizacional em termos de entidades e atividades, sem se importarem com situações em que os usuários poderão tomar diferentes decisões. Esses modelos têm como objetivo a descrição de sistemas técnicos em vez de fornecer descrições mais ricas sobre as organizações sócio-humanas.

As informações capturadas nos modelos existentes, como Diagrama de Fluxo de Dados (DFD) e Diagrama Entidade e Relacionamento (DER), não são suficientes, uma vez que esses modelos descrevem apenas entidades, funções, fluxo de dados e estados do sistema, não expressando as razões envolvidas no processo, ou seja, o porquê de fazer uma determinada ação ou o porquê de tomar uma decisão (alternativas para o “como fazer”). Assim, faz-se necessário uma ontologia mais rica que facilite os esforços da Engenharia de Requisitos para obter uma melhor compreensão sobre os relacionamentos da organização entre os vários atores do sistema e entender as razões envolvidas nos processos de decisões. Algumas técnicas de modelagem organizacional são:

- 1 – A técnica ORDIT (*Organizational Requirements Definition of Information Technology Systems*), de acordo com Dobson (apud Alencar, 1999), serve para ajudar os participantes das organizações a definir técnicas alternativas e o futuro organizacional e, conseqüentemente, o papel da tecnologia da informação, fornecendo um processo sistemático e aproveitável que seja capaz de suportar gerações de requisitos organizacionais e fornecer métodos e ferramentas associadas que suportem o processo.
- 2 – A técnica de modelagem de Furlan (1997) tem como princípio conhecer a missão da organização. A missão é o motivo pelo qual a empresa foi criada, ou seja, a identidade da organização, sendo um texto curto e objetivo. O próximo passo é definir os objetivos executivos ou objetivos da organização que são os alicerces para a missão, devem portanto, ser totalmente compatíveis com o que estabelece a missão. Os objetivos estarão melhor definidos conforme o desenvolvimento da empresa. Depois, serão definidos os objetivos estratégicos que estão relacionados com as áreas funcionais, com a finalidade de alcançar os objetivos executivos e os fatores chaves de sucesso. Para alcançar os fatores chaves de sucesso são desenvolvidas estratégias, que serão o diferencial da empresa no mercado. Os planos de ação representam a concretização das estratégias.
- 3 – A técnica F3 de Bubenko (1993), destaca áreas de conhecimento da organização. É constituída por cinco modelos elaborados a partir de objetivos (Modelo de Objetivos

–MO), atores (Modelo de Atores – MA), atividades e uso (Modelo de Atividades e Uso – MAU), conceitos (Modelo de Conceitos – MC) e requisitos (Modelo de Requisitos do Sistema de Informação – MRSI).

4 – A técnica i^* de Yu (apud Alencar, 1999), é composta por dois modelos: o Modelo de Dependências Estratégicas (SD) e o Modelo de Razões Estratégicas (SR). O Modelo de Dependências Estratégicas (SD) descreve as relações de dependências externas entre os atores da organização e o Modelo de Razões Estratégicas (SR) descreve interesses e conceitos dos participantes e as direções que podem seguir. As semânticas dos relacionamentos, no Modelo de Dependências Estratégicas (SD), são caracterizadas como algumas supostas características internas intencionais do agente, algumas das quais são explicitamente modeladas no segundo tipo de modelo previsto na técnica i^* , Modelo de Razões Estratégicas (SR). Na estrutura interna intencional de um agente, em que todas as variáveis livres nas fórmulas são entendidas como universalmente quantificadas, inclui-se, pelo menos, os três componentes seguintes: (a) U_x – um conjunto de rotinas; (b) H_x – um conjunto de regras meio-fim; e (c) E_x – um conjunto de elementos primitivamente manipuláveis;

O método EKD (*Enterprise Knowledge Development*), é uma evolução da técnica F3. Esse método é orientado a objetivos e trata explicitamente em um modelo particular as regras do negócio. O EKD é composto de seis modelos: Modelo de Objetivos (MO), Modelo de Regras do Negócio (MRN), Modelo de Conceitos (MC), Modelo de Processos de Negócio (MPN), Modelo de Atores e Recursos (MAR) e Modelo de Requisitos e componentes técnicos (MRCT). A proposta de usar o EKD é prover uma descrição clara de:

- Como a organização funciona atualmente.
- Quais são os requisitos e as razões para a mudança.
- Quais alternativas deveriam ser criadas para encontrar esses requisitos.
- Quais são os critérios e argumentos para avaliação dessas alternativas.

3.4. Considerações sobre as técnicas

A técnica de Furlan (1997) e a técnica ORDIT não desenvolvem modelos com múltiplas visões, não consideram as regras do negócio, não apresentam modelos de processos o

que dificulta o trabalho de captura dos requisitos organizacionais. A técnica de Furlan (1997) se concentra na definição da missão, objetivos, estratégias e plano de ação. Não considera especificação dos requisitos organizacionais, e os atores envolvidos. A técnica ORDIT trata a responsabilidade das pessoas envolvidas no trabalho. As práticas de trabalho são descritas como responsabilidades e relacionamentos em vez de atividades ou processos. Não trata objetivos organizacionais.

O principal enfoque da técnica *i** são os atores, concentrando-se na análise das implicações sob o ponto de vista dos relacionamentos de dependência entre os atores de uma organização. A técnica F3 e o método EKD tem como enfoque principal os objetivos, dos quais derivam-se as atividades e os atores. O objetivo, na técnica *i**, está vinculado a uma dependência entre dois atores, ou seja, o que um espera obter do outro. Em F3 e no método EKD, o objetivo está vinculado à organização e não a um ator em específico ou a um relacionamento entre dois atores. Na técnica *i**, por meio dos atores, decidem-se os domínios dos objetivos; em F3 e no método EKD partindo-se dos objetivos surgem os domínios.

Um fator muito importante para a Engenharia de Requisitos é a compreensão do modelo por todos (desenvolvedores, clientes e usuários), uma vez que o problema esteja claro é possível validar requisitos. Alencar *et al.* (1999) explicam que o entendimento dos modelos recai na questão de pontos de vista. Por parte dos profissionais técnicos, aptos a lidarem com modelos gráficos do tipo DFD, ou DER, o entendimento dos modelos propostos pelas duas técnicas não se diferenciariam quanto à dificuldade ou facilidade de compreensão. A técnica *i** trabalha com um pouco de rigor matemático, o que não é muito atraente para alguns desenvolvedores de software. Com um esforço razoável esses profissionais compreenderiam o significado dos modelos em ambas as técnicas. Contudo, por parte de outros profissionais não especificamente ligados à área de desenvolvimento de software, os modelos da técnica *i** seriam de difícil compreensão, ao passo que os modelos de F3 e do EKD seriam mais fáceis de serem entendidos e analisados. De acordo com Bubenko & Wangler (1993), para uma pessoa do negócio ser capaz de ler e validar os modelos, é necessário que a linguagem seja fácil de entender.

Em F3, o ponto principal está no conhecimento ou identificação dos objetivos da organização, retratados em seu modelo de Objetivos (MO), do qual resultam os modelos de Atividades e Uso (MAU) e o de Atores (MA). A técnica *i** estabelece de imediato a

noção de atores e seus relacionamentos, partindo do modelo de Relacionamentos Estratégicos (SD) e derivando o modelo de razões estratégicas (SR).

De acordo com Berztiss & Bubenko (1995), a experiência tem mostrado que o uso de uma abordagem dirigida a modelo de objetivos proporciona um melhor entendimento do domínio para os tomadores de decisões, definidores de requisitos, desenvolvedores de software e usuários finais, além do valor econômico de melhorar a comunicação e entendimento, minimizando retrabalho nos estágios finais do processo de desenvolvimento do sistema de informação. Um bom sistema de informação pode ser uma vantagem competitiva do negócio, mas se o sistema não atende as necessidades do negócio, este pode ter seu desempenho prejudicado. Esse é o motivo do início de uma troca de paradigma da Engenharia de Sistemas orientada à tecnologia para estruturas que valorizam a modelagem de regras do negócio (Bubenko & Wangler, 1993).

As regras do negócio propostas em F3 no Modelo de Objetivos (MO) e no método EKD no Modelo de Regras do Negócio (MRN), não são tratadas especificamente na técnica i*. Como afirmado anteriormente, a importância de regras do negócio na modelagem organizacional e especificação de requisitos tem sido discutida por diversos autores (Bubenko (1993) Bubenko & Wangler (1993), Stergiou & Johnson (1998), Rosca *et al.* (1997), Gottesdiener (1999), Herbst (1996), Kilov & Simmonds (1997), Dallavalle & Cazarini (2000), Pádua (2001) e Leite & Leonardi (1998)).

A preocupação inicial de Bubenko (1993) era considerar como os usuários e clientes do sistema olhavam para o sistema e como suas necessidades e seus conhecimentos eram coletados e registrados na especificação de requisitos. Na experiência do SISU – Swedish Institute for Systems Development (Instituto Suíço de Desenvolvimento de Software) a Modelagem de Negócio foi um componente essencial de formulação de propostas para o projeto ESPRIT F3 que objetiva formalizar aspectos confusos da organização. Em SISU foi desenvolvido um número de modelos especiais para discutir e representar a parte do "Por Que" dos requisitos do sistema de informação. Essa parte é chamada de Modelagem de Negócio. Muitos efeitos positivos foram registrados, tais como: melhora do entendimento do domínio da aplicação, aumento da comunicação entre projetista e usuários, e melhora na base para projetar sistemas de informação. A Modelagem de Negócio é bastante apreciada por pessoas que não são ligadas à computação, sendo mais difícil "vender" a idéia para especialistas em computadores e softwares.

É de fundamental importância a conexão entre o desenvolvimento de objetivos, estratégias do negócio e o desenvolvimento de sistemas de informação. O alto custo de manutenção é um fator importante para incentivar o entendimento do negócio e derivar os requisitos do sistema desse entendimento. Bubenko (1993) esclarece que a metodologia F3 não é utilizada apenas para capturar requisitos, mas um método para estabelecimento de uma base mais sólida e completa de projeto e implementação de sistemas considerando a completude, facilidade de mudança e de gerenciamento. De acordo com a abordagem F3, um Modelo Organizacional é uma descrição parcial de suposições básicas, requisitos e conhecimento do problema do domínio necessários para projetar e implementar um sistema de informação. O autor cita os seguintes itens que devem estar em uma especificação de requisitos:

- Por que o sistema é necessário? Quais são as metas do ambiente da aplicação? Quais são as prioridades das metas? Como as metas se relacionam?
- Quais conceitos ou temas da organização, incluindo seus relacionamentos com objetivos, atividades, processos e atores? Como são definidos? Quais regras do negócio e que restrições monitoram esses objetivos?
- Quais atividades e processos de negócio existem ou deveriam existir, para gerenciar a organização de acordo com os objetivos? Como são realizados os processos de negócio? Que informações são necessárias?
- Quem está realizando quais tarefas e processos?
- Que requisitos são gerados por esses processos?

O método EKD, como já afirmado anteriormente, é uma evolução da técnica F3. O método é orientado a objetivos e trata também os atores envolvidos nos processos, nas regras e nos objetivos da organização.

O quadro 2 e o quadro 3 ilustram o estudo comparativo entre as técnicas apresentadas. É possível observar que o desenvolvimento do modelo organizacional utilizando o método EKD proporciona o gerenciamento de mudanças, captura das melhores práticas do negócio, gerenciamento das regras do negócio, gerenciamento do conhecimento organizacional. O método apresenta: um conjunto de diretrizes que orienta todo o processo de modelagem, um conjunto de questões que apóia o desenvolvimento de todos os modelos e um outro conjunto de questões para apoio na verificação das

ligações entre componentes de todos os modelos depois de desenvolvidos. O método é utilizado para a especificação de requisitos.

De acordo com Kirikova (2000), a família de modelo do EKD é destinada para responder as questões: o que, como, onde, quem, quando e por quê. Essa estrutura serve como um esquema de classificação conveniente ou "tabela periódica" para entidades de informação. Como elementos químicos, essas entidades podem ser combinadas de infinitas formas para produzir o sistema de informação de interesse da organização. Em outro ponto de vista, é possível ver essa estrutura como uma família de muitos modelos inter-relacionados, em que relacionamentos entre elementos arbitrários pertencentes a submodelos são permitidos.

A conexão entre desenvolvimento de sistemas de informação e objetivos e estratégias do negócio é mais presente no método EKD, uma vez que ele dispõe do Modelo de Regras do Negócio ligado aos processos de negócio, atores e outros componentes do modelo organizacional. O método EKD representa passos importantes para um novo paradigma de desenvolvimento de software, que nos primeiros estágios de desenvolvimento concentra-se na organização e em questões do negócio em vez de em mecanismos de implementação de sistemas de informação. Para Kirikova (2000), essa talvez seja a teoria mais rica em uso.

Neste sentido, a abordagem que será utilizada neste trabalho é o EKD – Enterprise Knowledge Development, porque entre todas as abordagens encontradas na literatura, ele foi o único a tratar gerenciamento de mudanças, regras do negócio explicitamente em um modelo e a disponibilizar um conjunto de diretrizes para sua aplicação. O desenvolvimento de cada um dos sub-modelos é apoiado por um conjunto de questões. O método é independente de uma ferramenta, dessa forma os modelos podem ser desenvolvidos em qualquer editor de texto ou manualmente.

Técnica ou método de modelagem	Orientada	Principais características
ORDIT	Atores	<ul style="list-style-type: none">– Trata a responsabilidade das pessoas envolvidas no trabalho.– Focaliza como os componentes humanos são organizados no trabalho.– Não desenvolve modelos com múltiplas visões.– Não considera regras do negócio.– As práticas de trabalho são descritas como

Técnica ou método de modelagem	Orientada	Principais características
		<p>responsabilidades e relacionamentos em vez de atividades ou processos.</p> <ul style="list-style-type: none"> – Não trata objetivos organizacionais.
Furlan (1997)	Missão da empresa	<ul style="list-style-type: none"> – É constituída da definição da missão, objetivos executivos, objetivos estratégicos, fatores chaves de sucesso, estratégia, planos de ação. – Não desenvolve modelos com múltiplas visões. – Não considera regras do negócio. – Não tem um modelo dos processos de negócio. – Não trata especificação dos requisitos organizacionais. – Não trata o desenvolvimento de Sistemas de informação. – Não trata os Atores envolvidos
F3	Objetivos	<ul style="list-style-type: none"> – É constituída por cinco modelos elaborados a partir dos objetivos. – Relaciona objetivos, atividades, processos e atores. Destaca áreas da organização. – Não tem um modelo específico de regras do negócio. Trata as regras dentro do modelo de objetivos.
I*	Atores	<ul style="list-style-type: none"> – Trata o relacionamento de dependência de atores. – É composta por dois modelos: o Modelo de Dependências Estratégicas (SD) e o Modelo de Razões Estratégicas (SR). – Descreve as relações de dependências externas entre os atores da organização. – Descreve interesses e conceitos dos participantes e as direções que podem seguir. – Não consideram regras do negócio. – Não tem um modelo dos processos de negócio.
EKD	Objetivos	<ul style="list-style-type: none"> – É uma evolução de F3. – É composto de seis modelos: de objetivos, regras do negócio, de conceitos, processos de negócio, atores e recursos e de requisitos e componentes técnicos. – Possui um conjunto de questões que apóia o desenvolvimento de todos os modelos. – Oferece um conjunto de questões para apoio na verificação das ligações entre componentes de todos os modelos. – Captura as melhores práticas do negócio. – Gerenciamento de mudanças. – Gerenciamento das regras do negócio. – Pode ser utilizada para especificação de requisitos.

Técnica ou método de modelagem	Orientada	Principais características
		<ul style="list-style-type: none"> – É orientada para a aprendizagem organizacional. – Trata o gerenciamento do conhecimento organizacional. – Apresenta um conjunto de diretrizes que orienta todo o processo de modelagem.

Quadro 2 - Comparação entre as técnicasFonte: Pádua *et al.* 2004.

Características	Furlan	ORDIT	F3	i*	EKD
Descreve interesses e conceitos dos participantes e as direções que podem seguir.				X	
As práticas de trabalho são descritas como responsabilidades e relacionamentos em vez de atividades ou processos.		X			
As práticas de trabalho são descritas como processos, responsabilidades, relacionamentos com regras do negócio e recursos.			X		X
Trata como os componentes humanos são organizados no trabalho.		X	X	X	X
Trata a responsabilidade das pessoas envolvidas no trabalho.		X	X	X	X
Relaciona objetivos, atividades, processos e atores. Destaca áreas da organização.			X		X
É constituída por modelos elaborados a partir dos atores.		X		X	
Descreve as relações de dependências externas entre os atores da organização.			X	X	X
É constituída por modelos elaborados a partir dos objetivos.			X		X
Trata o relacionamento de dependência de atores.				X	X
Orientada a missão.	X				
Orientada a objetivos.			X		X
Orientada a atores.		X		X	
Tem um modelo específico de regras do negócio					X
Trata da definição da missão, objetivos executivos, objetivos estratégicos, fatores chaves de sucesso, estratégia, planos de ação.	X				
Considera regras do negócio.			X		X
Desenvolve modelos com múltiplas visões.			X	X	X
Tem um modelo dos processos de negócio.			X		X
Trata a especificação dos requisitos organizacionais.		X	X	X	X
Trata o desenvolvimento de Sistemas de informação			X		X
Trata os atores (pessoas) envolvidos		X	X	X	X
Apresenta um conjunto de diretrizes que orienta todo o processo de modelagem.					X
Trata o gerenciamento do conhecimento organizacional.					X
Orientada para a aprendizagem organizacional.					X

Características	Furlan	ORDIT	F3	i*	EKD
Trata o gerenciamento de mudanças					X
Trata o gerenciamento das regras do negócio.					X
Captura as melhores práticas do negócio.					X
Oferece um conjunto de questões para apoio na verificação das ligações entre componentes de todos os modelos.					X
Possui um conjunto de questões que apóia o desenvolvimento de todos os modelos.					X

Quadro 3 - Características das técnicasFonte: Pádua *et al.* 2004.

3.5.EKD

O trabalho que originou o EKD iniciou-se nos anos 1980 pelo projeto Plandata e foi refinado pelo SISU (Swedish Institute for Systems Development – Instituto Sueco para o Desenvolvimento de Software) no final dos anos 1980. O Instituto Sueco de Desenvolvimento de Software foi fundado em 1984 e realizou mais de 100 projetos de análise de sistemas e negócio, nos quais a metodologia de modelagem organizacional foi usada. A grande contribuição foi considerar componentes organizacionais de uma especificação, por exemplo, os objetivos (intenções) de um negócio, além dos tipos de componentes do modelo tradicional, como entidade, relacionamentos e processos. O uso dessa abordagem em muitas aplicações diferentes durante os últimos dez anos mostrou que a razão do sucesso não era apenas o Modelo Organizacional, mas também o gerenciamento apropriado do processo do negócio e Engenharia de Requisitos. A idéia inicial do Modelo do Negócio do SISU foi estendida para o Modelo Organizacional e mais tarde desenvolvida no projeto ESPRIT F3 (*From Fuzzy to Formal*). A modelagem organizacional F3 foi então elaborada pelo projeto ESPRIT ELKD e depois foi aplicada no projeto ESPRIT ELEKTRA (*Electrical Enterprise Knowledge for Transforming Applications*) (Bubenko *et al.*, 1998). O projeto ELEKTRA concentra-se principalmente a aplicação do método EKD para problemas de gerenciamento de mudanças dentro de organizações da Grécia e Suécia, gerando um conjunto de práticas genéricas de forma a aplicá-las em outras companhias. Segundo Rolland *et al.* (2000), a pesquisa do projeto ELEKTRA objetiva a criação de uma base de conhecimento para o gerenciamento de mudança. Um dos objetivos do projeto era: capturar as melhores práticas do negócio para reusá-las em situações similares em outras companhias de eletricidade.

3.5.1. Conhecimento Organizacional

As descrições estruturadas tornam possível ver como diferentes partes de uma organização interagem. Pode-se traçar descrições e verificar quais processos estão relacionados, e como contribuem para um particular objetivo. Inversamente, é possível verificar se existem objetivos para os quais nenhum processo está contribuindo. Percebe-se que as descrições estruturadas permitem a realização de diferentes tipos de análises e discussões sobre o que foi dito e escrito sobre a organização. Esses modelos podem ser efetivamente usados para representar e discutir como projetar e melhorar processos organizacionais e como desenvolver estratégias futuras. Entretanto, os modelos são limitados, uma vez que eles apenas apresentam percepções subjetivas de como as coisas são.

Segundo Bubenko *et al.* (1998), um grande benefício da modelagem é o efeito nos participantes. Em um projeto bem sucedido, os efeitos podem ser:

- Melhora no entendimento das partes essenciais da organização.
- Encontro de soluções para problemas práticos.
- Consenso sobre questões que no início eram difíceis de serem definidas.

Esses modelos são importantes para melhorar a comunicação e a aprendizagem organizacional (quadro 4).

Benefícios
Entender melhor o negócio.
Facilitar a aprendizagem e comunicação organizacional sobre questões essenciais.
Ajudar entender e promover as capacidades e processos da organização.
Melhorar a comunicação entre o sistema de informação <i>stakeholders</i> e desenvolvedores.
Desenvolver uma descrição estruturada do negócio para analistas da organização discutirem.
Auxiliar os desenvolvedores de sistemas de informação e <i>stakeholders</i> na determinação dos requisitos e objetivos do sistema.
Chegar a uma descrição dos objetivos da organização, entidades, processos, requisitos, que seja mais consistente e mais completo que a tradicional abordagem baseada em textos.

Benefícios
<p>Chegar a um documento, chamado repositório de conhecimento, que pode ser usado para:</p> <ul style="list-style-type: none">▪ Raciocínio sobre o negócio.▪ Discutir mudanças no negócio e componentes do sistema de informação.▪ Traçar a cadeia de componentes e decisões que leva a várias interpretações de decisões e componentes do sistema de informação.

Quadro 4 - Benefícios do EKD

Adaptada do texto de Bubenko *et al.* (1998)

A necessidade de reengenharia e de gerenciamento de mudanças está crescendo rapidamente, tornando-se fatores chaves de sobrevivência e competitividade das companhias. Nesse caso a organização precisa desenvolver uma disciplina que organize todo o conhecimento requerido para identificar a necessidade de mudança na organização e para cumprir essas mudanças de maneira conveniente e profissional (Bernus & Nemes, 1997). Segundo Kirikova (2000), o EKD pode ser usado em situações diferentes e com propósitos diferentes, como nas seguintes situações:

- Na Engenharia de Requisitos para definição e especificação de requisitos.
- Na análise do negócio para detecção do problema.
- Na reengenharia de processos de negócio para definição de novos sistemas de negócio.
- No gerenciamento de conhecimento organizacional ou aprendizagem organizacional para formar a base de propagação e ampliação de conhecimento.

Para tanto são necessárias descrições que representem e comuniquem percepções e idéias. Essas descrições são apresentadas por meio do modelo organizacional. O conhecimento da organização é estruturado por esse modelo que, sendo suficientemente detalhado e não-ambíguo, torna-se uma poderosa ferramenta para o entendimento ou desenvolvimento da organização, permitindo discussão entre objetos visuais e tangíveis que são o centro da atenção coletiva de um grupo de pessoas (Bubenko *et al.*, 1998). A tão conhecida frase "conhecimento é poder" tem sido aplicada nas organizações bem sucedidas. Gerenciar conhecimento tornou-se muito importante atualmente, mas não é uma tarefa simples (Nissen, 1999).

Para Bubenko *et al.* (1998), o modelo EKD não mostrará uma exata reflexão do mundo real. Ele é apenas uma coleção de percepções do mundo real, refletindo as estruturas de referências e experiências dos contribuidores. Um modelo de boa qualidade é baseado na discussão explícita dos participantes e nos relacionamentos entre elementos de diferentes submodelos. Para tanto, as discussões deveriam concentrar-se na organização por diferentes pontos de vistas, envolvendo participantes com conhecimentos diferentes. O modelo fornece, de forma natural, uma possibilidade para os participantes entrarem em questões e fenômenos relacionados à sua parte do negócio e ver o impacto de suas decisões ou requisitos de todos os processos da organização. De acordo com Pádua (2001), os modelos proporcionam benefícios para a cultura e aprendizado organizacional. Os participantes devem explicitamente contribuir com seus conhecimentos do domínio, suas habilidades e experiências. Eles devem ser abertos, construtivos, ativamente participativos. É muito importante que eles saibam ouvir, respeitar e responder aos outros, e que tentem encontrar e esclarecer relacionamentos e aspectos escondidos. Apenas dessa forma, os efeitos de sinergia do grupo poderiam ser alcançados. O desenvolvimento do conhecimento organizacional é bastante dependente dos participantes e não dos facilitadores. Facilitador é a pessoa que lidera e adverte durante as sessões de modelagem e apóia cada um na aquisição de conhecimento e de idéias do grupo de aplicação.

De acordo com Bubenko *et al.* (1998), o conteúdo básico da estrutura EKD inclui: um conjunto de técnicas de descrição, a participação de *stakeholders* e um conjunto de diretrizes para o trabalho. O termo *stakeholder* foi introduzido para nomear para todos os envolvidos no projeto, diretamente ou indiretamente, ou que tenha interesse no resultado do projeto. O conjunto de técnicas de descrições fornece um conjunto de modelos que é usado para descrever o sistema a ser analisado ou construído e a organização na qual ele será operado. Esse conjunto de técnicas é usado pelos desenvolvedores do sistema. As técnicas de descrições, sozinhas, não terão muito valor, sem o envolvimento direto dos atuais clientes, usuários finais, gerentes, proprietários, entre outros interessados. Um fator crítico de sucesso de um projeto que inclui a aplicação do EKD na construção de um sistema de informação ou reestruturação da organização é a participação e o envolvimento dos *stakeholders*.

3.5.2. Componentes do EKD

Os componentes do EKD são modelos conceituais que examinam a organização e seus requisitos de um número de perspectivas inter-relacionadas. Esses modelos são abstrações do mundo físico. Para uma dada organização, esses modelos irão constituir coletivamente o Modelo Organizacional. Alguns desses modelos podem conter informações relevantes que apontam a necessidade de avaliação de alternativas de situações operacionais. Tais informações incluem critério para avaliação, escolhas disponíveis, parâmetro de medidas e argumentos registrados a favor e contra escolhas (Rolland *et al.*, 2000).

Durante o estágio de desenvolvimento, o modelo organizacional pode ser um meio de entendimento e comunicação entre os participantes ou entre participantes e outros *stakeholders* do processo EKD, tornando-se um ponto de referência comum, cruzando muitas áreas diferentes a fim de que sua posse não seja confinada a aplicações específicas ou grupos particulares. Além disso, deve ser independente de qualquer tecnologia, podendo ser implementado por diversas tecnologias. O modelo precisará ser modificado apenas quando a situação e o contexto nos quais a organização existe, mudar. O modelo pode ser usado como um meio de avaliação das opções, no qual os custos de cada opção potencial podem ser avaliados tão bem quanto mais clara estiver a documentação de todos os aspectos intangíveis (Loucopoulos *et al.*, 1998).

De acordo com Rolland *et al.* (2000), o EKD apóia a construção de modelos diferentes, representando o estado inicial da organização e o futuro estado, além da expressão de estratégias alternativas para mudanças, a avaliação dessas estratégias, e outras atividades como *brainstorming*. A "regra de parada" para o desenvolvimento do Modelo Organizacional é um conjunto de modelos descritos em detalhes suficientes a fim de que possa ser usado como uma base para implementar as sugestões nos modelos, ou para resolução de problemas por outros meios.

3.5.3. Como usar o EKD

A abordagem EKD, de acordo com Rolland *et al.* (2000), envolverá estrategistas, gerentes táticos e funcionários do nível operacional, que juntamente com o facilitador e técnicos, familiarizados com EKD, iniciarão o processo de:

- 1 – Diagnóstico: modelar a situação corrente e os requisitos de mudanças.

- 2 – Entendimento: interpretar, entender, raciocinar, deliberar e discutir o estado corrente e futuro da empresa.
- 3 – Projeto: discutir e modelar as situações alternativas futuras e os cenários.

O modelo resultante estará disponível para os tomadores de decisões atuarem sobre as estratégias futuras, táticas e objetivos da empresa. O Modelo Organizacional contém vários submodelos inter-relacionados. Cada um representa algum aspecto da organização. De acordo com Bubenko *et al.* (1998), os tipos de submodelos e as questões que eles abordam são:

1 – Modelo de Objetivos (MO). Concentrado na descrição de idéias da organização. Descreve o que a organização e os empregados querem alcançar ou evitar, e quando. Usualmente esclarece questões como:

- Para onde deveria ser movida a organização.
- Quais os objetivos mais importantes e as prioridades desses objetivos.
- Como cada objetivo é relacionado aos outros e quais problemas dificultam a realização das metas.

2 – Modelo de Regras do Negócio (MRN). Usado para definir e manter explicitamente regras do negócio formuladas e consistentes com o Modelo de Objetivos. Regras do Negócio podem ser vistas como operacionalização ou como limites dos objetivos. O Modelo de Regras do Negócio geralmente esclarece questões como:

- Quais regras afetam os objetivos da organização.
- Quais são as políticas declaradas.
- Como cada regra do negócio é relacionada com os objetivos.
- Como os objetivos serão apoiados pelas regras.

3 – Modelo de Conceitos (MC). Usado estritamente para definir "coisas" e "fenômenos" relacionados a outros modelos. Representa entidades organizacionais, atributos e relacionamentos. Entidades são usadas para definir mais estritamente expressões do Modelo de Objetivos tanto quanto o conteúdo do conjunto de informação do Modelo de Processos de Negócio. O Modelo de Conceitos usualmente esclarece questões como:

- Quais entidades ou conceitos são reconhecidos na organização (incluindo seus relacionamentos com objetivos, atividades e processos, e atores).
- Como as entidades são definidas.
- Quais regras do negócio e restrições monitoram esses objetos e conceitos.

4 – Modelo de Processos de Negócio (MPN). Usado para definir processos organizacionais, e a forma pela qual eles interagem e manuseiam a informação e os materiais. Um processo de negócio deve consumir as entradas (informação e/ou material) e produzir uma saída (informação e/ou material). Em geral o MPN é similar aos tradicionais modelos de diagramas de fluxo de dados (DFD). O Modelo de Processos de Negócio esclarece questões como:

- Quais atividades e processos do negócio são reconhecidos na organização (ou deveriam ser) para gerenciar a organização em concordância com as metas.
- Como os processos de negócio e tarefas deveriam ser realizados (*workflows*, transição de estado, ou modelo de processos), e quais as informações necessárias.

5 – Modelo de Atores e Recursos (MAR). Usado para descrever como diferentes atores e recursos se relacionam, e como eles são relacionados a componentes do Modelo de Objetivos e a componentes do Modelo do Processo de Negócio. Em Pádua *et al.* (2003a) é apresentado o potencial e a importância do MAR para o Modelo Organizacional. Por exemplo: um ator pode ser responsável por um particular processo no MPN ou um ator pode buscar um particular objetivo no MO. O Modelo de Atores e Recursos esclarece questões como:

- Quem está ou deveria estar realizando quais processos e tarefas.
- Como estão as estruturas de informação e de responsabilidade entre os atores definidos.

6 – Modelo de Requisitos e Componentes Técnicos (MRCT). Usado quando a proposta do EKD é ajudar a definir os requisitos para o desenvolvimento de um sistema de informação. Esse modelo direciona para o sistema técnico que é necessário para apoiar os objetivos, processos e atores da organização. Inicialmente é necessário desenvolver um conjunto de requisitos de alto nível ou objetivos para o sistema de informação como um todo. Baseado nesses requisitos, o sistema de

informação é estruturado em um número de subsistemas, ou componentes técnicos. O MCRT é uma tentativa inicial de se definir toda a estrutura e propriedades do sistema de informação para apoiar as atividades do negócio, como definido no MPN. O Modelo de Requisitos Componentes Técnicos usualmente esclarece questões como:

- Quais requisitos são gerados pelos processos do negócio.
- Qual o potencial da tecnologia da informação para melhoria do processo.

De acordo com Pádua (2001), cada um desses submodelos inclui um número de componentes que descreve diferentes aspectos da organização. Por exemplo, o Modelo de Objetivos contém objetivos do negócio, problemas do negócio que são divididos em tratamento, fraquezas, causas, oportunidades do negócio e restrições. Os componentes dos submodelos são relacionados entre si (relacionamento intramodelo), e também com componentes de outros submodelos (relacionamento inter-modelos).

A habilidade de traçar decisões, componentes e outros aspectos em todas as partes da organização é dependente do uso e do entendimento desses relacionamentos. Quando se desenvolve um modelo organizacional completo, esses relacionamentos entre componentes de diferentes submodelos executam tarefa essencial. Por exemplo, declarações do modelo de objetivos permitem que diferentes conceitos sejam definidos no Modelo de Conceitos. Da mesma forma, objetivos do Modelo de Objetivos motivam processos particulares no Modelo de Processos de Negócio. Os processos são necessários para atingir os objetivos declarados. Uma ligação (*link*) é definida entre um objetivo e um processo. Ligações entre modelos tornam o modelo transparente, mostrando por que certos processos e requisitos do sistema de informação têm sido introduzidos (Loucopoulos *et al.*, 1998).

De acordo com Bubenko *et al.*(1998), existem limitações na forma em que os submodelos e seus relacionamentos podem ser povoados. Esses são controlados por algumas regras de consistência estáticas e dinâmicas que, por sua vez, controlam seus estados de transição permissíveis. Os relacionamentos entre modelos são importantes por possibilitarem a análise e comparação dos elementos organizacionais.

Os componentes do modelo organizacional têm representações gráficas, como, por exemplo, caixas retangulares, elipses, e outras formas. O relacionamento entre esses componentes é representado com ligações (*links*). Existem dois textos associados a cada

componente: *short name* e *long name*. O *short name* é usado para numerar e identificar os componentes do modelo, deve ser único, para servir de referência. O *long name* é usado para expressar e armazenar a descrição do componente.

Na abordagem EKD existe o cuidado de relacionar os componentes do Modelo de Objetivos por meio de ligações semânticas, das quais os principais tipos são (Bubenko *et al.*, 1998):

- Apoio: usado para refinar ou decompor objetivos e outros componentes.
- Impedimento: usado para mostrar influências negativas entre componentes do Modelo de Objetivos.
- Conflito: usado para definir situações em que ao atingir um objetivo, poderá haver um conflito com outro objetivo.

3.5.4. Refinamento e operacionalização dos objetivos

De acordo com Bubenko *et al.* (1998), a proposta de operacionalização de objetivos é detalhar como o alto nível de objetivos pode ser satisfeito. A operacionalização do Modelo de Objetivos encoraja o desenvolvimento da rede de objetivos. As principais atividades da operacionalização são o refinamento dos objetivos e o gerenciamento de conflitos. Durante o refinamento dos objetivos, novos objetivos são gerados por meio do detalhamento de alto nível. Nesse sentido, um objetivo de alto nível é refinado em um ou mais sub-objetivos, que podem tornar a ser refinados em novos sub-objetivos. O resultado desses refinamentos sucessivos é uma estrutura hierárquica multinível, iniciando-se de um objetivo de alto-nível vago, até objetivos operacionais específicos. É possível usar relacionamento AND/OR nas estruturas para refinar objetivos em muitas combinações alternativas de sub-objetivos, ou um sub-objetivo que pode ser realizado por muitas alternativas de modelo de projeto. O gerenciamento de conflitos consiste de um número de atividades:

- Detecção de conflitos: concentra-se identificação de conflitos entre objetivos. Pode ser difícil relacionar novos objetivos com os objetivos existentes e determinar o efeito daqueles para estes. Para conseguir encontrar conflitos deve-se procurar exaustivamente no modelo de objetivo e comparar o novo objetivo para cada objetivo existente. Uma forma razoável de procurar por potenciais conflitos é usar

os conflitos dos objetivos de alto nível que já foram identificados durante o estágio de aquisição.

- Classificação de conflitos: concentra-se na identificação do tipo de conflito detectado. Conflito final ocorre quando dois objetivos contraditórios são desejados. Conflito de recursos (*means*) ocorre quando os atores relacionados a objetivos idênticos querem usar o mesmo recurso. A classificação de conflitos pode ser usada por métodos de gerenciamento de conflitos para reagir de modo devido.

Muito frequentemente, de acordo com Loucopoulos *et al.* (1998), os objetivos de alto nível, os problemas, regras do negócio, entre outros, adquiridos no estágio de aquisição, contêm um número de requisitos informais e imprecisos; é recomendado que a saída do Modelo de Objetivos inicial seja estruturada no estágio inicial. Essa tarefa envolve:

1 – Classificação de Objetivos: classificá-los em uma tabela matriz é aconselhável para melhorar a compreensão e entendimento de inúmeros objetivos, na qual eles possam ser categorizados de acordo com origem, *stakeholder*, função, domínio etc. Isso permitirá a comparação, análise e, talvez, a descoberta da necessidade de mais discussões baseadas na análise dos padrões dos objetivos.

2 – Priorização de Objetivos: possibilita a resolução dos conflitos entre objetivos. Um alto nível de objetivos age como uma restrição em um objetivo de baixo nível.

3 – Correlação de Objetivos: é compreendida como interação positiva e negativa entre objetivos. Em geral, colaboração de objetivos é desejável, desde que a colaboração entre dois objetivos signifique que a satisfação de um apoiará a satisfação de outro; mostrando também um consenso entre objetivos e *stakeholders*. A existência de objetivos antagônicos poderia impedir a satisfação dos objetivos. Além disso, falhas para reconhecer objetivos antagônicos poderiam causar confusão no processo de projeto. Uma abordagem sistemática tem a função de revelar correlações de objetivos, usando matrizes para assegurar que todas as interações possíveis sejam descobertas e que os diagramas tornem claro o padrão dos relacionamentos.

3.5.5. Relacionamento entre modelos

Bubenko *et al.* (1998), explicam que no desenvolvimento de um modelo organizacional completo, as ligações entre os componentes de diferentes submodelos executam tarefa essencial. Por exemplo, declarações no Modelo de Objetivos possibilitam que conceitos

diferentes sejam definidos mais claramente no Modelo de Conceitos, no qual uma ligação é especificada entre o componente correspondente no Modelo de Objetivos e o conceito no Modelo de Conceitos. Da mesma forma, objetivos no Modelo de Objetivos motivam processos particulares no Modelo de Processos de Negócio. Os processos são necessários para alcançar os objetivos declarados. Assim, a ligação é definida entre o objetivo e o processo que deverá ser realizado para cumpri-lo. Ligações entre modelos tornam o conhecimento mais disponível, sendo possível ver por que certos processos e requisitos do sistema de informação têm que ser introduzidos. As ligações entre os submodelos do Modelo Organizacional são explicadas a seguir (figura 2):

- Ligações entre o Modelo de Objetivos e o Modelo de Atores e Recursos podem motivar ou requerer a introdução de novos atores particulares, como Agentes de Relações de Clientes (motivado pelo objetivo de melhorar relacionamentos com clientes). Podem também descrever quais componentes do Modelo de Atores e Recursos são responsáveis por alcançar um particular objetivo ou defini-lo.
- Ligações entre o Modelo de Atores e Recursos e o Modelo de Regras do Negócio descrevem como componentes diferentes do Modelo de Atores e Recursos são relacionados a Regras do Negócio do Modelo de Processos de Negócio. Exemplos de nomes de ligações: define, é responsável por.
- Ligações entre o Modelo de Objetivos e o Modelo de Conceitos são usadas para descrever componentes do Modelo de Objetivos que referenciam entidades do Modelo de Conceitos. Por exemplo, o objetivo: "Melhorar a satisfação do cliente" deveria referenciar o conceito "Cliente" no Modelo de Conceitos.
- Ligações entre o Modelo de Objetivos e o Modelo de Processos de Negócio relacionam objetivos do Modelo de Objetivos a processos do Modelo de Processos de Negócios com o relacionamento "motiva". Exemplo: "Melhorar a satisfação do cliente" poderia inicialmente motivar um particular processo de alto nível da organização como "Monitoramento das Relações com Cliente".
- Ligações entre o Modelo de Objetivos e o Modelo de Regras do Negócio descrevem como componentes diferentes do Modelo de Objetivos são implementados em termos de regras do negócio no Modelo de Regras do Negócio. Por exemplo, o objetivo "Registrar Maus Clientes" declarado no Modelo de Objetivos requer uma regra do negócio no Modelo de Regras do Negócio, que declara como, exatamente,

isso deve ser distinguido, como "Cliente é considerado mau cliente se demorar mais do que quatro semanas para efetuar pagamento".

- Ligações entre o Modelo de Regras do Negócio e o Modelo de Processos de Negócio descrevem como processos do Modelo de Processos de Negócio são disparados pelas regras do negócio do Modelo de Regras do Negócio. Por exemplo, se existe uma regra que declara "Clientes são registrados como maus clientes se demorem mais do que quatro semanas para efetuar pagamento", então essa regra dispara o processo que realiza o registro de maus clientes.
- Ligações entre o Modelo de Processos de Negócio e o Modelo de Conceito estão entre o conjunto de informação do Modelo de Processos de Negócio e os componentes do Modelo de Conceitos. Por exemplo, o Conjunto de Informação "vôo esperado" deve referir-se a entidades incluindo atributos e relacionamentos como Vôo, Linha Aérea, Piloto e dados temporais sobre chegadas.

Figura 2 - Relações entre submodelos

Fonte: Bubenko *et al.* (1998), p. 65

- Ligações entre o Modelo de Requisitos e Componentes Técnicos e outros componentes do modelo podem ser mais complexas do que os relacionamentos binários normais. O Modelo de Processos de Negócio motiva os objetivos do

sistema de informação e os requisitos do sistema de informação. Declarações binárias podem ser simples, como "O Sistema de Catálogo da Biblioteca deve ser capaz de manusear X pedidos simultaneamente", mas podem também ser mais complexas, como "O tempo de resposta para o usuário do tipo X, quando estiver realizando o processo P, na data definida como C, deve ser menor do que Z segundos".

Os componentes do Modelo podem ser ligados de muitas formas. As ligações que devem ser estabelecidas dependem do propósito de cada projeto EKD em particular. Cada Modelo Organizacional tem um propósito e um foco que as ligações de cada modelo devem refletir. Toda ligação representa uma declaração feita sobre a organização e, possivelmente, seus requisitos do sistema de informação. A semântica de todas as ligações deve ser analisada cuidadosamente. Existe um conjunto mínimo de ligações que deveria ser definido pela representação para ser considerado completo.

3.5.6. Considerações sobre o EKD

A modelagem realizada em grupo tem vantagens e desvantagens. As vantagens são as idéias criativas ressaltadas pelo número de pessoas que aumentam o conhecimento e as competências do grupo. As soluções são caracterizadas pelo consenso e são mais próximas à realidade quando "pessoas chaves" participam. Existe um bom balanço entre criatividade e crítica. As desvantagens são tensões sociais que impedem a cooperação: as pessoas presentes que têm prestígio político dentro da organização podem intimidar ou influenciar a participação de outras pessoas, desencorajando o pensamento inovador. Isso também ocorre se houver diferenças substanciais na posição e temperamento dos membros. O trabalho pode ser descoordenado se finalizado por vários indivíduos ou se algumas pessoas abdicarem de suas tarefas e responsabilidades (Bubenko *et al.*, 1998).

3.6.Considerações finais

A modelagem organizacional foi apresentada neste capítulo como forma de capturar requisitos organizacionais para melhorar a compreensão do domínio, para interagir com usuários a fim de que eles entendam o que o sistema pode fazer para melhorar o negócio e adquirir conhecimento da estrutura organizacional e estratégica.

A modelagem organizacional pode ser aplicada para diferentes propósitos e ser realizada por diferentes grupos. Em algumas situações o modelo organizacional pode

ser utilizado como uma técnica para analisar a situação corrente de uma organização, ou parte dela, e traçar novos objetivos para o futuro. Em outras situações, pode ser usado para desenvolvimento de novos produtos ou para reprojetar atividades do negócio.

A abordagem EKD propõe modelar o conhecimento organizacional para entender, analisar, melhorar e para consertar algum sistema. Os componentes do EKD são modelos que examinam a organização de um número de perspectivas inter-relacionadas. Esses modelos irão coletivamente constituir o Modelo Organizacional.

Existe um relacionamento natural e lógico entre o negócio e os requisitos do sistema, sendo assim, o modelo organizacional é parte importante do desenvolvimento de software para captura e especificação dos requisitos, nos quais a determinação explícita dos objetivos, problemas, conceitos, atividades, processos e atores direciona para um sistema que atende às reais necessidades do cliente, além de diminuir custos de manutenção.

CAPÍTULO 4

Redes de Petri

4.1. Considerações Iniciais

Rede de Petri é uma técnica de especificação formal bem estabelecida, amplamente difundida e adequada para a modelagem de sistemas que tenham atividades paralelas, concorrentes, assíncronas, não-determinísticas ou determinísticas. Tais redes possuem grande abrangência e aplicabilidade em diversas áreas como: ciência da computação, engenharias, química, administração de empresas, entre outras. Neste capítulo apresenta-se os principais conceitos das redes de Petri e seu potencial de aplicação.

4.2. Redes de Petri

A teoria inicial das redes de Petri foi apresentada em 1962, na tese de doutorado *Kommunikatin mit Automaten* defendida por Carl Adam Petri, na faculdade de Matemática e Física da Universidade de Darmstadt, Alemanha. A aplicabilidade das redes de Petri como ferramenta para estudo de sistemas, para Peterson (1981), é importante por permitir representar matematicamente e analisar os modelos e ainda fornecer informações úteis sobre a estrutura e o comportamento dinâmico dos sistemas modelados. Como admitem a introdução de alterações, as aplicações das redes de Petri podem se dar em muitas e diferentes áreas (sistemas de manufatura, desenvolvimento de software, sistemas administrativos etc.).

De acordo com Murata (1989), as redes de Petri são formadas por dois tipos de componentes: a transição, que é um componente ativo correspondente a alguma ação realizada dentro do sistema e o lugar, que é passivo e está relacionado a alguma variável de estado do sistema. A realização das ações está associada a pré-condições ou

condições das variáveis de estado do sistema, isto é, existe uma relação entre lugares e transições, a qual possibilita a realização de determinada ação. Da mesma forma, após a realização de uma ação, as informações de alguns lugares (pós-condições) são alteradas. Graficamente, lugares são representados por círculos e transições por traços ou barras (figura 3).

Figura 3 - Elementos básicos das Redes de Petri

Lugares e transições são vértices do grafo associado às redes de Petri e interligam-se por arcos direcionados. Arcos que ligam lugares a transições representam relações entre condições verdadeiras e possibilitam que, em algum momento, ações sejam executadas dentro do sistema. Arcos que ligam transições a lugares representam relações entre ações e condições as quais tornam-se verdadeiras com a execução das ações.

O disparo das transições (execução das ações) é controlado tanto pelo número de marcas, quanto pela distribuição destas nos lugares. Uma transição t está habilitada se e somente se, todos os lugares de entrada ($p_i \in P$) de t são tais que $M(p_i) \geq I(p_i, t)$; esta regra habilita o disparo de uma transição, mas não obriga seu disparo.

Formalmente, a rede de Petri é dada por uma quintupla, $RP = (P, T, F, W, M_0)$, de acordo com Murata(1989):

- $P = \{p_1, p_2, \dots, p_m\}$ é um conjunto finito de lugares;
- $T = \{t_1, t_2, \dots, t_m\}$ é um conjunto finito de transições;
- $F \subseteq (P \times T) \cup (T \times P)$ é um conjunto de arcos;
- $W: F \rightarrow \{1, 2, 3, \dots\}$ é uma função de pesos;
- $M_0: P \rightarrow \{0, 1, 2, 3, \dots\}$ é a marcação inicial e
- $P \cap T = \emptyset$ e $P \cup T \neq \emptyset$.

Denota-se por (N, M_0) uma rede de Petri com marcação inicial.

$M(p_i)$ é a marcação $\in M$ em p_i ,

- $\bullet p = \{t/ (t,p) \in F\}$ é o conjunto de transições de entrada de p ,
- $p\bullet = \{t/ (p,t) \in F\}$ é o conjunto de transições de saída de p ,
- $\bullet t = \{p/ (p,t) \in F\}$ é o conjunto de lugares de entrada de t ,
- $t\bullet = \{p/ (t,p) \in F\}$ é o conjunto de lugares de saída de t .

$R(N, M_0)$ ou simplesmente $R(M_0)$ é o conjunto de todas possíveis marcações alcançáveis a partir de M_0 e $L(N, M_0)$ ou simplesmente $L(M_0)$ é conjunto de todas possíveis seqüências de disparos de transições a partir de M_0 .

Na modelagem, usando o conceito de condições e eventos, o lugar pode ser interpretado como uma condição e a transição como um evento. Uma transição (evento) possui um certo número de lugares de entrada e de saída, representando as pré e pós-condições, respectivamente. A presença de uma marca em um lugar indica que a condição deste é verdadeira.

Diversos sistemas têm seu comportamento descrito em termos de seus estados e de sua respectiva mudança ou transição para outros possíveis estados. Na rede de Petri, a mudança de estados ocorre de acordo com a regra de habilitação e disparo das transições.

A ocorrência de um evento é denominada disparo de transição. A regra de habilitação e disparo de transições pode ser:

- Uma transição está habilitada para disparar se todo lugar de entrada, que possuir um arco para a transição, possuir um número de marcas maior ou igual ao peso do arco.
- Uma transição habilitada pode disparar ou não, dependendo de o evento realmente ocorrer.
- No disparo de uma transição habilitada, todo lugar que possui um arco para a transição tem seu número de marcas reduzido pelo valor do peso deste arco, e todo lugar que possui um arco vindo da transição tem seu número de marcas acrescido do valor do peso deste arco.

Uma transição sem lugares de entrada é denominada transição fonte, e a sem lugares de saída é chamada de transição sumidouro. Uma transição fonte está sempre habilitada, e o disparo de uma transição sumidouro consome fichas, mas produz nenhuma.

Atualmente existem muitas extensões de rede de Petri, tais como: Colorida, Temporizada, Estocástica, Virtual (Ferraz *et al.*, 2001; Inamasu, 1995) e Modular (Inamasu, 1995) entre outras, com finalidade de cobrir algumas deficiências e limitações existentes nas redes ordinárias. Uma das grandes vantagens das redes de Petri é a sua flexibilidade e o alto poder de abstração, possibilitando introduzir e adaptar elementos gráficos e/ou matemáticos com a finalidade de melhor familiarizar usuários de sistemas mais específicos.

As redes de Petri oferecem suporte para a modelagem, análise e simulação de sistemas de eventos discretos. O baixo nível das redes de Petri ordinárias torna muito difícil a sua aplicação para sistemas não triviais. Por isso, redes de Petri de alto nível foram desenvolvidas e estudadas. Todas apresentam a vantagem de permitirem especificações muito mais compactas tornando dessa forma possível a sua aplicação a sistemas muito mais complexos. Outra limitação das redes ordinárias é a impossibilidade de modelar tempo, que é necessário para a modelagem de sistemas com exigências de tempo-real. Em resposta a essa limitação várias formas de rede de Petri temporizadas foram surgindo. A decomposição hierárquica pode ser aplicada às redes de Petri e seu potencial eleva-se quando se utilizam redes de alto-nível.

Murata (1989) apresenta algumas características que considera importantes na utilização das redes de Petri:

- Visualização usando representação gráfica.
- Modelagem por meio de vários níveis de abstração.
- Modelagem de sistemas paralelos e concorrentes.
- Sincronização de eventos.
- Verificação das propriedades dos sistemas, de forma sistemática.

As redes de Petri tem sido amplamente utilizadas para modelar, analisar, simular e controlar sistemas de manufatura (Desrochers & Aljaar, 1994). Elas fornecem modelos muito úteis pois:

- Abstraem as relações de precedência e interações estruturais de eventos estocásticos, concorrentes e assíncronos;
- A natureza gráfica ajuda na visualização de grandes sistemas;

- Conflitos e tamanhos e *buffer* podem ser modelados de maneira fácil e eficiente.
- Travamentos do sistema podem ser detectados;
- Os modelos representam ferramentas de modelagem hierárquica com bem fundamentadas bases matemáticas e práticas;
- Várias extensões de redes de Petri (temporizadas, estocásticas, colorida, entre outras) permitem análises qualitativas e quantitativas de utilização de recursos, falhas, taxa de processamento e outras;
- Os modelos podem ser utilizados também para implementar sistemas de controle em tempo real para Sistemas Flexíveis de Manufatura (FMS). Logo, podem substituir controladores lógico-programáveis

As principais ferramentas para redes de Petri existentes no mercado podem ser encontradas na Internet (Petri, 2004). Cada uma dessas ferramentas oferece funcionalidades específicas para diversas soluções procuradas.

4.3. Subclasses de Redes de Petri

Alguns trabalhos investigam o uso de subclasses de redes de Petri para aumentar o poder de decisão sem reduzir o poder de modelagem das redes de Petri. Nestas subclasses são feitas algumas restrições estruturais às redes de Petri. As principais subclasses são:

- **Redes de Petri Máquina de Estado.** A subclasse das redes de Petri denominada Máquina de Estado, caracteriza-se por restrições ao número de arcos que podem ser entrada e saída de uma transição. Cada transição da rede só pode ter um arco como entrada e um arco como saída. Esta subclasse não possibilita a modelagem de sistemas paralelos e a sincronização.

Definição de Redes de Petri Máquina de Estado: seja uma rede $R_1 = (P, T, I, O, K)$. I é o conjunto das entradas às transições e O é o conjunto das saídas das transições. K é a capacidade dos lugares. R_1 é classificada como máquina de estado se, e somente se, $|I(t_j)| = 1$ e $|O(t_j)| = 1, \forall t_j \in T$.

- **Redes de Petri Grafo Marcado.** A subclasse das redes de Petri denominada Grafo Marcado ou grafo de eventos restringe o número de arcos de entrada e saída dos lugares da rede. Cada lugar só pode ter um arco como entrada e um arco como saída.

Definição de Redes de Petri Grafo Marcado: seja uma rede $R1 = (P, T, I, O, K)$. I é o conjunto das entradas às transições e O é o conjunto das saídas das transições. K é a capacidade dos lugares. $R1$ é classificada como grafo marcado se, e somente se, $|I(p_i)| = 1$ e $|O(p_i)| = 1, \forall p_i \in P$.

- **Redes de Petri Escolha Livre.** A subclasse de redes de Petri denominada escolha-livre possibilita a modelagem de conflito do paralelismo e a sincronização. Quando um lugar é entrada de diversas transições, este lugar é a única entrada destas transições. Desta forma, todas as transições estarão habilitadas ou nenhuma estará, possibilitando a escolha do evento livremente.

Definição de Escolha-livre. Seja uma rede de Petri $= (P, T, I, O, K)$. I é o conjunto das entradas às transições e O é o conjunto das saídas das transições. K é a capacidade dos lugares. Esta rede é classificada como uma rede de escolha-livre se, e somente se, $I(t_j) = \{p_i\}$ ou $O(p_i) = \{t_j\}, \forall t_j \in T$ e $p_i \in I(t_j)$.

4.4. Propriedades das Redes de Petri

Existe uma série de métodos para analisar um grande número de propriedades de sistemas. De acordo com Murata (1989), as propriedades das redes de Petri podem ser divididas em dois grandes grupos: as comportamentais, que dependem da marcação inicial e as estruturais (que não dependem). Algumas das propriedades comportamentais mais comumente utilizadas são descritas a seguir.

- **Alcançabilidade.** Base fundamental para o estudo de propriedades dinâmicas para alguns sistemas. O disparo de uma transição habilitada mudará a distribuição de marcas (marcação) em uma rede de acordo com as regras de transição. Sequência de disparos resultará em seqüências de marcações. Uma marcação M_n é dita ser alcançável de uma marcação M_0 se existe uma seqüência de disparos que transforma M_0 para M_n .
- **Limitabilidade.** Uma rede de Petri (N, M_0) é dita k -limitada ou simplesmente limitada se o número de marcas em cada lugar não exceder um número finito k para alguma marcação alcançável de M_0 , isto é, $M(p) \leq k$, para todos lugares p e para todas marcações $M \in R(M_0)$. Uma rede de Petri (N, M_0) é dita segura se é 1-limitada.

- **Vivacidade.** O conceito de vivacidade é relacionado com a ausência completa de *deadlocks* em sistemas operacionais. Uma Rede de Petri (N, M_0) é dita viva (ou equivalentemente M_0 é dita ter uma marcação viva para N) se, sem se preocupar com qual marcação foi alcançada por M_0 , é possível finalmente disparar alguma transição na rede por processo por meio de alguma seqüência disparada. Isso significa que uma Rede de Petri viva garante operações livres de situações de travamento do sistema, independente de qual seqüência disparada é escolhida inicialmente.
- **Segurança** (*safeness*). Esse conceito é uma particularização do conceito de limitação. Essa propriedade é uma das mais importantes na especificação de sistemas digitais. Um lugar p_i é dito K -limitado se o número de marcas que este lugar pode acumular estiver limitado ao número K . Um lugar que é 1-limitado pode ser simplesmente chamado de seguro.
- **Reversibilidade.** Uma Rede de Petri (N, M_0) é dita reversível se, para cada marcação M em $R(M_0)$, M_0 é alcançável por M . Então, uma rede reversível pode sempre voltar para o seu estado inicial enquanto um pode voltar para algum estado interno. Uma marcação M' é dita um estado interno se, para cada marcação M em $R(M_0)$, M' é alcançável por M .
- **Abrangência de cobertura** (*coverability*). Uma marcação M em uma Rede de Petri (N, M_0) é passível de cobertura (*coverable*) se existir uma marcação M' em $R(M_0)$ igual a $M'(p) \geq M(p)$, para cada p na rede. A abrangência de cobertura está relacionada com a propriedade L1-viva (potencialmente disparável). Seja M uma marcação mínima necessária para habilitar uma transição t . Então t está morta (não é L1-viva) se e somente se M não é abrangente de cobertura. Isto é, t é L1-viva se e somente se M é abrangente de cobertura. Os estados passíveis de cobertura (*coverable*) são importantes, pois possibilitam analisar se em todas as transições haverá possibilidade de disparar pelo menos uma vez.
- **Persistência.** Uma Rede de Petri (N, M_0) é dita ser persistente se, para quaisquer duas transições habilitadas, o disparo de uma desabilita a outra. Uma transição em uma rede persistente, uma vez que seja habilitada, estará habilitada até que ela dispare. A notação de persistência é usada no contexto de esquemas de programas paralelos e em circuitos assíncronos velocidade-independente. Persistência é

relacionada a redes livre de conflito e uma rede persistente segura pode ser transformada em um grafo marcado duplicando alguns lugares e transições. Note que todos grafos marcados são persistentes, mas nem todas redes persistentes são grafos marcados.

- **Fairness.** De acordo com o ponto de vista de *Fairness* Limitado – *B-fair* (*Bounded-fairness*), duas transições t_i e t_j são classificadas como *B-fair*, se o número de vezes que uma delas dispara, enquanto a outra não dispara, é limitado. E de acordo com o conceito de justiça incondicional ou global, uma seqüência de transições $s1$ é classificada como *fair* incondicional se essa seqüência é finita ou se todas as transições da rede aparecem um número infinito de vezes nessa seqüência.
- **Distância simultânea.** Um método relacionado de certo modo com dependência mútua entre dois eventos em um sistema condição/evento. Definimos a distância simultânea entre duas transições t_1 e t_2 em uma Rede de Petri (N, m_0) por: $d_{12} = \max_{\sigma} |\sigma'(t_1) - \sigma'(t_2)|$ (1), em que σ é uma seqüência disparada iniciando por alguma marcação M em $R(M_0)$ e $\sigma'(t_i)$ é o número de vezes que a transição t_i , $i = 1, 2$ dispara em σ . A notação de distância simultânea é um conceito fundamental introduzido por C. A. Petri.

As propriedades Estruturais são descritas a seguir:

- **Limitação estrutural.** Uma rede de Petri $R = (P, T, I, O, K)$ é classificada como estruturalmente limitada (*structural bounded*) se é limitada para qualquer marcação inicial.
- **Conservação.** É uma propriedade importante das redes de Petri possibilitando, por exemplo, a verificação da não-destruição de recursos por meio da simples conservação de marcas. O disparo de qualquer transição não altera o número de marcas, ou seja, os recursos não são criados nem destruídos.
- **Consistência.** Uma rede de Petri tem a propriedade de consistência se disparando uma seqüência de transições a partir de uma marcação inicial M_0 porém todas as transições da rede são disparadas pelo menos uma vez.

4.5. Métodos de Análise das Redes de Petri

Os métodos são utilizados para a verificação das propriedades apresentadas anteriormente. De acordo com Murata (1989), os métodos de análise podem ser classificados em três grupos: Árvore de Cobertura, Equação Fundamental das redes de Petri, Técnicas de Redução e Análise usando invariantes. O método de análise denominado Árvore de Cobertura baseia-se na construção de uma árvore que possibilite a representação de todas as possíveis marcações de uma rede. Para uma dada rede de Petri, com uma marcação inicial, é possível obter diversas marcações para um grande número de transições potencialmente habilitadas. Essas marcações são representadas por uma árvore, na qual os nós são as marcações e os arcos as transições disparadas. Algumas propriedades, tais como limitação (*boundness*), segurança (*safeness*), transição morta (*dead*), e alcançabilidade de marcações podem ser analisadas por meio da Árvore de Cobertura: um gráfico utilizado para representar, de maneira finita, um número infinito de marcações. Para possibilitar a representação finita das marcações, usando a árvore, utiliza-se o símbolo especial w .

A Equação Fundamental das Redes de Petri (EFRP), ou Equação de Estados, possibilita a verificação de diversas propriedades de modelos. Essa técnica de análise tem sido utilizada para a verificação de problemas de alcançabilidade, ou seja, pode-se verificar se uma determinada marcação M_k é acessível a partir de M_0 . A Equação Fundamental possibilita a representação de aspectos comportamentais da rede, pois descreve a inserção e remoção de marcas nos lugares, conforme o disparo das transições. A Equação Fundamental das Redes de Petri é: $M'(p) = M_0(p) + C \cdot s^{-1}$, $\forall p \in P$.

As Técnicas de Redução facilitam a análise de grandes sistemas. A técnica de validação por redução baseia-se na transformação de uma rede de grandes dimensões em uma outra rede mais abstrata, de forma que propriedades como vivacidade (*liveness*), limitação (*boundness*) e segurança (*safeness*) sejam preservadas no modelo resultante após a transformação. A transformação reversa, ou seja, a transformação de modelos abstratos em modelos refinados, de forma hierárquica, pode ser utilizada no processo de síntese. Essas regras são denominadas regras de síntese.

A análise de rede de Petri usando invariantes consiste em determinar os *S-invariantes* e *T-invariantes* (Al-Jaar & Desrochers, 1995 e Murata, 1989). Considerando uma rede de Petri (N, M_0) composta por n transições, m lugares e uma matriz de incidência A , define-se:

- Um *S-invariante* consiste em um vetor y ($m \times 1$) de inteiros positivos, de forma que $Ay = 0$, assim, $M^T y = M_o^T y$, para qualquer $M \in R(M_o)$. Essa equação implica que o total de marcas iniciais em M_o , pesado pelo *S-invariante*, é constante para qualquer seqüência de disparo.
- Um *T-invariante* consiste em um vetor x ($n \times 1$), de inteiros positivos, de forma que $A^T x = 0$, e assim, $M_o = Mf$ para x sendo a seqüência de disparos. Essa equação implica que se o vetor de disparo for um *T-invariante*, então o sistema retorna para o estado inicial após os disparos.

4.6. Aplicabilidade de Redes de Petri

Rede de Petri é uma técnica de especificação formal bem estabelecida, largamente difundida e adequada para a modelagem de sistemas que tenham atividades paralelas, concorrentes, assíncronas, não-determinísticas ou determinísticas. Por ser uma ferramenta gráfica e matemática, a rede de Petri é adaptável a várias aplicações em que são importantes os conceitos de eventos e evoluções simultâneas de um sistema. Algumas aplicações, dentro da Engenharia, estão descritas a seguir.

- **Sistemas de manufatura.** Redes de Petri são muito aplicadas em modelagem, análise e controle de sistemas discretos de manufatura, especialmente devido à sua capacidade de descrição gráfica o que propicia melhor entendimento das interações mais complexas do sistema e livre de *deadlock* (Gang & Zhiming, 2002). Basicamente, o uso de redes de Petri na simulação dos ambientes de manufatura é feito da seguinte forma: lugares representam os recursos do sistema, disparos de transições simbolizam uma atividade que começa e termina com dois eventos consecutivos e o conjunto lugar-transição representa as condições e relações de preferência nas operações do sistema. Nos sistemas de manufatura flexíveis utiliza-se, com muita frequência, a rede de Petri temporizada (Di Febbraro, 2002).
- **Planejamento de processos e estimação de custos.** Um modelo genérico para a representação de planejamento de processos com rede de Petri é apresentado em Kiritsis & Porchet (1996). Tal modelo pode ser utilizado como ferramenta de simulação na detecção de possíveis problemas durante o planejamento. Além disso, a árvore de alcançabilidade da rede de Petri associada retorna todos os planos de processos possíveis para a fabricação de determinada peça sob condições distintas. A estimação de custos em um sistema de manufatura consiste na atividade de

calcular e prever o custo de um conjunto de atividades, antes que essas aconteçam realmente (Silva, 2001). Alguns autores consideram que o projeto da peça é uma tarefa importante, diretamente relacionada aos custos de manufatura (Liebers & Kals, 1997); outros propõem metodologias que classificam e avaliam os custos, relacionando-os às características pertinentes ao formato das peças fabricadas em um sistema de manufatura (Feng *et al.*, 1996); outros ainda sugerem métodos baseados na descrição das características da peça, conjuntamente às operações de manufatura requeridas para sua confecção e aos recursos requeridos em tais operações (Kiritzis *et al.*, 1999). Di Febbraro *et al.* (2002) aplica Integer Timed Petri Net para modelar sistemas de manufaturas automatizados e trata o problema de otimização em que os custos a serem otimizados podem ser associados com a permanência do sistema nos estados tangíveis.

- **Especificação e Verificação de Sistemas de Controle.** O processo de modelagem dos Sistemas de Controle Sequencial parte da descrição comportamental de cada componente, através de regras operacionais. A partir das regras operacionais, são construídas as redes de Petri elementares que modelam cada componente do sistema. A rede de Petri global é obtida pela técnica de fusão de “lugares” comuns das redes elementares. Em Simon (2001) é apresentada uma extensão de redes de Petri temporizada uma lógica de ações que possibilita especificar e verificar processos. O trabalho de Macià *et al.* (2003) apresenta uma extensão Markoviana para melhorar a sincronização e definir uma relação de equivalência estocástica.
- **Modelagem e controle de tarefas em robôs.** Amplamente utilizadas na modelagem de sistemas dinâmicos, tais como os de manufatura, as redes de Petri também são passíveis de utilização na modelagem de tarefas em robôs. Tipos distintos de redes de Petri podem ser utilizados para modelar tipos diferentes de tarefas: rede de Petri interpretada para projeto e execução das tarefas; estocástica generalizada para avaliação quantitativa do desempenho dessas tarefas e rede de Petri ordinária para avaliação qualitativa. Dois exemplos de aplicação, descritos em Lima *et al.* (1998) são: visualização e pegada de objetos e controle de tarefas em robôs móveis. O trabalho de Kim *et al.* (2000) apresenta um sistema de multi-agente de um robô-futebol desenvolvido baseado na teoria de redes de Petri.
- **Workflow.** Zisman (1977) usou redes de Petri para modelar *workflow* pela primeira vez, e na sequência muitos outros autores publicaram trabalhos que procuravam

também integrar os dois assuntos. Entre eles, pode-se citar: Cindio *et al.* (1988), Li *et al.* (1993), Ferscha (1994), Merz *et al.* (1995), Schömig & Rau (1995), Ellis & Nutt (1996), Nutt (1996), Wikarski (1996), Oberweis (1997), Aalst (1998a), Aalst (1998b) Adam *et al.* (1998), Badouel & Oliver (1998), Verbeek *et al.* (2001), Voorhoeve (2000), Janssens *et al.* (2000), Aalst & Hee (2002), Desel & Erwin (2000) e Holt (2000). A ausência de técnicas mais complexas para análise do gerenciamento do *workflow*, e principalmente dos modelos organizacionais, é a principal barreira enfrentada nas pesquisas desenvolvidas até o presente momento.

4.7. Considerações Finais

No presente capítulo apresentou-se os conceitos principais e mais importantes de redes de Petri para demonstrar que a representação gráfica tem sido muito útil, por permitir a visualização dos processos e a comunicação entre eles. Além disso, as redes de Petri exigem uma definição precisa, o que habilita o uso de muitas técnicas analíticas para a análise de desempenho e para a verificação de propriedades lógicas. As várias áreas de aplicações, em especial na Engenharia de Produção, mostram que essa ferramenta constitui um forte e potencial campo de pesquisa.

CAPÍTULO 5

Workflow e Redes de Petri

5.1. Considerações Iniciais

A necessidade de automação de parte ou de todos os processos de negócio, no qual documentos, informações ou tarefas são passados de um participante para outro de acordo com o conjunto de regras procedurais é o que fez surgir o conceito de *workflow*. Esse conceito é cada vez mais valorizado no atual mercado competitivo uma vez que o Sistema de Gerenciamento de *Workflow* deve oferecer apoio à decisões importantes na organização. Entretanto, a falta de uma definição bem formalizada no que se refere à sintaxe e semântica de tais técnicas, dificulta análises mais complexas dos modelos. Nesse caso, como afirmado na introdução deste trabalho, redes de Petri atuam com excelente potencial, uma vez que elas possuem representação gráfica, são de fácil aprendizado, funcionam como linguagem de comunicação entre especialistas de diversas áreas, permitem a descrição dos aspectos estáticos e dinâmicos do sistema a ser representado e ainda possuem o formalismo matemático necessário à utilização de métodos de análise já consagrados. Neste capítulo apresenta-se os conceitos de *Workflow*, processos de negócio e trabalhos que modelam *workflow* diretamente em redes de Petri. Este capítulo está baseado nos conceitos da *Workflow Management Coalition*, uma vez que essa organização tem uma grande representatividade neste assunto.

5.2. Definição

Um *workflow* é um conjunto de atividades que podem ou não ser executadas simultaneamente, com alguma especificação de controle e fluxo de dados entre essas atividades (Leymann & Roller, 1997).

De acordo com Aalst & Hee (2002), o termo “*workflow*” é muito utilizado na literatura com o significado de “processos de negócio”. Para Casati *et al.* (1995), *workflow* é uma coleção de atividades que cooperam entre si para alcançar um objetivo comum.

Workflow Management Coalition (WfMC) é uma entidade cujo objetivo é aumentar a utilização das tecnologias de *workflow* por meio do desenvolvimento de terminologias comuns e padrões. De acordo com a *Workflow Management Coalition* (1996), *workflow* consiste da automação de um processo de negócio, na sua totalidade ou parte, durante o qual documentos, informações ou tarefas são passadas de um participante para outro por ações, de acordo com um conjunto de regras procedurais.

Workflow tem como um dos seus principais objetivos minimizar o problema da coordenação do trabalho nos processos de negócios. De acordo com Pádua & Bispo (2003), o principal problema das atividades baseadas em processos é que não se tem o controle total das atividades que devem ser executadas. Além disso, não se tem uma visão de quais informações estão sendo manipuladas por essas atividades em um dado instante e nem quem as manipula. As técnicas de modelagem e gerenciamento de *workflow* devem ter como objetivo básico minimizar esses problemas, procurando oferecer uma solução flexível no apoio aos processos de negócios, por meio da facilitação de modificações e da criação de novos processos.

Um *workflow* pode representar vários processos e seus relacionamentos. Sistemas de *workflow* consistem de componentes de modelagem e componentes de representação do sistema. A área de modelagem de *workflow* é bastante recente, não existindo, ainda, um conjunto de modelos ou técnicas que formalizem seu desenvolvimento. De acordo com Aalst & Hee (2002), por falta de um padrão de modelagem, muitas organizações estão relutantes em usar os softwares existentes de gerenciamento de *workflow*.

A WfMC desenvolveu um modelo para Sistemas de *Workflow*. Este modelo, chamado *Workflow Reference Model*, identifica as características, terminologias e componentes dos WfMSs, e ainda habilita especificações individuais para serem desenvolvidas dentro desse contexto.

A idéia básica para a compreensão de sistemas de *workflow* é o conceito de processos de negócio. Um processo de negócio consiste em um conjunto de atividades relacionadas que visam atingir um objetivo de negócio, no contexto de uma estrutura organizacional (*Workflow Management Coalition*, 1996). Na figura 4 são representados os relacionamentos existentes entre os conceitos de *workflow*.

Figura 4 - Conceitos e Terminologia Básica de *Workflow*

Fonte: *Workflow Management Coalition* (1999)

De acordo com WfMC (1996), os principais conceitos relacionados a *workflow* estão na figura 4 e são descritos a seguir:

- Processo de negócio: conjunto formado por um ou mais procedimentos ligados ou atividades que coletivamente compreendem um objetivo de negócio, definindo regras e relacionamentos para seu funcionamento.
- Definição de Processos: representação dos processos.
- Sistema de Gerenciamento de *Workflow*: sistema que completamente define,

gerencia e executa *workflow* (processos de negócio) por meio da execução de software cuja ordem de execução é dirigida por uma representação computacional da lógica do *workflow*.

- Instância de Processo: é a execução do processo.
- Atividades: são descrições formais de uma parte do trabalho, que formam um passo lógico dentro de um processo. Uma atividade requer recursos humanos e/ou máquinas para apoiar a execução de processos.
- Instâncias de atividade: representação da execução da atividade, sendo relacionada a uma única instância de processo.
- Itens de trabalho: representação do trabalho que deve ser processado por um participante do *workflow* no contexto de uma atividade dentro de uma instância de processo.

O fluxo de controle permite a sincronização do processo de execução das atividades. A interação dessas atividades por meio do fluxo de controle é representada por conexões. A possibilidade de representação formal dessas conexões é um fator fundamental que deve estar presente em um modelo de *workflow*. Para descrever a natureza do fluxo dos processos e de suas interações, existe um conjunto de termos que são apresentados a seguir:

- *AND-Split*: ponto em que, de uma única linha de fluxo, partem duas ou mais linhas que são executadas em paralelo.
- *AND-Join*: ponto em que duas ou mais atividades, executando em paralelo, convergem em uma única linha de fluxo comum.
- *OR-Split*: ponto em que uma linha de fluxo faz uma decisão entre um número de opções.
- *OR-Join*: ponto no *workflow* no qual uma atividade que tem um número de alternativas reconverge para uma única alternativa.
- Iteração: quando uma atividade fica em repetição até encontrar a condição satisfatória.
- Seqüência: quando as atividades são executadas em seqüência.

Para cada organização, processos de negócios representam características próprias, nesse sentido, um modelo de *workflow* deve representar de forma realística as informações qualitativas e quantitativas da entidade em estudo. A identificação dos tipos de sistemas de *workflow* permitirá que a seleção de um modelo para representar o *workflow* seja facilitada. Alguns dos vários tipos de *workflow* são apresentados a seguir (Nicolao, 1998):

- *Workflow* Orientado a Humanos: envolve humanos na execução de tarefas. Controla e coordena tarefas humanas. Nesse tipo de *workflow*, as seguintes questões são consideradas importantes: tratamento de exceções, liberações de usuários, priorização e encerramento.
- *Workflow* Orientado a Sistemas: envolve sistemas de computadores que executa operações computacionais intensas e softwares especializados em tarefas. Nesse caso, são consideradas importantes as mesmas questões do item anterior.
- *Ad hoc*: apóia a definição rápida e execução de modelos de processos menos complexos que podem ser usados para facilitar o fluxo de um único documento em uma única ocasião. *Workflow Ad hoc* tipicamente envolve pequenos grupos de profissionais que têm a intenção de apoiar pequenas atividades as quais requerem uma solução rápida. A ordenação e coordenação de atividades são controladas por humanos (não são automatizadas).
- Administrativo: envolve processos repetitivos com regras de coordenação de tarefas simples tal como roteamento de um relatório de despesa ou requisição de viagem por meio de um processo de autorização. A coordenação e ordenação de tarefas podem ser automatizadas. *Workflows* Administrativos não englobam um processo de informações complexas e não requerem acesso para sistemas de informação múltiplos usados para apoiar produção ou serviços de compradores.
- Produção: envolve processos de negócios repetitivos e previsíveis, tais como empréstimos e seguros. *Workflows* de produção englobam processos complexos envolvendo acesso para múltiplos sistemas de informação.
- *Workflow* transacional: envolve execução e coordenação de múltiplas tarefas que podem: a) envolver humanos, b) requerer acesso para sistemas do tipo HAD (autônomo, heterogêneo, distribuído) e c) apoiar o uso seletivo de propriedades transacionais (exemplos: atomicidade, consistência, isolamento e durabilidade) para

tarefas individuais e entradas de *workflow*. O uso seletivo de propriedades transacionais é necessário para permitir a especialização das funcionalidades de cada *workflow*.

5.3. Gerenciamento de *Workflow*

O termo Gerenciamento de *Workflow* (*Workflow Management*) refere-se a idéias, métodos, técnicas e software usados para apoiar processos de negócios estruturados. O fluxo e a distribuição de trabalho devem ser controlados pelos Sistema de Gerenciamento de *Workflow* (WfMS). Após a definição de processo, o WfMS verifica se as atividades estão sendo executadas na seqüência correta e notifica o usuário sobre a execução das tarefas. Em geral, os WfMS predefinem a seqüência em que as atividades serão executadas, determinam qual participante executará cada atividade e acompanham a situação de uma determinada instância de *workflow*. Outros termos para caracterizar sistemas de gerenciamento de *workflow* são: sistemas de operações de negócio, gerenciamento de *workflow*, gerenciamento de casos (case manager) e sistema de controle logístico (Aalst & Hee, 2002).

Por meio do armazenamento de alguns atributos de cada instância de *workflow*, o WfMS poderá gerar dados estratégicos que refletem a eficiência e a eficácia dos processos, atualmente, desempenhados pela organização. A confiança nos dados e o controle dos processos proporcionam melhor atendimento aos clientes, melhor controle do andamento das atividades e menores tempos para conclusão dos processos.

Na figura 5 é descrito a composição de um Sistema de Gerenciamento de *Workflow* (WFSG) dividindo-o em duas partes: modelo de negócios (definição dos processos) e módulo de execução dos negócios (representação de serviços do negócio). A primeira é utilizada durante a construção para gerar uma definição computacional de um processo de negócio e fornecer ferramentas gráficas de modelagem, auxiliando, assim, o desenvolvedor durante o projeto, teste e validação do processo. O módulo de execução é composto de um conjunto de programas responsáveis pela criação e controle de exemplos de processos durante a execução e fornece a interface necessária no cumprimento de passos específicos do processo (Salimifard & Wright, 2001).

Janssens *et al.* (2000) afirmam que, comparados a Modelos de Base de Dados, os Modelos de *Workflow* têm muito a amadurecer. Há a necessidade de padrões em modelagem de *workflow*.

De acordo com a WfMC (1996), a idéia básica para buscar a padronização de interfaces importantes de *workflows* funcionais é dirigida por duas maiores considerações:

- Apoio para reengenharia de negócios e flexibilidade operacional.
- Integração dos requisitos resultantes da especialização de produtos e variedade de mercado.

Figura 5 - Fases e componentes de um sistema de negócios

Fonte: Salimifard & Wright, 2001.

Os modelos de *workflow* precisam disponibilizar expressões relacionadas aos processos, restrições temporais, trocas dinâmicas e tratamento de exceções. O sistema precisa habilitar a execução dinâmica, a modelagem baseada em processos, o processamento distribuído e providenciar a coordenação e assistência aos usuários a cada passo da modelagem. Além disso, é necessário que o sistema de *workflow* realize a recuperação histórica do fluxo de trabalho para permitir o monitoramento histórico (auditorias).

De acordo com Janssens *et al.* (2000), a estrutura do *workflow* é extremamente volátil como uma consequência das mudanças das regras dos processos de negócio. O ambiente do negócio e as condições mudam muito rapidamente. A evolução dos sistemas é inevitável porque processos de negócio modificam devido às reformas organizacionais e mudanças do ambiente externo. Assim, os modelos de negócios influenciam nos dois fatores responsáveis por mudanças: dados do sistema do negócio e regras do sistema.

Desel & Erwin (2000) elevam a importância dos Sistemas de Gerenciamento de *Workflow* no atual mercado competitivo, uma vez que devem oferecer apoio para

decisões importantes. Entretanto, para que o sistema seja capaz de conduzir algumas avaliações, requisitos de formalismo devem ser usados para construir os modelos, o que reduz significativamente o risco de decisões errôneas em relação ao projeto do modelo de negócios. Os cenários podem ser executados imediatamente para discussão e análise.

5.4. Por que usar Redes de Petri em Modelagem de *Workflow*

O principal propósito de um sistema de gerenciamento de processos de negócio é o apoio da definição, da execução, do registro e do controle de processos. Em razão dos processos de negócio serem fatores dominantes no gerenciamento de *workflow*, é importante usar uma estrutura estabelecida para modelagem e análise dos processos de negócio. Nesse caso, rede de Petri é uma técnica de modelagem bem fundamentada. Desde que a rede de Petri foi criada, ela tem sido utilizada para modelar e analisar todos os tipos de processos com aplicações, estendendo desde protocolos, hardware e sistemas de manufatura flexível, interação com usuário e processos de negócio (Aalst & Hee, 2002). Nas últimas duas décadas, a rede de Petri Clássica foi estendida para colorida, temporizada e hierárquica. Essas extensões facilitam a modelagem de processos complexos em que dados e tempo são fatores importantes.

Na modelagem de processos de negócio utilizando redes de Petri, cada tarefa é representada por uma transição correspondente. Lugares representam pré-condições e pós-condições ou ainda, os recursos requeridos para a execução de uma determinada tarefa. Os arcos representam as relações lógicas entre as tarefas e o próprio fluxo de trabalho (Salimifard & Wright, 2001).

De acordo com Holt (2000), as redes de Petri permitem desenhar planos de sistemas gerais caracterizados por um alto nível de concorrência. Na estrutura de *workflow*, tais planos são esboços globais ou parciais da forma específica de realizar alguns tipos de atividades complexas e principalmente distribuídas. Essas atividades serão focadas como específicas (planejamento de recursos de vários tipos) ou, como controladoras de falhas, custos ou carga de trabalho. Desde que todo o negócio possa ser considerado como um sistema que envolve operações complexas e concorrentes (produção, marketing, contabilidade, procedimentos financeiros, negociações etc), o uso das redes de Petri em seu planejamento ou análise parece uma idéia natural. Um modelo em rede de Petri da estrutura causal e operacional do *workflow* pode fornecer firme

fundamentação lógica e contexto realístico nas questões de gerenciamento do negócio, o que permite a aplicação de algoritmos provenientes de: matemática aplicada, teoria da decisão, teoria de jogos, pesquisa de operação, estatística etc. A chave do sucesso para a implantação desses algoritmos dentro de redes de Petri é a flexibilidade da modelagem de redes do tipo Predicado / Transição.

Oberweis *et al.* (1997), identificam outros motivos para optar pelo uso de redes de Petri nesse contexto:

- Integração de dados e aspectos comportamentais.
- Apoio concorrente e corporativo.
- Níveis diferentes de formalidade.
- Disponibilidade de técnicas de análise.
- Flexibilidade.

Segundo Merz *et al.* (1995), a principal força de redes de Petri na modelagem de processos de negócio é resultante da combinação de: fundamentação matemática, representação gráfica abrangente, possibilidade de executar simulações e verificações.

Aalst (1998a) declara que experiências no projeto Sagitta 2000² mostraram que redes de Petri podem ser usadas para modelar processos de negócio de forma simples. Pessoas que têm experiência anterior em ciência da computação são capazes de especificar procedimentos de processos de negócio. Além disso, redes de Petri são fáceis de usar devido a sua natureza gráfica, são bem fundamentadas e possuem semânticas formais. O fato de procedimentos de *workflow* serem especificados usando uma técnica com semântica formal é vital para o sucesso de projetos de processos de negócio, tais como, Sagitta 2000. A semântica formal de redes de Petri representa muitas vantagens:

- O procedimento de processo de negócio especificado em termos de redes de Petri não é ambíguo. O resultado da construção é claro e não existe abertura para múltiplas interpretações. Dessa forma, é possível evitar discussões intermináveis sobre o significado preciso do procedimento.

² O projeto Sagitta-2000 iniciou-se em 1994. O objetivo desse projeto é desenvolver um sistema de informação para o processamento de todos os tipos de declarações de clientes, que são complexas e sujeitas a muitas mudanças.

- Uma descrição de rede de Petri de um processo de negócio pode servir como um contrato entre sub-departamentos. A semântica formal pode ser usada para resolver conflitos sobre interpretação de procedimentos comuns de processos de negócio.
- A interpretação de uma rede de Petri baseada em procedimentos de processos de negócio é uma ferramenta independente, não é trocada quando uma nova versão do sistema de gerenciamento de processos de negócio é substituída.
- A semântica formal permite raciocínio entre propriedades de um procedimento de processos de negócio. É possível provar a ausência de propriedades dinâmicas, tais como, *deadlock* e *livelock*.
- A semântica formal forma um pré-requisito para aplicações de todos os tipos de técnicas de análise.

Aalst (1998a) apresenta ainda três razões para utilizar redes de Petri na modelagem de *workflow*:

- A primeira razão é o fato da lógica de negócios poder ser representada por uma linguagem formal, mas também gráfica. A semântica de redes de Petri clássica e das muitas extensões (colorida, hierárquica, por exemplo) foram definidas formalmente.
- A segunda razão é que em contraste a muitas outras técnicas de modelagem de processos, o estado de um caso pode ser modelado explicitamente em redes de Petri. As técnicas de modelagem de processos, abrangendo desde técnicas informais como diagrama de fluxo de dados até técnicas formais como álgebra de processos, são baseadas em eventos. As transições são modeladas explicitamente e os estados entre subseqüentes transições são modelados implicitamente. Atualmente, Sistemas de Gerenciamento de Processos de Negócio são tipicamente baseados em eventos. As tarefas são modeladas explicitamente e os estados entre tarefas subseqüentes são suprimidos.
- A terceira razão é que as redes de Petri são marcadas pela disponibilidade de muitas técnicas de análise. Essa é uma grande vantagem dos Sistemas de Gerenciamento de Processos de Negócio baseados em redes de Petri. O formalismo de redes de Petri permite a representação de processos de negócio de uma maneira natural. A representação serve como ponto inicial de vários tipos de análise. Nesse sentido, a representação de redes de Petri serve como uma interface entre o processo de negócio tratado e o método de análise. Na realidade, redes de Petri fornecem um

solucionador independente que pode ser usado para fazer um esquema da definição do processo de negócio que deve ser analisada. Esse esquema pode ser usado em diferentes níveis de tomada de decisão e como um ponto inicial para vários meios de análise. Comparado a abordagens algorítmicas usuais, nas quais a ênfase é na análise do processo ao invés de no processo de modelagem, essa abordagem é caracterizada pelo fato de que durante o processo de modelagem o usuário não é “engessado” pelas técnicas que são utilizadas para analisar o modelo.

As características de processos de Negócio baseadas em redes de Petri são mais proeminentes na fase de projeto e análise. Seis tipos primitivos de processos de negócio identificados pela Coalisão de Gerenciamento de Processos de Negócio (*Workflow Management Coalition*) podem ser mapeados dentro de redes de Petri (figura 6):

- Tarefas são mapeadas em transições e relações causais são representadas no modelo por lugares.
- A transição t1 representa no modelo a sincronização de dois subfluxos (*AND-join*).
- Transição t21 e t22 representam um *OR-join*: dois subfluxos são unidos em um subfluxo.
- A transição t3 representa um *AND-split*: um subfluxo é dividido em dois subfluxos paralelos.
- Transições t41 e t42 representam um *OR-split*: uma seleção é apresentada entre duas alternativas.
- A iteração pode ser representada pela adição de uma transição de *feedback* (t52).
- Conectando duas transições (t61 e t62) por meio de um lugar intermediário resulta em duas tarefas seqüenciais.

De acordo com Aalst (1998a), as primitivas da figura 6 são usadas para definir processos de negócio. Entretanto para completar a definição de processos de negócio deve-se adicionar uma dimensão que cuida do mapeamento das tarefas dos recursos. A dimensão de procedimentos e a dimensão de gerenciamento de recursos são ortogonais e por essa razão são difíceis de visualizar em uma rede de Petri. Ainda é possível modelar procedimentos e gerenciamento de recursos em uma maneira integrada usando uma rede de Petri de alto nível com extensão colorida e hierárquica.

Muitos dos Sistemas de Gerenciamento de Processos de Negócio disponíveis no mercado fornecem construções sem qualquer semântica formal. Existem sistemas que impõem sérias restrições nas primitivas apresentadas na figura 6. Aalst (1998a) afirma que a ferramenta WANG's OPEN não apóia fluxos paralelos. Alguns Sistemas de Gerenciamento de Processos de Negócio também fornecem construções diferentes cuja semântica não é totalmente clara. A mudança nas definições de processo de negócios entre dois sistemas baseados em redes de Petri é mais fácil comparada à troca de definições de processos entre Sistemas de Gerenciamento de Processos de Negócio baseados em conceitos diferentes. Na figura 7 é apresentado um diagrama típico que define o procedimento do negócio. As tarefas A, B, C, D, E e F são representadas explicitamente em contraste ao estado.

Figura 6 - Primitivas de Processos de Negócio

Fonte: Aalst (1998a) p. 189

Figura 7 - Uma descrição baseada em evento de um procedimento de *workflow*
 Fonte: Aalst (1998a) p. 192

Se uma descrição baseada em evento como na figura 7, for convertida para uma rede de Petri, obtém-se a rede exibida na figura 8. As tarefas são modeladas pelas transições e os estados intermediários são modelados por lugares. Em contraste à descrição da figura 7, é possível referir a estados entre a execução das tarefas subsequentes.

Figura 8 - Uma descrição baseada em estados de um procedimento de processos de negócio.
 Fonte: Aalst (1998a) p. 192

Para Aalst (1998a), a descrição baseada no estado permite uma clara distinção entre a habilitação e a execução de uma tarefa. Posto que a habilitação de uma tarefa não implica que a tarefa será executada imediatamente, é importante ter essa distinção. A execução de uma instância de tarefa para um caso específico inicia-se no momento em que a instância é disparada. Uma tarefa instância pode apenas ser disparada se o caso correspondente está em um estado que habilita a execução de uma tarefa. Na figura 8, a

tarefa D pode apenas ser disparada para o caso c se houver uma marca em cada lugar de entrada de D. Os seguintes disparos são possíveis:

- 1 – Automático: uma tarefa é disparada no momento em que é habilitada. Esse tipo de disparo é usado para tarefas que são executadas por uma aplicação que não requer interação humana.
- 2 – Usuário: a tarefa é disparada por um participante humano, o usuário seleciona uma instância da tarefa habilitada para ser executada. Em um sistema de gerenciamento de processos de negócio cada usuário tem instâncias de tarefas que podem ser executadas por ele. Na literatura é chamada de *in-basket*.
- 3 – Mensagem: um evento externo dispara uma instância de tarefa habilitada.
- 4 – Tempo: uma instância de tarefa habilitada é disparada pelo relógio, assim dizendo, uma tarefa é executada em um tempo pré-definido. Por exemplo, a tarefa remover documento é disparada se um caso estiver preso em um estado específico por mais de 15 horas.

Apenas para tarefas automáticas a habilitação e a execução de uma tarefa coincidem. Por essa razão, é importante modelar os estados intermediários explicitamente. Além disso, existe a possibilidade de tarefas competitivas. Duas tarefas são competitivas se elas são habilitadas e apenas uma pode ser executada. Na figura 9 são apresentadas duas tarefas competitivas B e C. Na figura 9 é demonstrada a forma que cada tarefa é disparada. A tarefa A é disparada por uma mensagem externa. A execução da tarefa A é seguida pelo disparo de B ou C. Se o usuário seleciona a instância da tarefa B antes de algum tempo definido, então B é executada, se não, a tarefa C é executada. A execução da tarefa C implica que instâncias de tarefas têm que ser removidas da lista de tarefas dos participantes permitidos a executar a tarefa B. Para modelar essa situação não é possível usar uma descrição baseada em eventos. A escolha entre a tarefa B ou C é feita durante a execução da tarefa A. Essa escolha é feita pelo ambiente do Sistema de Gerenciamento de Processos de Negócio. De acordo com Aalst (1998a), muitas ferramentas de modelagem são incapazes de modelar essa situação porque o estado intermediário p2 é suprimido.

Figura 9 - Tarefa B e C são competitivas

Fonte: Aalst (1998), p. 194

Embora seja admitido por todos que WfMSs devem fornecer flexibilidade, de acordo com Bassil *et al.* (2002), muitos WfMSs têm problemas para tratarem mudanças. Aalst & Hee (2002) afirmam que algumas vezes, é necessário modificar ou retirar um caso. Para muitos sistemas isso é uma situação difícil. Instâncias de tarefas devem ser removidas dos *in-baskets* dos participantes. Em um sistema baseado em redes de Petri tal retirada é inteiramente fácil: simplesmente remove-se todas as marcas e disparadores que correspondem a um caso cancelado.

Existem muitas razões para usar Sistemas de Gerenciamento de Processos de Negócio baseados em estados e não em eventos. Sistemas baseados em eventos podem ser usados, satisfatoriamente, apenas em situações em que a execução é conduzida, por exemplo, as tarefas são disparadas pelo Sistema de Gerenciamento de Processos de Negócio ao invés de ser disparada pelo ambiente do sistema. Em muitas situações essa não é a realidade. O sistema deveria seguir e guiar o ambiente no lugar de impor todos os tipos de restrições (Aalst, 1998a).

Em geral, as técnicas de análise são utilizadas para provar propriedades e para calcular medidas de performance (tempo de resposta, tempo de espera, taxa de ocupação). Pela construção da árvore de alcançabilidade é possível verificar se existe uma determinada propriedade ou estados indesejáveis como *deadlock*. Além disso, é possível usar técnicas que exploram a estrutura de redes de Petri fundamental. Por exemplo, é possível gerar lugares invariantes para verificar propriedades de segurança (Aalst & Hee, 2002).

Soundness é um critério de verificação de correitude definido para *workflow-nets*. *Sound* é sinônimo de correto de acordo com Aalst & Hee (2002). Aalst (1997)

desenvolveu uma técnica que verifica em tempo polinomial³ se o procedimento satisfaz os seguintes requisitos (de *soundness*):

1. Não deve existir tarefa que não contribui para o processamento dos casos.
2. Para qualquer caso, o procedimento terminará eventualmente.
3. No momento em que o procedimento termina para casos específicos, todas as referências a esse caso devem ser removidas.

Não existe nenhuma unanimidade sobre quais classes de redes de Petri, entre as de alto e baixo nível, que se adaptam melhor para as necessidades específicas da modelagem de processos de negócio. Um comentário interessante, entretanto, é que o bom formalismo de redes de Petri não é muito valorizado nesse caso, se não existir nenhum sistema de gerenciamento ou outras ferramentas computacionais baseadas nele. A utilidade de ferramentas computacionais adequadas é um fator crítico no uso prático de redes de alto nível e métodos de análise relacionados (Janssens *et al.*, 2000).

De acordo com Voorhoeve (2000), processos simples de *workflow* podem ser modelados diretamente como sistemas de transição. Entretanto, modelar processos mais avançados (com muitos paralelismos) torna-se difícil através de *workflow*, devido ao grande número de estados. Nesse caso, redes de Petri permitem uma representação mais concisa e natural dos processos. A idéia básica é que a tarefa pode ocorrer quando certos objetos (materiais, recursos ou permissões) estão presentes. Quando a tarefa inicia, esses objetos são consumidos. No final, novos objetos são produzidos, permitindo que novas tarefas se iniciem. Em Voorhoeve (2000), são tratadas algumas combinações de redes de Petri e álgebra de processo.

5.5. Modelando os conceitos de Processos de Negócio para Redes de Petri

Usar um processo em um sistema de Gerenciamento de Processos de Negócio indica que uma categoria particular de casos deveria ser gerenciada. O processo define quais tarefas precisam ser executadas. É importante que existam as informações sobre as tarefas a serem realizadas e sobre as condições dos processos. Dessa forma, é definida a

³ Para medir a eficiência de um algoritmo freqüentemente usa-se o tempo teórico que o programa leva para encontrar uma resposta em função dos dados de entrada. Este cálculo é feito associando-se uma unidade de tempo para cada operação básica que o procedimento executa. Se a dependência do tempo com relação aos dados de entrada for polinomial, o programa é tratável.

ordem na qual as tarefas precisam ser executadas. Um processo pode consistir de sub-processos, tarefas e condições.

Um determinado estado do sistema pode levar à execução de um processo (disparo de uma transição) que, muitas vezes, depende da disponibilidade de uma pessoa. As condições têm duas funções importantes: assegurar que as tarefas procedam na ordem correta e verificar se o estado do caso pode ser estabelecido

Aalst & Hee (2002) apresentam um exemplo de especificação do processo de gerenciamento de reclamações usando redes de Petri. A entrada da primeira reclamação é registrada; o cliente que reclamou e o departamento afetado pela reclamação são informados. O departamento informado da reclamação pode ser questionado e o cliente é consultado para fornecer mais informações. Essas duas tarefas podem acontecer simultaneamente (em paralelo) ou, ainda, em qualquer ordem. Depois, os dados são reunidos e a decisão, que pode ser um pagamento ou o envio de uma carta, é tomada. Na figura 10 é demonstrado como representar esse processo com redes de Petri.

Cada uma das tarefas registro, informar cliente, informar departamento, pagar e arquivar é modelada utilizando uma transição. A avaliação de uma reclamação é modelada utilizando duas transições: positivo e negativo. As transições positiva e negativa correspondem respectivamente a uma decisão positiva e negativa. Os lugares início e fim correspondem ao início e fim do processo. Os outros lugares correspondem às condições. As condições asseguram que as tarefas prossigam em uma ordem correta e que o estado do caso possa ser estabelecido. O lugar c8, por exemplo, assegura que a reclamação é arquivada apenas se estiver completamente resolvida.

Casos são ilustrados por meio de marcas; um caso pode ser representado por uma ou mais marcas. Na figura 10, a marca no lugar início mostra a presença de um caso. Assim que a transição registrar disparar existirá uma marca em c1 e outra em c2, que representam o mesmo caso. O número de marcas que existirá em um determinado caso é igual ao número de condições satisfeitas. Quando existir uma marca no lugar fim, o caso foi completado. Em princípio cada processo deveria obedecer dois requisitos: (1) ser possível de realizar por meio da execução das tarefas; (2) todas as marcas devem desaparecer para que ocorra o aparecimento de uma marca no fim.

Figura 10 - O processo gerenciar reclamação modelado em Redes de Petri
 Fonte: Aalst & Hee (2002). p. 50.

Esses dois requisitos asseguram que cada caso que começa no lugar início será completado corretamente. Não é possível haver uma marca no lugar fim se existir tarefas ainda não completadas. O estado de um caso não é determinado somente pelas condições que devem ser satisfeitas, o caso pode ter um ou mais atributos. Nesse caso, a extensão colorida pode ser interessante. Uma marca pode ser diferenciada e distinguida sobre os atributos do caso em questão (Aalst & Hee, 2002)

As marcas que correspondem as situações particulares são mantidas estritamente separadas (pelo sistema de gerenciamento de processos de negócio). É possível traduzir as marcas dentro do modelo de redes de Petri em duas formas: produzindo cópias separadas da rede de Petri para cada caso ou fazendo uso da rede de Petri com marcação diferenciada. Neste caso, cada marca tem uma identidade sendo possível através desse valor identificar a situação que a marca se refere (figura 11).

Na figura 11, é ilustrada a existência de cinco casos correntemente em progresso no modelo. Para assegurar que as propriedades das marcas de diferentes casos não se confundam, cada transição possui uma pré-condição de que apenas marcas de um mesmo caso podem ser consumidas em qualquer disparo. Se a transição coletar da figura 11 disparar, sua pré-condição assegura que duas marcas, para o caso 3, serão consumidas.

Figura 11 - Exemplo de modelo em que cada caso usa uma ou mais marcas
 Fonte: Aalst & Hee (2002), p. 51.

A construção das rotas dos processos é determinada pela necessidade do negócio. Algumas tarefas podem ser opcionais logo, a ordem em que essas são realizadas variam de um caso para outro. De acordo com Aalst & Hee (2002), existem quatro construções básicas, para cada construção apresenta-se uma correspondência em redes de Petri:

1 – Sequencial — referenciada quando as tarefas devem ser executadas uma após a outra; se duas tarefas precisam ser executadas sequencialmente, elas são interdependentes, ou seja, a realização da primeira é requerida para que a segunda possa ocorrer; em rede de Petri modela-se essa forma de rota com o uso de um lugar para fazer a ligação de duas tarefas (figura 12).

Figura 12 - Rota sequencial
 Fonte: Aalst & Hee (2002), p. 53.

2 – Paralela — quando mais de uma tarefa pode ser executada ao mesmo tempo ou em qualquer ordem; a situação, com duas tarefas paralelas, pode acontecer de três formas diferentes (figura 13):

- i. Ambas as tarefas podem ser realizadas simultaneamente.
- ii. A tarefa 1 pode ser executada primeiro e em seguida a tarefa 2.
- iii. A tarefa 2 pode ser executada primeiro e depois a tarefa 1.

Figura 13- Rota paralela
 Fonte: Aalst & Hee (2002), p. 53.

Para habilitar a execução paralela das tarefas 1 e 2 com a marca correspondente em c1, inicia-se com o *AND-split*. Essa é uma tarefa adicionada para permitir que mais de uma tarefa seja gerenciada ao mesmo tempo. Na figura 13, a transição t1 é equivalente a um *AND-split*. A transição t1 dispara quando existe uma marca em c1 e produz uma marca em c2 e c3. Se uma condição em c2 for satisfeita, para um caso particular, então a tarefa 1 poderá ser executada. Da mesma forma, se a condição c3 for satisfeita, a tarefa 2 pode ser executada. Ao disparar t1, as duas tarefas são habilitadas. As tarefas 1 e 2 podem ser executadas em paralelo. Apenas quando ambas estiverem concluídas é que a t2 dispara. *AND-join* é uma tarefa adicionada para sincronizar dois ou mais fluxos paralelos. Essa tarefa pode ser realizada apenas quando as condições c4 e c5 estiverem completadas.

Na figura 13, foram inseridas as tarefas t1 e t2 para modelarem o *AND-split* e *AND-join*. Essas adições são chamadas artificiais porque não correspondem a uma parte de trabalho reconhecível. Em um processo de negócio no qual os casos são executados totalmente de forma manual (sem ajuda de um sistema de gerenciamento), a rota seqüencial é mais comumente utilizada devido a limitações físicas. Por exemplo, as tarefas em um caso particular devem ser executadas uma depois da outra visto que o documento de acompanhamento pode estar apenas em um lugar naquele momento. Pela introdução de um sistema de gerenciamento de processos de negócio, tais limitações são amplamente eliminadas. Tarefas que previamente eram executadas seqüencialmente podem agora estar em paralelo, proporcionando grandes economias de tempo.

- 3 – Rota seletiva: o processo determina a rota para um tipo de caso específico. Mas pode haver diferenças de rotas entre casos individuais. Considere-se, por exemplo,

um processo que negocia com reivindicação de seguro (figura 14). Dependendo das circunstâncias específicas de um pedido, uma rota particular será selecionada. Uma vez que a condição c1 for completada, t11 ou t12 dispara. Se t11 disparar então a tarefa 1 é habilitada, caso contrário, a tarefa 2 é habilitada. A rede consistindo das transições t11 e t12 e lugares c2 e c3 é um *OR-split*. No momento em que as tarefas estiverem concluídas, o *OR-join* assegura que uma marca aparece em c6. Nesse caso, o *OR-join* é modelado usando uma rede consistindo de dois lugares (c4 e c5) e duas transições (t21 e t22). Assim o *OR-split* seleciona uma das duas alternativas e *OR-join* restabelece as duas juntas. Isso é necessário quando se objetiva exibir o *OR-split* e *OR-join* como tarefas gerenciais explícitas. Entretanto, é possível modelar essa situação implicitamente como na figura 15. Quando um caso completa a condição c1, a tarefa 1 ou a tarefa 2 será executada. Esse é um outro exemplo de rota seletiva.

Figura 14 - Rota seletiva 1
Fonte: Aalst & Hee (2002), p. 55.

Figura 15 - Rota seletiva 2
Fonte: Aalst & Hee (2002), p. 55.

A escolha entre duas alternativas é não-determinística. Em outras palavras, não é explicado como a escolha entre a tarefa 1 e a tarefa 2 é feita, porque não importa

qual tarefa é realizada: a seleção é deixada para o ambiente do sistema de gerenciamento de processo de negócio. Em muitos casos, entretanto, a seleção é feita de acordo com específicas propriedades do caso. Dependendo dos valores de atributos do caso (parâmetros gerenciais), é possível escolher entre alternativas. Na figura 16 é demonstrado como modelar essa situação. A rede de Petri colorida deve ser usada para habilitar a decisão. Usando os atributos do caso em questão, a regra de decisão em t1 determina qual tarefa deveria ser realizada. Assim, assume-se que todos os atributos semelhantes desse caso estão contidos no valor da marca em c1. No caso da rota paralela, entretanto, pode existir mais do que uma marca designada para o mesmo caso. Por causa dos atributos relacionados a todo caso, essas marcas devem ter valores idênticos. Em outras palavras, nunca deve haver duas marcas designadas para o mesmo caso com valores diferentes. Para reforçar isso, é importante assegurar que a mudança de um atributo do caso, causada pela performance de uma tarefa, atualiza os valores de todas as marcas pertencentes àquele caso.

Na figura 16, o número de marcas produzidas em cada lugar de saída de t1 é variável (0 ou 1). A escolha é feita baseada no valor (atributos de casos) de uma marca em c1 e a regra de decisão em t1. Entretanto, é possível produzir essa escolha usando duas transições com pré-condições apropriadas. Aqui, a pré-condição é baseada nas cores das marcas para ser consumida e age como uma transição guarda. Na figura 17 é apresentada essa situação.

Figura 16 - Rota seletiva 3

Fonte: Aalst & Hee (2002), p. 57.

Figura 17 - Rota seletiva 4

Fonte: Aalst & Hee (2002), p. 58.

A pré-condição na transição t11 corresponde aos requisitos necessários para justificar a escolha da tarefa 1. A pré-condição t12 determina quando a tarefa 2 deve ser selecionada. Se a pré-condição em t11 é a negação da pré-condição em t12, então cada marca em c1 resultará em uma escolha determinística entre as tarefas 1 e 2. Nesse caso, então, o *OR-split* na figura 16 e 17 são equivalentes.

- 4 – Rota repetitiva (iterativa): é a execução repetida de uma tarefa particular. Idealmente, uma tarefa será realizada uma vez por caso. Em certas situações, entretanto, é necessário aplicar a rota repetitiva. Um exemplo é quando uma certa tarefa precisa ser repetida até que o resultado de um subsequente teste seja positivo (figura 18). Tomando o caso correspondente com a marca em c1, percebe-se que a tarefa 1 e a tarefa 2 são realizadas sucessivamente. Uma vez que a tarefa 2 tenha sido completada, o *OR-split* determina se é necessário ou não realizar novamente. Uma vez que a tarefa 2 foi executada uma ou mais vezes o caso move para a tarefa 3.

Na figura 18, a tarefa 2 deve ser realizada ao menos uma vez entre a tarefa 1 e tarefa 3 (repete... até...). Na figura 19, a tarefa 2 pode não ser realizada nem uma vez (enquanto... faça). Imediatamente após completar a tarefa 1, o *OR-split* determina se a tarefa 2 precisa ou não ser executada, sendo possível para a tarefa 1 seguir diretamente para a tarefa 3.

Em ambos os exemplos, existe um *OR-split* que representa uma decisão baseada sobre os valores correntes dos atributos dos casos. As duas rotas ilustradas correspondem às construções familiares “*repeat..until e while ... do*” que aparecem em muitas linguagens de programação.

Figura 18 - Rota repetitiva (iterativa)
Fonte: Aalst & Hee (2002), p. 60.

Figura 19 - Rota repetitiva (iterativa) 2
Fonte: Aalst & Hee (2002), p. 60.

A tarefa primária de um sistema de gerenciamento de processos de negócio é representar processos de negócio dirigidos a casos pela junção de muitas perspectivas. As seguintes perspectivas são relevantes para a modelagem e execução de processos de negócio (Verbeek *et al.*, 2001).

- 1 – Fluxo de controle ou processo: as definições dos processos são criadas para especificar quais tarefas precisam ser executadas e em que ordem.
- 2 – Recurso: nessa perspectiva são especificadas a estrutura organizacional e a população. Recursos formam a população organizacional e são mapeados em classes de recursos. Em ambientes de escritório, nos quais os Sistemas de Gerenciamento de Processos de Negócio são utilizados, os recursos são principalmente pessoas. Para facilitar a alocação de itens de trabalho para recursos, os recursos são agrupados em classes. Uma classe de recurso é um grupo de recursos com características similares. Podem existir muitos recursos na mesma classe e um recurso pode ser membro de múltiplas classes. Se uma classe de recursos é baseada nas capacidades, por exemplo, nos requisitos funcionais de seus membros, é chamada *role*. Se a

classificação é baseada na estrutura da organização, tal como classe de recurso, é chamada unidade organizacional (time, departamento). A classificação do recurso descreve a estrutura da organização.

- 3 – Dados: trata-se do controle e produção dos dados. Controle de dados são dados introduzidos unicamente com propósitos de gerenciamento de processos. Controle de dados é, freqüentemente, usado para rotear decisões em *OR-split*. A produção de dados é objeto de informação (documentos, formulários e tabelas) cuja existência não depende do gerenciamento de processos de negócio.
- 4 – Tarefas: descreve as características de cada tarefa. Uma tarefa é uma unidade lógica de trabalho com características, tais como, um conjunto de operações que precisam ser realizadas: descrição, duração esperada, prioridades, disparador (tempo, recurso, mensagem) e classes de recursos requeridas (tarefas e unidades organizacionais).
- 5 – Operação: nessa perspectiva, são descritas ações elementares. Uma tarefa pode envolver muitas operações. Essas operações são executadas, freqüentemente, usando aplicações. Tipicamente, essas aplicações criam, lêem ou modificam o controle e a produção de dados em uma perspectiva de dados.

5.6. Verificação do Modelo de Processos de Negócio

Muitos Sistemas de Gerenciamento de Processos de Negócio não apóiam a verificação dos processos de negócio. Como resultado, a definição do processo torna-se operacional antes que sua corretitude seja perfeitamente verificada. Isso gera, freqüentemente, erros em tempo de execução, os quais precisam ser reparados em uma fase que proporciona altos custos. Exemplos de erros, apresentados por Aalst & Hofstede (2000), são:

- 1 – *Deadlock*, quando não é possível executar nenhuma tarefa.
- 2 – *Livelock*, quando um caso fica em *loop* infinito sendo possível executar tarefas, mas nenhum progresso é possível.
- 3 – *Deadtask*, quando uma tarefa nunca pode ser executada em nenhuma situação.

Os erros citados podem ser identificados sem nenhum conhecimento da aplicação em particular. Existem alguns erros que apenas podem ser detectados com conhecimento sobre a aplicação. Um exemplo de tais erros poderia ser o cenário em que o cliente recebe os produtos e não recebe a conta (ou recebe a conta duas vezes). Outro exemplo,

é a situação na qual as tarefas são executadas em uma ordem errada (a conta é enviada antes dos produtos).

Uma investigação compreensiva do processo de negócio na companhia pode resultar em um conjunto completo de processos. Essa é a base da introdução bem sucedida de um sistema de negócios numa empresa e, a verificação formal pode ajudar na obtenção da efetividade e eficiência desejada (Verbeek *et al*, 2001). Basicamente, existem três tipos de análise de um modelo *sound*:

- 1 – Validação: testa se o sistema comporta como esperado.
- 2 – Verificação: estabelece a corretitude de um sistema de processos de negócio.
- 3 – Análise da performance: avalia a habilidade de encontrar os requisitos com ênfase nos tempos de respostas, níveis de serviço e utilização de recursos.

A validação pode ser feita pela simulação iterativa: um número de casos fictícios é alimentado para o sistema verificar como são tratados. Para verificação e análise de performance são necessárias técnicas de análise mais avançadas.

Os Sistemas de Gerenciamento de Processos de Negócio oferecem apoio limitado para análise de performance. Muitos desses sistemas fornecem um simulador rudimentar ou uma passagem para uma ferramenta de simulação. A simulação pode ser usada para estimar indicadores chaves de performance por meio da experimentação de processos de negócio especificados sujeitos a um comportamento específico do ambiente. Exemplos de indicadores de performance são: média de tempo de resposta, média de tempo de espera, taxa de ocupação dos recursos, nível de serviço e a média de número de casos pendentes (Aalst & Hofstede, 2000).

É muito importante a verificação do modelo organizacional, uma vez que erros sérios como *deadlocks* e *livelocks* podem permanecer não detectados. Um modelo de processos de negócio incorreto proporciona um sistema incorreto que pode trazer problemas sérios para a organização. Um sistema incorreto pode levar a trabalhos desnecessários, clientes insatisfeitos, problemas gerenciais, funcionários desencorajados. Por isso, é importante verificar a corretitude do modelo antes que o sistema torne-se operacional (Verbeek *et al.*, 2001).

Figura 20 - Uma definição de modelo incorreta
 Fonte: Aalst (1999), p. 345

De acordo com Aalst (1999), o processo da figura 20 trata reclamações que entram no sistema pelo lugar início e sai do sistema por meio do lugar pronto. Inicialmente, a reclamação é registrada pela execução da tarefa registrar, e em paralelo, o questionário é enviado à pessoa que está fazendo a reclamação (tarefa enviar formulário). Depois, a reclamação é avaliada (tarefa avaliar). Se o questionário for retornado em duas semanas, a tarefa formular o processo é executada. Se o questionário não é retornado dentro de duas semanas, o resultado do questionário é descartado (tarefa tempo esgotado). Baseada no resultado da avaliação, a reclamação é processada ou não. O processamento atual da reclamação (processar reclamação) é atrasado até o formulário ser processado. O processamento da reclamação é conferido através da tarefa checar processamento. Finalmente, a tarefa arquivar é executada. Na figura 20, a definição do processo é especificada usando uma rede de Petri. A tarefa registrar, enviar formulário, processar formulário, esgotar tempo, checar processamento e arquivar foram modeladas pelas transições. As condições foram adicionadas para modelar os estados entre as tarefas. Cada condição é modelada por um lugar. Por exemplo, o lugar c2 corresponde à condição “pronto para avaliar a reclamação”. A condição c5 é verdadeira (contém uma marca) se o questionário foi processado ou o tempo esgotou. A transição registrar é chamada *AND-split*, na qual uma marca é produzida para cada um dos seus lugares de

saída. As transições avaliar e checar processamento são *OR-split*, nesse caso, precisamente, um dos lugares de saída obtém uma marca.

Uma observação mais detalhada da definição do processo da figura 20 revela muitos erros de projeto. Se uma reclamação precisa ser processada e a pessoa que está reclamando não retorna o formulário no tempo determinado, a reclamação ficará em *deadlock* com uma marca em c4 e c5. Um *deadlock* similar ocorre se a reclamação precisa ser processada em uma segunda vez. Embora o modelo apresentado na figura 20 possa ter consequências sérias, pode ser processado e se tornar operacional usando as ferramentas disponíveis no mercado que avisam sobre os potenciais *deadlocks*.

Apesar de existir muitas ferramentas para Sistemas de Gerenciamento de Processos de Negócio, a questão crítica de verificação é negligenciada. Poucas ferramentas dispõem de qualquer forma de verificação. Essa necessidade de apoio pode ser explicada pelo fato de que a verificação de processos de negócio é complicada do ponto de vista computacional e algorítmico. As consequências, de qualquer forma, são que poucos *workflows* são checados a fundo, antes que eles estejam em execução, o que frequentemente resulta em erros que serão corrigidos com custos altos (Aalst, 1999).

Especificação de modelos de negócio direciona para muitas questões incluindo fluxo de dados, tratamento de exceções entre outras. Por essa razão, a verificação de uma especificação completa, normalmente, não é viável. No entanto, tipicamente a especificação do controle de processo é o núcleo de muitas especificações. As linguagens de especificação precisam apoiar a especificação no momento de escolha, execução sequencial, paralelismo, sincronismo e iteração.

O trabalho de Aalst & Hofstede (2000) é direcionado para o problema de verificação de *workflow* na perspectiva de fluxo de controle. Nessa perspectiva, apenas a rota de casos é tratada, isso significa que habilita a execução de caminhos presentes na perspectiva de fluxo de controle. Entretanto, basta focar no *deadlock* como um resultado das restrições impostas pela perspectiva de recursos. Potenciais *deadlocks* podem existir quando:

- Múltiplas tarefas tentam alocar múltiplos recursos ao mesmo tempo.
- Existem tarefas impondo restrições em um nível de exigência que nenhum recurso é qualificado para satisfazer.

O primeiro tipo de *deadlock* pode acontecer frequentemente na manufatura flexível onde ferramentas e espaço são necessários para completar operações ocasionando problemas. Ocorre quando, em um momento qualquer, apenas um recurso está

trabalhando em uma tarefa que está sendo executada para um caso específico. Nos sistemas atuais de gerenciamento de *workflow* não é possível especificar que muitos recursos estão colaborando na execução de uma tarefa. Nesse caso, se múltiplas pessoas estão executando uma tarefa, escrevendo uma reportagem, por exemplo, apenas uma pessoa é alocada para aquela tarefa na perspectiva do sistema de gerenciamento de *workflow*. O fato de uma pessoa estar trabalhando em uma tarefa, em um momento e cada tarefa ser conseqüentemente executada por uma pessoa (embora a tarefa possa ser alocada por um grupo de pessoa) é satisfatório para checar a presença de recursos suficientes. Do ponto de vista de verificação, ou seja, análise lógica da corretitude do processo de negócio, é razoável abstrair *deadlock* desses problemas.

O segundo tipo de *deadlock* ocorre quando não existe nenhum recurso adequado para executar uma tarefa para um dado caso. Geralmente, tais problemas podem ser completamente evitados checando se todas as tarefas e expressões dos grupos produzem um conjunto vazio de recursos. De qualquer forma, podem existir erros sutis que resultaram do gerenciamento de casos (um subconjunto de tarefas para um determinado caso é requerido para ser executado pelo mesmo recurso) e separação de função (duas tarefas não são executadas pelo mesmo recurso para evitar violações de segurança). Por exemplo, a tarefa 1 deveria ser executada pela mesma pessoa que executa a tarefa 2 e a tarefa 2 deveria ser executada pela mesma pessoa que executa a tarefa 3. Entretanto, a tarefa 3 não deveria ser executada pela pessoa que executou a tarefa 1. Nesse caso, não existe nenhuma pessoa qualificada para executar a tarefa 3. Tais problemas dependem muito do sistema de gerenciamento do *workflow* que está sendo usado e são independentes da estrutura de rota.

Entretanto, de acordo com Aalst & Hofstede (2000), é possível abstrair *deadlocks* das seguintes perspectivas:

- Perspectiva de dados. Os dados estão parcialmente fora do escopo do sistema de gerenciamento de processos de negócio. Esses dados podem ser modificados em qualquer momento sem notificar o sistema de gerenciamento de processos de negócio. Na verdade, sua existência não depende igualmente da aplicação do sistema e ainda, os dados podem ser compartilhados entre diferentes processos do sistema. Alguns desses dados são atualizados por humanos ou por aplicações. Por exemplo, a decisão tomada por um gerente é baseada na intuição ou um caso é classificado baseado em cálculos complexos envolvendo produção de dados. Claramente, o comportamento de um humano ou de aplicações complexas não

pode ser modelado completamente. Então, alguma abstração é necessária para incorporar a perspectiva de dados quando se verifica um *workflow*. A abstração usada em Aalst & Hofstede (2000) segue a seguinte linha: uma vez que o controle de dados é utilizado, normalmente, apenas para planejar o melhor caminho de um caso, eles incorporaram a ação de planejar um melhor caminho de decisão. Dessa forma, as redes de Petri ordinárias são utilizadas em vez de redes de alto nível. Do ponto de vista analítico, é preferível utilizar redes de Petri ordinárias por causa da disponibilidade de ferramentas de análise.

- Perspectivas de tarefa e operação. As tarefas são consideradas atômicas e abstratas da execução de operações dentro das tarefas. O sistema de gerenciamento de *workflow* pode apenas iniciar aplicações, atividade de pessoas e monitoração dos resultados, sem poder controlar a execução atual da tarefa. Entretanto, do ponto de vista de verificação, é razoável focar na perspectiva de fluxo de controle. Na verdade, é o suficiente para considerar o ciclo de vida de um caso isolado.

5.7. Análise Qualitativa

A análise qualitativa, de acordo com Salimifard & Wright (2001), concentra nas propriedades, tais como: ausência de *deadlock*, fluxo correto de trabalho, ausência de *overflow*, e a presença de exclusão mutual em qualquer lugar que os recursos sejam divididos. O principal objetivo da análise qualitativa é provar que o modelo é válido como também encontrar respostas para algumas questões, tais como:

- Existe algum *deadlock* no sistema?
- A seqüência do estado é correta?
- O processo termina corretamente?
- Existe algum conflito não resolvível para acessar recursos entre atividades diferentes?

A análise qualitativa tem sido o principal assunto dos trabalhos na literatura. Alguns autores têm aplicado com sucesso redes de Petri, principalmente concentrando na modelagem estrutural e análise das propriedades qualitativas do processo de negócio. Em Adam *et al.* (1998), uma versão modificada de redes de Petri ordinárias, chamada *Temporal Constraint Petri Net* (TCPN), foi usada para modelar aplicações de modelos de negócio em um nível conceitual como um teste para verificar as propriedades do

sistema. Cada tarefa é decomposta em algumas primitivas, e então os estados entre as primitivas podem ser representados. Dependências de tarefas podem ser categorizadas em dependência do fluxo de controle, dependência de valor e dependência externa. Uma verificação lógica é usada para verificar as principais propriedades do modelo de processos de negócio.

Uma rede de Petri que modela a dimensão do fluxo de controle é chamada de *workflow-net* (WF-net). Em Aalst & Hee (1996) e em Aalst & Hee (1995), foram utilizadas redes de Petri de alto nível para modelar um processo de negócio. O modelo determina qual, como e por quem o processo é realizado. Em Aalst (1995) foi utilizado um tipo de rede escolha-livre de Desel & Esparza (1995), conhecida por Rede de Procedimento de Negócio (BP-net). Essa rede propõe uma representação, validação e verificação dos processos de negócio. Esse modelo foi modificado dentro de uma *workflow-net* (WF-net) como um modelo *sound* para processos de negócio, mostrando que a WF-net é bem estruturada. Uma rede hierárquica é proposta para modelar processos complicados. Kindler (1999) apresentou que a propriedade de *soundness*, proposta em Aalst (1996) não é suficiente para garantir a corretude do modelo de processo. Nesse sentido foi adicionado à propriedade safe-1 a WF-net para obter um modelo de processo de negócio *sound*.

Verificação do processo de negócio é o ponto focal de Voorhoeve (2000). O processo é representado como um modelo de rede de Petri. Uma técnica combinando redes de Petri e álgebra de processo é usada para a verificação formal do processo.

Redes de Petri de alto nível foram sugeridas em Aalst *et al.* (1994), nas quais uma atividade de controle é usada para rotear trabalhos (*work-items*) dentro do procedimento, sincronizar o trabalho e duplicar *tokens-job*. O comportamento do processo de negócio tem sido analisado usando redes de Petri ordinárias. O processo é definido como uma composição de quatro entidades básicas. Cada entidade do processo é modelada como uma rede de Petri ordinária. Essas redes têm sido combinadas para obter uma rede de Petri global baseada no modelo de processo.

Em Atluri & Huang (1996), foi proposta uma representação de um modelo de autorização de processo de negócio (WAM- *workflow authorisation model*). Esse modelo assegura que em um sistema de gerenciamento de processos de negócio, os recursos autorizados são privilégios reconhecidos nas tarefas requeridas apenas enquanto a tarefa está sendo executada. Para garantir que os recursos certos estivessem liberados no tempo certo, uma rede de Petri colorida foi usada para sincronizar a

autorização com o fluxo de trabalho. Exceções foram consideradas em Ellis & Nutt (1993) e outro tipo de redes de Petri de alto nível, nomeada ICN, foi usado para modelar procedimentos independentes da sua implementação.

Processos de negócio interorganizacionais e a disponibilidade da Internet têm afetado a tecnologia de processos de negócio. Um sistema de gerenciamento de processo de negócio baseado na Internet tem sido considerado em Weitz (1998). Uma rede SGML (Standard Generalized Markup Language), uma variante de redes de Petri de alto nível, foi proposta para capturar o processo de geração e manipulação de documentos estruturados como uma camada de dados. Marcas são instâncias de documentos com uma estrutura interior. Cada lugar é associado com um tipo de documento SGML representando o tipo do lugar. Uma marca de um lugar é um conjunto de instâncias de documentos reabilitados no lugar. Cada arco é inscrito com um chamado documento *template*, o qual é usado para manipular instâncias do documento no lugar adjacente ao arco. A regra de disparo para a SGML é modificada para prevenir a violação entre os respectivos lugares.

Uma rede de Petri baseada em processos de negócio pode ser usada no ambiente de distribuição de acordo com Aalst (1999b). Para checar a corretitude de processos de negócio interorganizacionais, a noção de *soundness* global baseada na propriedade *soundness* de um modelo de processos de negócio (Aalst (1997); Aalst (2000)) é introduzida. O modelo é baseado na mensagem que passa entre os modelos de negócio e uma seqüência de mensagem do diagrama é usada para conectar diferentes processos de negócio. Kindler *et al.* (2000) enfatizaram que algumas vezes não é possível usar a técnica introduzida em Aalst (1999) para checar a *soundness* global de modelos de processos interorganizacionais. Kindler *et al.* (2000) propuseram uma técnica algébrica baseada em redes de Petri para automaticamente analisar um simples processo de negócio isolado. Essa técnica checa *soundness* local do modelo de processo, usando uma constituição (*composition*) do teorema, assim garante global *soundness* do modelo de processo.

Em Guth *et al.* (1998), foi considerada a fragmentação do modelo de processos de negócio. Nesse caso, foram usadas redes de Petri de alto nível (rede predicado/transição), nas quais os lugares representam esquemas de relação entre fragmentos, e as transições representam uma classe de operações nas relações dos lugares adjacentes. A fragmentação foi modelada e interpretada pelo Sistema de Gerenciamento de Processos de Negócio com sucesso. Um Sistema de Processos de

Negócio pode ser movido entre diferentes departamentos e organizações usando tipos diferentes de ferramentas.

Merz *et al.* (1995) integraram um modelo de processos de negócio a uma rede de Petri colorida baseada no modelo do *Common Open Service Market* (COSM), o qual é um ambiente distribuído. CPN foi usada para especificação formal de um processo de negócio e gerenciamento e controle de atividade concorrente. A simulação foi usada para verificar a coordenação entre as atividades do negócio em uma arquitetura distribuída.

Em Sistemas de Gerenciamento de Processos de Negócio atuais, é difícil lidar com mudanças radicais e *ad hoc*. Como uma solução para o problema, foi proposta por Badouel (1998) uma *WF-net* reconfigurável que é uma extensão da *WF-net* introduzida em Aalst (1997). Essa rede modela a modificação da configuração de um processo de negócio, e descreve a estrutura do processo por algum tipo de modo de operação predefinido. Dessa forma, é possível modificar, dinamicamente, sua configuração por reescrever alguns dos seus lugares. A rede proposta é considerada apenas em casos individuais, ou seja, um processamento concorrente de múltiplos casos não pode ser modelado. Em Voorhoeve (1997) e em Aalst (1999), a noção de sistemas de processos de negócio adaptativo é introduzida e dois tipos de mudanças, *ad hoc* e evolutiva, foram direcionadas. Mudanças *ad hoc* são relacionadas a eventos raros, os quais são baseados em casos, enquanto mudanças evolutivas são, freqüentemente, baseadas nos resultados da reengenharia do sistema. Para tratar essas mudanças foram propostos modelos de processos genéricos, os quais são ilustrados por redes de Petri.

Alguns autores têm concentrado na modelagem de processos de negócio orientada a objetos e redes de Petri. Wikarski (1996) introduziu a noção de Redes de Petri Modular para a representação formal de processos de negócio. Essa é uma rede de baixo nível combinada com uma abordagem orientada a objeto que permite muitos tipos de abstração. O autor apresentou, também, a noção de sensor de transição para ser capaz de modelar a comunicação entre redes diferentes. O gerenciamento de recursos é explicitamente considerado em Lowe *et al.* (1995) e Han *et al.* (1994). Nesses trabalhos, foi proposta uma extensão orientada a objeto de Redes de Petri chamada *Higher-Order Object Net* (HOON) para modelar sistemas de processo de negócio e seus recursos relacionados como modelos de objetos diferentes. Todos os modelos de objetos estão interagindo em um ambiente cliente/servidor distribuído.

Em Moldt & Valk (2000) é apresentada uma abordagem orientada a objeto baseada em CPN para modelar processos de negócio chamada Redes de Petri de Processos de Negócio (BPP-nets). Um sistema contendo processos e recursos é mapeado dentro de unidades funcionais e modelado como objetos. Métodos são representados por transições e objetos comunicam-se por meio de algumas interfaces. Uma marca em BPP-net pode ter uma estrutura interna complexa. BPP-nets apóiam adaptação dinâmica do modelo, importante para tratar a mudança dos processos de negócio.

Aalst *et al.* (2000) iniciaram um repositório para padrões de modelagem de processos de negócio. Todos os padrões possíveis do modelo de processos de negócio incluem padrões de: controle básico, ramificações e sincronizações, estrutura, estado e de cancelamento. Quatro padrões de sistemas de processos de negócios são considerados em Aalst *et al.* (2000):

- a) União da sincronização.
- b) Terminação implícita.
- c) Múltiplas instâncias requerendo a sincronização.
- d) Escolha adiada.

Esses padrões foram descritos independentemente das linguagens do sistema corrente. Baseado nesses padrões selecionados algumas ferramentas foram avaliadas e nenhuma apóia todos os padrões. Assim, como resultado, foram indicados novos requisitos para linguagens de Sistemas de Gerenciamento de Negócios. O padrão proposto pode ser usado como uma lista de verificação (*checklist*) na seleção do produto.

5.8. Análise Quantitativa

A análise quantitativa consiste em calcular os índices de performance, responsabilidade e utilização de recursos. Apesar da importância das propriedades quantitativas no sistema de processos de negócio, elas não têm despertado a atenção dos pesquisadores comparadas aos trabalhos que são direcionados na análise qualitativa (Salimifard & Wright, 2001).

Um sistema com a funcionalidade aprovada não garante que o sistema irá satisfazer os requisitos do cliente. Essas questões devem ser respondidas antes que o sistema seja instalado (Salimifard & Wright, 2001):

- a) O processo é terminado em um tempo predefinido?

- b) Qual é a utilização do recurso para o processo?
- c) Qual é o tempo que uma determinada atividade deve esperar antes de iniciar o processamento?

Algumas extensões de Redes de Petri, nas quais a noção de tempo é considerada, estão disponíveis para análise. A rede de Petri estocástica generalizada (GSPN), entre outros tipos de redes de Petri é umas das que mais tem sido usadas. Em Shomig & Rau (1995), redes de Petri de alto nível foram utilizadas para modelar um processo de negócio. O processo foi mapeado dentro de uma GSPN. Uma simples GSPN, que é conhecida como modelo *Load Equivalence Aggregation* (LEA), foi desenvolvida para avaliar a performance dos recursos humanos. Assim, o modelo é simulado usando uma GSPN colorida (CGSPN).

Uma representação da GSPN de um modelo de processos de negócio foi proposta em Ferscha (1994). Cada elemento do processo é descrito por uma transição simples e uma GSPN representa a relação precedente entre elementos do processo. Um método baseado em um tempo contínuo da cadeia de Markov (CTMC) foi usado para obter limites superiores da execução da performance. Em Adam *et al.* (1998), cada lugar e transição são associados com um intervalo de tempo para denotar os limites inferiores e superiores de tempo de espera para lugares e transições associados. Marcas são associadas a um tempo que representa tempo de chegada da marca em cada lugar.

Em Desel & Erwin (2000) é apresentada uma abordagem baseada em redes lugar/transição para modelagem de processos de negócio. Para estudar a performance do processo, o modelo lugar/transição é melhorado com a notação de tempo e custo. O modelo resultante é simulado e algumas medidas são obtidas relacionadas em termos de tempo e custos.

A análise da performance do processo de negócio foi considerada em Hee (2000). Todas as construções das rotas do processo são modeladas em redes de Petri (alto nível) estocásticas. A duração de tempo que pode seguir qualquer distribuição arbitrária é associada com as transições representando atividades. Assumindo uma agenda de recursos liberados e uma infinita disponibilidade de recursos, o tempo de resposta do processo é analiticamente computado.

Em adição a análise de redes de Petri quantitativa, a técnica de simulação é uma alternativa, ou seja, uma abordagem que integra modelagem de simulação e análise de capacidades dentro de Sistemas de Gerenciamento de Processos de Negócio. A

simulação de modelos é usada para estudar a eficiência de projetos, além do estudo de performance geral e confiabilidade do Sistema de Gerenciamento de Processos de Negócio.

Muitos dos trabalhos encontrados na literatura têm usado versões diferentes de redes de Petri de alto nível e a maioria tem se concentrado na modelagem e na análise qualitativa do processo de negócio. Propriedades quantitativas de processo de negócio, tais como índices de performance e responsabilidade, são raramente consideradas.

A ferramenta de modelagem deveria ser escolhida de acordo com o problema. Para análise estrutural do processo de negócio, redes de Petri ordinárias são indicadas por serem apoiadas pelos métodos analíticos. Análise comportamental do sistema necessita de uma versão modificada de redes de Petri de alto nível. Extensões de redes de Petri coloridas (CPN) temporizadas são necessárias para representar um processo de negócio com detalhes requeridos, por serem capazes de calcular medidas quantitativas relacionadas ao comportamento dinâmico do processo. Modelos de processos de negócio que usam ferramentas disponíveis não têm nenhum benefício prático para o Sistema de Gerenciamento de Processos de Negócio se não houver uma ligação para um ambiente de execução. Redes de Petri baseadas nas ferramentas de modelagem deveriam ser inseridas no sistema de gerenciamento de Processos de Negócio, se o projetista desejar usar as vantagens disponíveis de redes de Petri.

5.9. Exemplo de modelo de processos de negócio utilizando redes de Petri (com parâmetros de tempo e custos)

Salimifard & Wright (2001) apresentam um exemplo da aplicação de redes de Petri na modelagem de processos de negócio. No caso, o processo é de uma companhia de seguros e consiste de cinco tarefas: submissão de pedido, checagem do seguro, contato com a garagem, envio de carta (resposta negativa ao pedido) e pagamento do dano (resposta positiva). O modelo em rede de Petri desse processo de negócio está na figura 21.

Figura 21 - Processo de uma companhia de seguros incluindo as informações tempo e custo
Fonte: Adaptado de Salimifard & Wright (2001).

No modelo, cada tarefa do processo está representada por uma transição e assume-se que duas transições *contatar garagem* e *checar seguro* podem disparar simultaneamente. Em contrapartida, existem duas transições *iniciar* e *consultar situação* que modelam *split* e *join* para representar execuções de tarefas paralelas.

Existe uma marca em *pronto para iniciar*, então a transição *submeter pedido* está habilitada. O disparo da transição *submeter pedido* coloca uma marca no lugar *pedido submetido*. Na seqüência a transição *iniciar* é habilitada, e quando ela é disparada, uma marca é adicionada nos lugares *processo de garagem* e *processo de seguro*. As transições *contatar garagem* e *checar seguro* são disparadas ao mesmo tempo. Esse estado representa a possibilidade de execução paralela de duas tarefas modeladas por essas transições. A transição *consultar situação* é habilitada assim que houver uma marca em *contato efetuado* e em *situação da apólice efetuada*. Isso garante a execução correta de tarefas paralelas. Quando a transição *consultar situação*, uma marca será produzida no lugar *pronto para executar ação*, a qual habilitará a transição *pagar dano* e *enviar carta*. Essas duas transições estão em conflito e apenas uma delas pode disparar.

O disparo de uma delas consome uma marca do lugar pronto para executar uma ação e produz uma marca no lugar fim do processo.

Resumindo, as tarefas checar seguro e contatar garagem (habilitadas após o disparo de t_2) são independentes e podem ocorrer, conjuntamente, enquanto que as tarefas que vêm a seguir (pagamento do dano e envio de carta) são mutuamente exclusivas.

No fim do processo, o tempo total é fornecido pela soma de todos os fatores de tempo e o custo total pela soma dos fatores de custo. Na transição cada fator de tempo e custo é apresentado pela porcentagem do custo total de cada transição.

No modelo de processos de negócio, o parâmetro de tempo (e igualmente custo) é consumido entre as transições de disparo. Em uma abordagem comum, esse consumo ocorre entre os disparos. A diferença ocorre, justamente, por causa da transição da atividade e utilização de lugares internos. Estudos sobre o parâmetro de custos e redes de Petri podem ser encontrados em Silva (2000).

5.10. Árvore de Alcançabilidade

A árvore de alcançabilidade é fundamental para o estudo de propriedades dinâmicas de qualquer sistema. Por meio da árvore de alcançabilidade é possível deduzir o total de casos atingíveis e verificar o número típico de erros que pode ocorrer na definição de processo (Murata, 1989).

O disparo de transições habilitadas altera as marcações da rede, ou seja, uma sequência de disparos resulta em uma sequência de marcações. A árvore de alcançabilidade é utilizada como base para o estudo das propriedades dependentes de marcação inicial do modelo como, por exemplo, se ele é seguro, livre de *deadlocks*, limitado etc. A validação do modelo, tarefa importante na modelagem, pode ser obtida com embasamento na formalidade matemática da rede de Petri.

Por meio da árvore de alcançabilidade (iniciando com apenas uma marca no lugar início) é possível verificar a estrutura do processo checando os três requisitos de *soundness*. Os dois primeiros requisitos são checados, confirmando se na árvore de alcançabilidade tem apenas um estado final e que ao final do caso deve existir apenas uma marca no lugar fim. Para checar o último requisito, é necessário verificar se para cada tarefa existe um estado correspondente na árvore de alcançabilidade (Aalst & Hee, 2002).

Existem duas desvantagens nessa abordagem. Primeiro, a construção da árvore de alcançabilidade para processos em grande escala requer tempo computacional relativamente grande. Segundo, a árvore de alcançabilidade fornece pouco apoio no reparo de processos não *sound*. A árvore de alcançabilidade é infinita se houver a possibilidade de marcas ficarem acumuladas em um lugar. Nesse caso, variantes da árvore de alcançabilidade podem ser utilizadas, como a árvore de cobertura que permite detectar tal comportamento ilimitado.

5.11. Considerações Finais

Empresas que possuem um Sistema de *Workflow* bem elaborado, oferecem maior qualidade em seus serviços, maior padronização, mais agilidade, cometem menos erros, e portanto, são mais eficazes e eficientes na execução das atividades gerenciadas por esse sistema. Aperfeiçoamentos nesses processos também são mais fáceis e mais ágeis.

A área de sistemas de *workflow* encontra-se em evidência atualmente. A modelagem de *workflow* consiste em representar, por meio de um formalismo, as atividades que compõem os processos, sua seqüência de execução e seus inter-relacionamentos, assim como os agentes responsáveis por sua execução e os recursos envolvidos. É nesse modelo que o Sistema Gerenciador de *Workflow* baseia se para coordenar a execução das atividades.

A tecnologia de gerenciamento *workflow* procura oferecer uma solução flexível no apoio aos processos de negócios, por meio da facilitação de modificações e da criação de novos processos.

As redes de Petri permitem a participação de diferentes especialistas envolvidos na atividade de modelagem processos de negócio e possibilita troca de informações padronizadas entre os mesmos nas diferentes fases do projeto. Em contrapartida, o formalismo a torna uma técnica importante de modelagem na representação dos processos, permitindo a exibição de: concorrência, paralelismo, sincronização, não-determinismo e exclusão mútua. Nesse sentido, a grande conveniência no uso de redes de Petri, na modelagem de processos de negócios é possibilitar um rastreamento minucioso e não-ambíguo de cada etapa da operação. No capítulo 6 são apresentados a formalização do Modelo de Processos de Negócio do EKD, o mapeamento do Modelo

de Processos de Negócios formal em redes de Petri e o método de avaliação do modelo de processos de negócio, todos criados neste trabalho.

CAPÍTULO 6

Desenvolvimento do método de avaliação do Modelo de Processos de Negócio

6.1. Considerações Iniciais

A utilização de redes de Petri fornece uma definição precisa. Ambigüidades, incertezas e contradições são, assim, prevenidas, em contraste a métodos de diagrama informal como todas as técnicas de modelagem apresentadas no capítulo 3. O formalismo pode ser usado para argumentar sobre o processo e estabelecer alguns padrões. Essa idéia está fortemente ligada ao fato de que o formalismo freqüentemente habilita o uso de um número de técnicas analíticas que podem ser utilizadas para a análise de desempenho (eficácia e eficiência do sistema a ser estudado) e para a verificação de propriedades, possibilitando inclusive determinar todas as situações desejadas e principalmente as indesejadas de modo que sejam evitadas no sistema real. No capítulo 5 foram apresentados vários trabalhos em que processos de negócio são modelados diretamente em redes de Petri, sem a utilização de um modelo organizacional. Realizar a modelagem organizacional trás muitos benefícios como afirmado anteriormente. Este capítulo tem a finalidade de apresentar a formalização do Modelo de Processos de Negócio do EKD, o procedimento de mapeamento do Modelo de Processos de Negócio em Redes de Petri e o método de avaliação do modelo de processos de negócio, todos desenvolvidos neste trabalho.

6.2. Modelo de Processo de Negócio

O Modelo de Processo de Negócio destina-se a analisar o processo e o fluxo de informação e de material da organização. Processos podem ser decompostos em sub-processos.

O Modelo de Processos de Negócio descreve as atividades organizacionais (funções e processos da organização). O núcleo da organização é o conjunto de processos, contribuindo para os valores da organização. Para alcançar uma boa abstração, o Modelo de Processos de Negócio permite total liberdade no processo de decomposição em sub-processos (figura 22 e 23). Dependendo do propósito da modelagem, os processos descritos podem ser existentes ou futuros (processos planejados). Um exemplo de um processo de negócio é o "Gerenciamento de retornos de empréstimos" que pode ser decomposto em "registro de retorno", "checar se o livro está reservado" e "retornar à prateleira".

Os componentes do Modelo de Processos de Negócio são processos internos (ou simplesmente processos), processos externos e conjunto de materiais e de informações (*inf-set*). Processo é uma coleção de atividades que:

- Consomem entradas e produzem saídas em termos de informação ou material.
- São controladas por um conjunto de regras, indicando como processar as entradas e produzir as saídas.
- Têm um relacionamento com o Modelo de Atores e Recursos, em termos do executor ou responsável pelo processo.
- Como uma instância do Modelo de Processos de Negócio é esperado para consumir, quando iniciado, um montante finito de recursos e tempo.

Processos externos representam uma coleção de atividades que:

- São localizadas fora do escopo da área da atividade organizacional.
- Comunicam-se com processos ou atividades da área do domínio do problema.
- São essenciais para documentar.

Processos externos, algumas vezes, podem ser considerados como fluxo de entrada ou saída de alguma informação e material. Um exemplo típico de processo externo pode ser o de um cliente que pede ou recebe algum serviço da Biblioteca. Conjunto de

material ou informação (*inf-set*) é um conjunto de informação ou material enviado de um Processo ou Processo Externo para outro. Os conteúdos do fluxo de material e informação são descritos no Modelo de Conceitos, no qual são decompostos se necessário. O fluxo de material e informação deve ter ao menos um envio de processos ou processos externos e ao menos um recebimento de processo ou processo externo. Os processos de alto nível deveriam ser separados dos processos de baixo nível com o mecanismo de decomposição. Igualmente, se não existir nenhum nível máximo de intensidade de decomposição, a restrição é desejada pois evita estruturas complexas desnecessárias.

Um exemplo de decomposição é apresentado na figura 22, sendo o processo de verificação que tem como entrada "Endereço" e duas saídas "Endereço Inválido" e "Endereço Válido" (figura 22). Na figura 23 o processo é decomposto em quatro sub-processos. Cada um desses sub-processos tem um fluxo de dados de entrada. No exemplo considera-se que "Endereço" é um fluxo de informação complexa sendo, então, dividido em vários componentes. No caso de uma biblioteca, por exemplo, existem vários processos como: emprestar e devolver livros, recebimento, entre outros. Todos eles podem ser decompostos.

Figura 22 - Exemplo de um processo verificação do endereço do cliente
Fonte: Bubenko *et al.*(1998), p. 51

Os componentes do Modelo de Processos de Negócio devem ser motivados pelos objetivos da organização definidos no Modelo de Objetivos. Processos do Modelo de Processos de Negócio são realizados com a informação descrita pelos componentes do Modelo de Conceitos, tais como entidades, atributos e relacionamentos ou grupos de componentes do Modelo de Conceitos. Os componentes do Modelo de Processos de Negócio também estão intimamente relacionados com todos componentes do Modelo de Atores e Recursos. O relacionamento entre o Modelo de Processo de Negócio e Modelo de Atores e Recursos pode ser de diferentes tipos, como segue:

- Ator A realiza processo P.

- Ator A é responsável por processo P.
- Ator A apóia processo P.

Figura 23- O processo verificação de endereço é decomposto em quatro sub-processos
 Fonte: Bubenko *et al.* (1998) p. 51

Cada componente do Modelo de Processos de Negócio deve, ao menos no nível de decomposição, ter um relacionamento definido no Modelo de Atores e Recursos. É importante observar que o Modelo de Processos de Negócio descreve processos da área do negócio, e não unidades organizacionais e sistema. Para um componente ser qualificado para inclusão no Modelo de Processos de Negócio, deve-se descrever um conjunto de possíveis processos com um tempo de início e de fim. Em um alto nível de abstração, esse conjunto de processos pode ser razoavelmente bem definido.

6.3. Mapeando os conceitos de Processos de Negócio para Redes de Petri

Uma grande vantagem do Modelo Processos de Negócio a ser ressaltada é a facilidade de uso. A notação utilizada tem como principal objetivo ser compreendida por todos usuários, sem requerer muito treinamento. O alto custo de manutenção dos sistemas de informação é um fator importante para incentivar o entendimento do negócio e derivar

os requisitos desse entendimento para que o sistema atenda realmente as necessidades do negócio. Por esse motivo o Modelo Organizacional é utilizado para capturar todos os requisitos organizacionais e modelar os processos de negócio ao invés de modelar os processos de negócio diretamente em redes de Petri.

Apesar do método de modelagem organizacional EKD dispor de um conjunto de diretrizes para sua introdução e aplicação, ambigüidades e incertezas não podem ser prevenidas uma vez que o processo de modelagem não é rigoroso.

Em contrapartida, as redes de Petri exigem uma definição precisa, o que habilita o uso de muitas técnicas analíticas que podem ser utilizadas para a análise de desempenho e para a verificação de propriedades lógicas.

Como afirmado anteriormente, a rede de Petri demonstra uma complementação ao Modelo de Processos de Negócio por possibilitar que haja rastreamento passo a passo, a cada etapa da operação, sem ambigüidade, por ter uma representação matemática e disponibilizar mecanismos de análise que permitem a verificação de propriedades e aumentar a corretitude do sistema. As redes de Petri podem ser utilizadas para modelar sistemas paralelos, concorrentes, assíncronos e não-determinísticos.

No capítulo 5 foi afirmado que é muito importante a verificação do modelo organizacional, uma vez que erros sérios como *deadlock* e *livelocks* podem permanecer indetectados. Um Modelo de Processos de Negócios incorreto proporciona um sistema incorreto que pode levar a trabalhos desnecessários, clientes insatisfeitos, problemas gerenciais, funcionários desencorajados. Muitas vezes a definição do processo se torna operacional antes que sua corretitude seja perfeitamente verificada. Isso gera freqüentemente erros em tempo de execução, os quais precisam ser reparados em uma fase que proporciona altos custos.

Um modelo de processos de negócio mapeado em redes de Petri fornecerá uma firme fundamentação lógica e contexto realístico nas questões de gerenciamento do negócio, o que permite a aplicação de algoritmos provenientes de: matemática aplicada, teoria da decisão, teoria de jogos, pesquisa de operação, estatística, entre outros.

O processo define quais tarefas precisam ser executadas. É importante que existam informações sobre tarefas a serem realizadas e sobre as condições dos processos. Dessa forma, é definida a ordem na qual as tarefas precisam ser executadas. Um grande processo pode consistir de sub-processos, tarefas e condições.

Para especificar um processo usando redes de Petri são necessárias uma entrada (um lugar sem arcos de entrada) e uma saída (um lugar sem arcos de saída).

Como já mencionado anteriormente, *soundness* é um critério de verificação de correção definido para *workflow-nets*. Uma *workflow-net* é *sound*, se em qualquer caso, o procedimento termina com uma marca no lugar fim e com todos os outros lugares vazios. Além disso, não deve haver nenhuma transição *dead* (morta) e todas as tarefas devem ser passíveis de serem executadas por meio de uma rota apropriada (Aalst & Hee, 2002).

A definição de *soundness* está relacionada com a noção de *fairness*, ou seja, se uma tarefa pode potencialmente ser executada, então não é possível atrasar sua execução indefinidamente. Na rota iterativa, em princípio, é possível repetir uma parte do processo infinitamente, considera-se que a iteração não necessariamente viola o requisito *soundness*. Similarmente, assume-se que duas tarefas não podem impossibilitar que uma tarefa execute por tempo indeterminado. Se não for feita essa suposição, qualquer processo com rota seletiva ou iterativa não seria *sound* (Verbeek *et al*, 2001).

Esse requisito da propriedade *soundness* que exige que todos os lugares, exceto o lugar fim, estejam vazios ao final do procedimento é para indicar que todas as tarefas foram executadas, uma vez que as marcas representam os elementos que estão sendo analisados no processo.

Em redes de Petri, uma subrede inteira pode ser substituída por um simples lugar ou transição para a modelagem num nível de abstração maior. Por outro lado, também podemos substituir um lugar ou uma transição por uma subrede para obter um modelo mais detalhado, desenvolvido por meio de blocos bem formados.

6.4. Formalização dos Conceitos do Modelo de Processos de Negócio

Os métodos de modelagem organizacional encontrados na literatura são técnicas de diagrama informal e a sintaxe e a semântica não são bem definidas. Os modelos resultantes podem ser ambíguos. Além disso, é impossível checar a consistência e completude. Isso impede o uso de ferramentas analíticas. Para que seja possível realizar o mapeamento do Modelo de Processos de Negócio em Redes de Petri foi criada neste

trabalho uma definição formal do Modelo de Processos de Negócio do EKD (MPN-EKD). Dessa forma será possível descrever os requisitos que um Modelo de Processos de Negócio deve satisfazer para que o mapeamento seja desenvolvido.

Para essa definição formal do Modelo de Processos de Negócio foi criado um conjunto de conectores para o Modelo de Processos de Negócio do EKD. O conjunto de conectores é representado por C e é composto por C_{AND} , C_{OR} , C_J , C_S , C_{IP} e C_{PI} . Os conectores C_{OR} e C_{AND} foram criados para identificar escolha (exclusiva) e paralelismo para que os casos de paralelismo e escolha não sejam modelados exatamente da mesma forma, criando ambigüidades e dificuldades de compreensão. Os conectores C_J e C_S definem conectores do tipo *join* e *split*. Para descrever a natureza do fluxo dos processos e de suas interações existe um conjunto de termos, utilizados em *Workflow Management Coalition* (1996) e Aalst & Hee (2002), que são apresentados a seguir:

- *AND-Split*: ponto em que, de uma única linha de fluxo, partem duas ou mais linhas que são executadas em paralelo.
- *AND-Join*: ponto em que duas ou mais atividades, executando em paralelo, convergem em uma única linha de fluxo comum.
- *OR-Split*: ponto em que uma única linha de fluxo faz uma decisão entre um número de opções.
- *OR-Join*: ponto no qual uma atividade que possui um número de alternativas direciona-se para uma única opção.

De acordo com essas definições de *AND-Split*, *AND-Join*, *OR-Split* e *OR-Join*, as construções da figura 24 não são permitidas no MPN-EKD formal.

Figura 24 - Construções que não são permitidas em um MPN-EKD formal

Neste trabalho foi desenvolvida uma estratégia para tratar essas construções que não são permitidas de acordo com a definição apresentada de *AND-Split*, *AND-Join*, *OR-Split* e *OR-Join*. Essa estratégia baseia-se na introdução de um *inf-set* adicional e um processo adicional (figura 25). Essa estratégia deve ser usada durante a construção do MPN-EKD formal.

Figura 25 - Estratégia para tratar as construções que não são permitidas

Os conectores C_{IP} e C_{PI} demonstram que um conector C é um caminho de um *inf-set* para um processo ou um caminho de um processo para um *inf-set*. Um conector não poderá ligar-se a outro conector. De acordo com essa definição, as construções da figura 26 não são permitidas. Para tratar essas construções foi desenvolvida neste trabalho uma estratégia que também se baseia na introdução de um *inf-set* adicional e um processo adicional. A estratégia é apresentada na figura 27. Essa estratégia deve ser usada durante a construção do MPN-EKD formal.

Figura 26 - Construções que não são permitidas de acordo com as definições de conectores

Figura 27 - Estratégia para tratar conectores ligados a conectores

Os conectores C_{IP} e C_{PI} demonstram que um conector C é um caminho de um *inf-set* para um processo ou um caminho de um processo para um *inf-set*. Os estados inicial e final não são especificados no Modelo de Processos de Negócio do EKD, sendo necessário criar esses estados para que a formalização seja efetivamente realizada.

Algumas construções não são permitidas de acordo com as definições do MPN-EKD que serão apresentadas a seguir. As situações em que essas construções parecem necessárias devem ser analisadas cuidadosamente, sendo necessário o discernimento da equipe ou pessoa que está modelando para tratar essas construções de forma que o

modelo seja construído de acordo com as definições apresentadas. Não serão apresentadas todas essas situações por não fazer parte do escopo deste trabalho.

Definição 1. Um MPN-EKD é uma quintupla (I, P, C, Q, A) :

- I é um conjunto finito de *inf-set* (conjunto de informações),
- P é um conjunto finito de processos,
- C é um conjunto finito de conectores lógicos,
- $Q \in C \rightarrow \{AND, OR\}$ é uma função que mapeia cada conector dentro de um tipo de conector,
- $A \subseteq (I \times P) \cup (P \times I) \cup (I \times C) \cup (C \times I) \cup (P \times C) \cup (C \times P)$ é um conjunto de arcos

Um MPN-EKD é composto por três tipos de elementos: *inf-set* – conjunto de informações (I), processos (P) e conectores (C). O tipo de cada conector é dado pela função Q : $Q(c)$ é o tipo (AND ou OR) de um conector $c \in C$. A relação A especifica um conjunto de arcos conectando processos, conjunto de informações (*inf-set*) e conectores. A definição 1 demonstra que não é permitido ter um arco conectando dois processos ou dois *inf-sets* ou dois conectores.

Definição 2. Um caminho direcionado p de um nodo n_1 para um nodo n_K é uma seqüência $\langle n_1, n_2, \dots, n_K \rangle$, tal que $\langle n_i, n_{i+1} \rangle \in A$ para $1 \leq i \leq k - 1$. p é elementar se, e somente se, para qualquer um dos nodos n_i e n_j em p , $i \neq j \rightarrow n_i \neq n_j$.

A definição de caminho direcionado será usada para limitar o conjunto de construções de rotas que podem ser usadas. Essa definição permite a definição de C_{IP} (conjunto de conectores de um *inf-set* para um processo) e C_{PI} (conjunto de conectores de um processo para um *inf-set*). C_{IP} e C_{PI} divide o conjunto de conectores C . Baseado na função Q , o C é particionado em C_{AND} e C_{OR} . O conjunto C_J e C_S é usado para classificar os conectores em conectores *join* ou *split*.

Definição 3. Seja $MPN-EKD = (I, P, C, Q, A)$ um:

- $N = I \cup P \cup C$ é um conjunto de nodos do MPN-EKD.
- $C_{AND} = \{c \in C \mid Q(c) = AND\}$.
- $C_{OR} = \{c \in C \mid Q(c) = OR\}$.

- Para $n \in N$: $\bullet n = \{m \mid (m,n) \in A\}$ é o conjunto de nodos de entrada, e $n\bullet = \{m \mid (n,m) \in A\}$ é um conjunto de nodos de saída.
- $C_J = \{c \in C \mid |\bullet c| \geq 2\}$ é um conjunto de conectores do tipo *join*.
- $C_S = \{c \in C \mid |c\bullet| \geq 2\}$ é um conjunto de conectores do tipo *split*.
- $C_{IP} \subseteq C$ tal que $c \in C_{IP}$ se, e somente se, existe um caminho $p = \langle n_1, n_2, n_3 \rangle$, tal que $n_1 \in I, n_2 \in C, n_3 \in P$.
- $C_{PI} \subseteq C$ tal que $c \in C_{PI}$ se, e somente se, existe um caminho $p = \langle n_1, n_2, n_3 \rangle$, tal que $n_1 \in P, n_2 \in C, n_3 \in I$.

A definição 3 possibilita especificar requisitos adicionais que um Modelo de Processos de Negócios do EKD deveria satisfazer.

Definição 4. Um Modelo de Processo de Negócio do EKD satisfaz os seguintes requisitos:

- O conjunto I, P e C são conjuntos disjuntos, isto é, $I \cap P = \emptyset, I \cap C = \emptyset$, e $P \cap C = \emptyset$.
- Para cada $i \in I$: $|\bullet i| \leq 1$ e $|i\bullet| \leq 1$.
- Existe ao menos um *inf-set* $i \in I$, tal que $|\bullet i| = 0$ (*inf-set* inicial).
- Existe ao menos um *inf-set* $i \in I$, tal que $|i\bullet| = 0$ (*inf-set* final).
- Para cada $p \in P$: $|\bullet p| = 1$ e $|p\bullet| = 1$
- Para cada $c \in C$: $|\bullet c| \geq 1$ e $|c\bullet| \geq 1$.
- O gráfico induzido pelo MPN-EKD é fracamente conexo, isto é, se para cada dois nodos $n_1, n_2 \in N$, $(n_1, n_2) \in (A \cup A^{-1})^*$.
- C_J e C_S é divisão de C , isto é $C_J \cap C_S = \emptyset$ e $C_J \cup C_S = C$.
- C_{IP} e C_{PI} é divisão de C , isto é $C_{IP} \cap C_{PI} = \emptyset$ e $C_{IP} \cup C_{PI} = C$

O primeiro requisito da definição 4 declara que cada componente tem um identificador único (nome). Os nomes dos conectores são omitidos no Modelo de Processos de Negócio do EKD. Os outros requisitos correspondem a restrições na relação A . *Inf-sets* não podem ter múltiplos arcos de entrada e deve existir ao menos um *inf-set* inicial e um *inf-set* final. Cada processo tem ao menos um *inf-set* inicial e um *inf-set* final, um arco

de entrada e um arco de saída, para os dois nodos n_1 e n_2 (ignorando a direção dos arcos). Um conector c é um conector *join* ($|c \bullet| = 1$ e $|\bullet c| \geq 2$) ou *split* ($|\bullet c| = 1$ e $|c \bullet| \geq 2$). O último requisito declara que o conector c é um caminho de um *inf-set* para um processo ou um caminho de um processo para um *inf-set*. O Modelo de Processos de Negócio do EKD é sintaticamente correto se todos os requisitos declarados na definição 4 são satisfeitos.

6.5. Formalização do Mapeamento do Modelo de Processos de Negócio em Redes de Petri

Nesta seção, será apresentado o procedimento de mapeamento do Modelo de Processos de Negócio em redes de Petri. O procedimento de mapeamento é desenvolvido baseado em redes de Petri lugar/transição.

As definições (1) e (4) apresentadas no início do capítulo apenas relatam a sintaxe de um modelo de Processos de Negócio do EKD e não a semântica.

Os lugares representam *inf-sets* ou são construções necessárias para modelar o comportamento do conector do MPN-EKD. As transições representam processos ou estão representando o comportamento do conector. Cada conector $c \in C$ corresponde a lugares, transições e/ou arcos.

O conector pode corresponder a um número de arcos da rede de Petri ou uma pequena rede de lugares e transições. O conector OR corresponde ao comportamento de um lugar. O conector AND corresponde ao comportamento de uma transição. Na definição 5 o elemento Lugar de redes de Petri será representado por L para evitar confusão com o P de processo de MPN-EKD. A definição 5, apresentada a seguir, demonstra como é o mapeamento dos conectores do MPN-EKD desenvolvido neste trabalho.

Os arcos sempre têm peso igual a 1. No contexto do MPN-EKD não tem sentido ter peso maior que 1, porque lugares correspondem a condições. Em uma rede de Petri que corresponde a um MPN-EKD correto (*sound*) um lugar nunca conterá múltiplas marcas. A rede é segura. Os estados com múltiplas marcas em um lugar são resultados de erros de projeto. Para capturar esses erros é necessário considerar redes não-seguras.

Definição 5. Considere um MPN-EKD = (I, P, C, Q, A) . $N(EKD) = (L^{PN}, T^{PN}, F^{PN})$ é uma rede de Petri gerada pelo MPN-EKD:

$$L^{PN} = I \cup (\bigcup_{c \in C} L_c^{PN})$$

O conjunto de lugares (L^{PN}) é formado pela união de todos os *inf-sets* com lugares que foram incluídos para representar conectores ($\bigcup_{c \in C} L_c^{PN}$).

$$T^{PN} = P \cup (\bigcup_{c \in C} T_c^{PN})$$

O conjunto de transições (T^{PN}) é formado pela união de todos os Processos com transições que foram incluídas para representar conectores ($\bigcup_{c \in C} T_c^{PN}$).

$$F^{PN} = (A \cap ((I \times P) \cup (P \times I))) \cup (\bigcup_{c \in C} F_c^{PN})$$

O conjunto de arcos da rede (F^{PN}) é formado pelos arcos do modelo que vão de I a P e de P a I e a união dos arcos incluídos para representar conectores ($\bigcup_{c \in C} F_c^{PN}$).

A seguir serão apresentadas as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} de acordo com as regras de mapeamento relacionadas ao tipo de conectores do MPN-EKD. Em seguida a cada definição são apresentados exemplos que representam as regras utilizadas para mapear os conectores do MPN-EKD em redes de Petri.

Regra 1

$$c \in C_{IP} \cap C_J \cap C_{AND}$$

Quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_J (*join*) intersecção de C_{AND} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

$$I. L_c^{PN} = \emptyset$$

$$II. T_c^{PN} = \emptyset$$

$$III. F_c^{PN} = \{(x, y) | x \in \bullet c \text{ e } y \in c \bullet\}$$

A equação I determina que para representar esse conector não é necessário acrescentar lugares. A equação II determina que para representar esse conector não é necessário acrescentar transições. A equação III determina que os arcos que vão do conjunto de entrada do conector ao conjunto de saída do conector.

Nesse sentido, observa-se que o conector *AND-join* corresponde a dois ou mais arcos em redes de Petri se, e somente se, a saída é um processo. Na figura 28 é apresentado um exemplo de mapeamento do conector $c \in C_{IP} \cap C_J \cap C_{AND}$.

Figura 28- Exemplo de Mapeamento do C_{AND} entre dois ou mais *inf-sets* para um processo

Regra 2

$$c \in C_{PI} \cap C_J \cap C_{AND}$$

Quando o conector c pertence a C_{PI} (caminho de processo para *inf-set*) intersecção de C_J (*join*) intersecção de C_{AND} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

- I. $L_c^{PN} = \{ l_x^c \mid x \in \bullet c \}$
- II. $T_c^{PN} = \{ t^c \}$
- III. $F_c^{PN} = \{ (x, l_x^c) \mid x \in \bullet c \} \cup \{ (l_x^c, t^c) \mid x \in \bullet c \} \cup \{ (t^c, x) \mid x \in c \bullet \}$

Na equação I é determinado que para representar esse conector é necessário acrescentar um lugar para cada processo de entrada do conector. Na equação II é indicado que para representar esse conector é necessário acrescentar uma transição. Na equação III é determinado que para representar esse conector é necessário acrescentar arcos que ligam as transições aos lugares de entrada do conector, entre os lugares e a transição correspondente ao conector e entre a transição e o lugar de saída do conector.

Neste caso, o conector *AND-join* tem o comportamento de uma transição. É acrescentado um lugar para cada processo de entrada do conector. Na figura 29 é apresentado um exemplo de mapeamento do conector $c \in C_{PI} \cap C_J \cap C_{AND}$.

Figura 29- Exemplo de Mapeamento do C_{AND} entre dois ou mais processos para um *inf-set*

Regra 3

$$c \in C_{IP} \cap C_J \cap C_{OR}$$

Quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_J (*join*) intersecção de C_{OR} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

- I. $L_c^{PN} = \{l^c\}$
- II. $T_c^{PN} = \{t_x^c \mid x \in \bullet c\}$
- III. $F_c^{PN} = \{(x, t_x^c) \mid x \in \bullet c\} \cup \{(t_x^c, l^c) \mid x \in \bullet c\} \cup \{(l^c, x) \mid x \in c \bullet\}$

Na equação I é determinado que para representar este conector é necessário acrescentar um lugar. Na equação II é estabelecido que para representar este conector é necessário acrescentar uma transição para cada *inf-set* de entrada do conector. Na equação III é determinado que para representar este conector é necessário acrescentar um conjunto de arcos dos lugares às transições de entrada do conector, entre as transições e o lugar que corresponde o conector e entre o lugar e a transição de saída do conector.

O conector *OR-join* tem o comportamento de um lugar quando o conector é C_{IP} . Na figura 30 é apresentado um exemplo de mapeamento do conector $c \in C_{IP} \cap C_J \cap C_{OR}$.

Figura 30- Exemplo de Mapeamento C_{OR} entre dois ou mais *inf-sets* para um processo

Regra 4

$$c \in C_{PI} \cap C_J \cap C_{OR}$$

Quando o conector c pertence a C_{PI} (caminho de processo para *inf-set*) intersecção de C_J (*join*) intersecção de C_{OR} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

- I. $L_c^{PN} = \emptyset$
- II. $T_c^{PN} = \emptyset$
- III. $F_c^{PN} = \{(x, y) | x \in \bullet c \text{ e } y \in c \bullet\}$

Na equação I é apresentado que não é necessário acrescentar lugares para representar esse conector. Na equação II é determinado que não é necessário acrescentar transições para representar esse conector. Na equação III é apresentado que o conjunto de arcos estão entre o conjunto de entrada e o conjunto de saída.

O conector *OR-join* corresponde a dois ou mais arcos em redes de Petri se, e somente se, o conector é C_{PI} . Na figura 31 é apresentado um exemplo de mapeamento do conector $c \in C_{PI} \cap C_J \cap C_{OR}$.

Figura 31- Exemplo de Mapeamento C_{OR} entre dois ou mais processos para um *inf-set*

Regra 5

$$c \in C_{IP} \cap C_s \cap C_{AND}$$

Quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_s (*join*) intersecção de C_{AND} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

- I. $L_c^{PN} = \{ l_x^c \mid x \in c \bullet \}$
- II. $T_c^{PN} = \{ t^c \}$
- III. $F_c^{PN} = \{ (x, t^c) \mid x \in \bullet c \} \cup \{ (t^c, l_x^c) \mid x \in c \bullet \} \cup \{ (l_x^c, x) \mid x \in c \bullet \}$

Na equação I é apresentado que é necessário acrescentar um lugar para cada processo de saída do conector. Na equação II é determinado que para representar este conector é necessário acrescentar uma transição. Na equação III é apresentado que o conjunto de arcos necessários para representar este conector deve estar entre o lugar e a transição correspondente ao conector, entre a transição e os lugares de saída do conector e entre os lugares de saída do conector e a transição.

Dessa forma, o conector *AND-split* do tipo C_{IP} tem o comportamento de uma transição seguida de um número de lugares igual ao número de processos. Na figura 32 é apresentado um exemplo de mapeamento do conector $c \in C_{IP} \cap C_s \cap C_{AND}$.

Figura 32- Exemplo de Mapeamento do C_{AND} um *inf-set* para dois ou mais processos

Regra 6

$$c \in C_{PI} \cap C_S \cap C_{AND}$$

Quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_S (*join*) intersecção de C_{AND} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

- I. $L_c^{PN} = \emptyset$
- II. $T_c^{PN} = \emptyset$
- III. $F_c^{PN} = \{(x, y) | x \in \bullet c \text{ e } y \in c \bullet\}$

Na equação I é apresentado que não é necessário acrescentar lugares para representar esse conector. Na equação II é determinado que não é necessário acrescentar transições para representar esse conector. Na equação III é apresentado que o conjunto de arcos estão entre o conjunto de entrada e o conjunto de saída.

O conector *AND-split* corresponde a um número de arcos em redes de Petri se, e apenas se, a saída são dois ou mais *inf-sets*. Na figura 33 é apresentado um exemplo de mapeamento do conector $c \in C_{PI} \cap C_S \cap C_{AND}$.

Figura 33- Exemplo de Mapeamento do C_{AND} de um processo para dois ou mais *inf-sets*

Regra 7

$$c \in C_{IP} \cap C_S \cap C_{OR}$$

Quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_S (*join*) intersecção de C_{OR} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

- I. $L_c^{PN} = \emptyset$
- II. $T_c^{PN} = \emptyset$
- III. $F_c^{PN} = \{(x, y) | x \in \bullet c \text{ e } y \in c \bullet\}$

Na equação I é apresentado que não é necessário acrescentar lugares para representar esse conector. Na equação II é determinado que não é necessário acrescentar transições para representar esse conector. Na equação III é apresentado que o conjunto de arcos estão entre o conjunto de entrada e o conjunto de saída.

O conector *OR-split* corresponde a um número de arcos em redes de Petri se, e somente se, a saída são dois ou mais processos. Na figura 34 é apresentado um exemplo de mapeamento do conector $c \in C_{IP} \cap C_S \cap C_{OR}$.

Figura 34- Exemplo de Mapeamento do C_{OR} de um *inf-set* para dois ou mais processos

Regra 8

$$c \in C_{PI} \cap C_S \cap C_{OR}$$

Quando o conector c pertence a C_{PI} (caminho de processo para *inf-set*) intersecção de C_S (*join*) intersecção de C_{OR} , as definições de L_c^{PN} , T_c^{PN} e F_c^{PN} são as seguintes:

- I. $L_c^{PN} = \{l^c\}$
- II. $T_c^{PN} = \{t_x^c \mid x \in c \bullet\}$
- III. $F_c^{PN} = \{(x, l^c) \mid x \in \bullet c\} \cup \{(l^c, t_x^c) \mid x \in c \bullet\} \cup \{(t_x^c, x) \mid x \in c \bullet\}$

Na equação I é determinado que para representar este conector é necessário acrescentar um lugar. Na equação II é apresentado que é necessário acrescentar uma transição para cada *inf-set* de saída do conector. Na equação III é determinado que o conjunto de arcos deve estar entre a transição inicial e o lugar correspondente ao conector, entre o lugar e as transições de saída do conector e entre as transições de saída do conector e os lugares.

A única forma de existir um lugar com mais de um arco de saída é o mapeamento do conector *OR-split* de um processo para dois ou mais *inf-sets*. As regras de mapeamento garantem que o número de transições é igual ao número de lugares. A rede é escolha-livre. Na figura 35 é apresentado um exemplo de mapeamento do conector $c \in C_{PI} \cap C_S \cap C_{OR}$.

Figura 35- Exemplo de Mapeamento do C_{OR} de um processo para dois ou mais *inf-sets*

O MPN-EKD (I,P,C,Q,A) é um Modelo de Processos de Negócio do EKD e $PN=N(MPN-EKD)$ a rede de Petri gerada pelo MPN-EKD. PN é escolha-livre.

Definição 6. Um MPN-EKD é regular se e somente se:

- MPN-EKD tem dois *inf-sets* especiais: $i_{início}$ e i_{final} . *inf-set* $i_{início}$ é um nodo fonte: $\bullet i_{início} = \emptyset$. *inf-set* i_{final} é um nodo final: $i_{final} \bullet = \emptyset$.
- Todo nodo $n \in N$ está no caminho do $i_{início}$ para o i_{final} .

A identificação do $i_{início}$ e do i_{final} permite uma definição clara do estado inicial e do estado final. Um MPN-EKD com múltiplos *inf-sets* iniciais (por exemplo, *inf-sets* sem arcos iniciais) ou múltiplos *inf-sets* finais (por exemplo, *inf-sets* sem arcos finais) podem facilmente ser estendidos com uma parte de inicialização ou finalização que o primeiro requisito é satisfeito. O segundo requisito precisa que todo *inf-set* ou processo esteja entre $i_{início}$ e i_{final} . Se o segundo requisito não é satisfeito, então o MPN-EKD é:

- i. Composto de partes completamente disjuntas.
- ii. Tem partes que nunca são ativadas.
- iii. Partes do MPN-EKD formam armadilhas.

Como o MPN-EKD descreve o ciclo de vida de um caso (por exemplo, uma instância de processo), os dois requisitos são razoáveis. O ciclo de vida deveria ter um evento inicial e um evento final, e todos os passos deveriam estar em um caminho entre esses dois eventos. No presente trabalho será considerado MPN-EKD regulares.

Um MPN-EKD descreve um procedimento com um estado inicial e um estado final. O procedimento deveria ser projetado de tal forma que sempre termine corretamente.

Além disso, deveria ser possível executar qualquer processo seguindo a rota apropriada do MPN-EKD.

Definição 7. Um MPN-EKD regular é *sound* se e somente se:

1. Para cada marcação M alcançável do estado inicial (por exemplo, o estado, onde o *inf-set* $i_{\text{início}}$ é o único *inf-set* que existe), existe uma seqüência de disparos levando da marcação M para a marcação final (por exemplo, onde o *inf-set* i_{final} é o único *inf-set* que existe).
2. A única marca existente no final do processo está no estado final i_{final} .
3. Não existir processos *dead*, por exemplo, para cada processo p existe uma seqüência disparável, a qual executa p .

O critério de corretitude definido em redes de Petri é o requisito mínimo que qualquer MPN-EKD deveria satisfazer. Um MPN-EKD *Sound* é livre de potenciais *deadlocks* e *livelocks*. Se assume *fairness*, então os primeiros dois requisitos implica que eventualmente a marcação final será alcançada (nota-se que esse é um resultado da combinação da propriedade de *soundness* e de escolha-livre). A propriedade escolha-livre implica que para cada transição t_1 e t_2 , $\bullet t_1 \cap \bullet t_2 \neq \emptyset$ que implica que $\bullet t = \bullet t_2$.

Os MPN-EKD complexos encontrados na prática não são fáceis de verificar a propriedade *soundness*. Felizmente, técnicas e ferramentas baseadas em redes de Petri podem ser usadas para analisar essa propriedade. A inspeção da árvore de cobertura da rede de Petri, que corresponde ao MPN-EKD é suficiente para verificar *soundness*. Para MPN-EKD complexos, o gráfico de *coverability* pode tornar-se muito grande. Esse fenômeno é conhecido como o “problema de explosão de estado”. Um MPN-EKD com 80 processos pode facilmente ter mais de 200.000 marcações. Embora, os computadores atualmente tenham dificuldade para analisar gráficos de *coverability* desse tamanho, existem muitas técnicas avançadas que exploram a estrutura de redes de Petri, nesse caso gerada por um MPN-EKD. Essas técnicas permitem eficientes procedimentos de decisão. Antes de apresentar tal procedimento, é necessário primeiramente listar algumas propriedades presentes em qualquer rede de Petri gerada por um MPN-EKD *sound*.

Considerando que o $\text{MPN-EKD} = (I, P, C, Q, A)$ é *sound* e $\text{PN} = N(\text{MPN-EKD})$ a rede de Petri gerada pelo MPN-EKD. Considere $\overline{\text{PN}}$ ser PN com uma transição t adicional conectando i_{final} para $i_{\text{início}}$ e deixar M ser a marcação inicial com uma marca em $i_{\text{início}}$.

- $\overline{PN}$ é fortemente conexa;
- $\overline{PN}$ é passível de cobertura.
- $(\overline{PN}, M)$ é viva.
- $(\overline{PN}, M)$ é limitada.

Uma rede de Petri é fortemente conexa se, e somente se, para cada par de nodos (lugares e transições) x e y , existe um caminho de x até y (Aalst, 1997).

PN é fortemente conexa por que todos os nodos estão no caminho do $i_{início}$ para o i_{final} e i_{final} é conectado em $i_{início}$ via t adicional. PN é WF-net de acordo com Aalst (1997). Portanto, *soundness* coincide com vivacidade e limitação.

$(\overline{PN}, M)$ é escolha-livre, viva e limitada e, de acordo com Aalst & Hee (2002), implica que PN é passível de cobertura e (PN, M) é segura. Construindo os resultados apresentados em Aalst (1997), a propriedade de *soundness* pode ser verificada em tempo polinomial.

Um MPN-EKD corresponde a uma rede WF-net escolha-livre. WF-net foi apresentada em detalhes no capítulo 5. Um rede WF é *sound* se e somente se a rede extendida é *viva* e limitada. Vivacidade e limitação podem ser decididos em tempo polinomial. Por essa razão *soundness* pode ser verificado em tempo polinomial.

Nesse sentido, é possível estender ferramentas com procedimentos eficientes de decisão para verificar a propriedade *soundness* de um MPN-EKD. Para guiar o usuário em procurar e corrigir defeitos em um projeto de um MPN-EKD, é possível também suprir diagnósticos adicionais baseados na estrutura do MPN-EKD/redes de Petri.

6.6. Método de avaliação do Modelo de Processos de Negócio

Neste trabalho foram desenvolvidos a formalização do Modelo de Processos de Negócio do EKD e o mapeamento desse mesmo modelo em redes de Petri. Neste item, será apresentada a primeira versão, desenvolvida neste trabalho, do método de avaliação do Modelo de Processos de Negócio do EKD. Na figura 36 é apresentado um esquema do método.

A primeira versão do método de avaliação do Modelo de Processos de Negócio consiste em:

1. Desenvolver o modelo organizacional EKD utilizando as diretrizes apresentadas em Bubenko *et al.* (1998).
2. Desenvolver o Modelo de Processos de Negócio de acordo com a formalização do MPN-EKD desenvolvida neste trabalho. Conferir se o modelo atende aos seguintes requisitos:
 - 2.1. Todos os processos devem ter condição de entrada e saída. Quando um processo não tem nenhuma condição de entrada, não fica claro quando poderá ser realizado. Quando um processo não tem condições de saída, não contribui para o sucesso do caso e pode ser omitido.
 - 2.2. Deve haver ao menos um *inf-set* final e um *inf-set* inicial.
 - 2.3. A entrada do processo deve ser igual a 1.
 - 2.4. A saída do processo deve ser igual a 1.
 - 2.5. A saída do *inf-set* deve ser igual ou menor que 1. Menor que 1 caso seja um *inf-set* final.
 - 2.6. A entrada do *inf-set* deve ser igual ou menor que 1. Menor que 1 caso seja um *inf-set* inicial.
 - 2.7. A entrada do conector deve ser maior ou igual a 1.
 - 2.8. A saída do conector deve ser maior ou igual a 1.
 - 2.9. Todo conector deve ser do tipo OR ou AND.
 - 2.10. Todo conector deve ser do tipo *Split* ou *Join*.
 - 2.11. Todo conector deve ser do tipo PI ou IP.
 - 2.12. Um conector do tipo *split* deve ter a entrada igual a 1.
 - 2.13. Um conector do tipo *split* deve ter a saída igual ou maior que 2.
 - 2.14. Um conector do tipo *join* deve ter a entrada maior ou igual a 2.
 - 2.15. Um conector do tipo *join* deve ter a saída igual a 1.
 - 2.16. Não é permitido conectar processo a processo.
 - 2.17. Não é permitido conectar *inf-set* a *inf-set*.

- 2.18. Não é permitido utilizar o conector ligando processo(s) a processo(s) e *inf-set(s)* a *inf-set(s)*.
- 2.19. Não é permitido a conexão de conector (es) com conector(es).
- 2.20. Todos os *inf-sets* que não foram gerados pelo processo devem ser habilitados.
3. Mapear o modelo em redes de Petri. Os *inf-sets* são representados por lugares e os processos são representados por transições. Para mapear os conectores é necessário seguir as regras apresentadas a seguir:
- 3.1. O conjunto de lugares é formado pela união de todos os *inf-sets* com lugares que foram incluídos para representar os conectores.
- 3.2. O conjunto de transições é formado pela união de todos os Processos com transições que foram incluídas para representar conectores.
- 3.3. O conjunto de arcos da rede é formado pelos arcos de modelo que vão de I (*inf-set*) a P (processo) e de P a I e a união dos arcos incluídos para representar conectores.
- 3.4. A regra 1 estabelece que quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_J (*join*) intersecção de C_{AND} , o conector ($c \in C_{IP \cap C_J \cap C_{AND}}$) corresponde a dois ou mais arcos em redes de Petri.
- 3.5. A regra 2 estabelece que quando o conector c pertence a C_{PI} (caminho de processo para *inf-set*) intersecção de C_J (*join*) intersecção de C_{AND} , o conector ($c \in C_{PI \cap C_J \cap C_{AND}}$) tem o comportamento de uma transição. É acrescentado um lugar para cada processo de entrada do conector.
- 3.6. A regra 3 estabelece que quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_J (*join*) intersecção de C_{OR} , o conector ($c \in C_{IP \cap C_J \cap C_{OR}}$) tem o comportamento de um lugar.
- 3.7. A regra 4 estabelece que quando o conector c pertence a C_{PI} (caminho de processo para *inf-set*) intersecção de C_J (*join*) intersecção de C_{OR} , o conector ($c \in C_{PI \cap C_J \cap C_{OR}}$) corresponde a dois ou mais arcos em redes de Petri.
- 3.8. A regra 5 estabelece que quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_S (*split*) intersecção de C_{AND} , o conector (c

$\in C_{IP} \cap C_S \cap C_{AND}$) tem o comportamento de uma transição seguida de um número de lugares igual ao número de processos.

- 3.9. A regra 6 estabelece que quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_S (*split*) intersecção de C_{AND} , o conector ($c \in C_{PI} \cap C_S \cap C_{AND}$) corresponde a um número de arcos em redes de Petri.
- 3.10. A regra 7 estabelece que quando o conector c pertence a C_{IP} (caminho de *inf-set* para processo) intersecção de C_S (*split*) intersecção de C_{OR} , o conector ($c \in C_{IP} \cap C_S \cap C_{OR}$) corresponde a um número de arcos em redes de Petri.
- 3.11. A regra 8 estabelece que quando o conector c pertence a C_{PI} (caminho de processo para *inf-set*) intersecção de C_S (*split*) intersecção de C_{OR} , o conector ($c \in C_{PI} \cap C_S \cap C_{OR}$) corresponde a um lugar seguido de um número de transições igual ao número de processos de saída do conector.
4. Construir a árvore de alcançabilidade. Por meio da árvore de alcançabilidade é possível verificar vários erros que podem ocorrer na definição de processo, mesmo sem conhecimento específico do processo de negócio. Na ausência de ferramenta de edição de redes de Petri é possível verificar a propriedade *soundness* por meio da inspeção da árvore de alcançabilidade que corresponde ao MPN-EKD
5. Depois de fazer o mapeamento em redes de Petri, avaliar o modelo utilizando uma ferramenta de edição de redes de Petri. Neste trabalho, está sendo utilizado *Petri Net Tools*. Os seguintes itens devem ser considerados:
 - 5.1. Verificação de possível travamento (*deadlock*), ou seja, quando não é possível executar nenhuma tarefa.
 - 5.2. Eliminação de casos que ficam em loop infinito (*livelock*).
 - 5.3. Verificação de possíveis tarefas que não podem ser executadas (*deadtask*).
 - 5.4. Eliminação de conflitos.
 - 5.5. Verificação de possíveis caminhos.
 - 5.6. Verificar a existência de marcas nos outros lugares depois que a condição fim foi completada. Uma vez que a marca aparece no lugar fim, todas as outras marcas devem ter desaparecido.
6. Apresentar relatório de problemas encontrados.

Figura 36 - Esquema do método de avaliação do modelo de processos de negócio

6.7. Considerações Finais

Ambigüidades e confusão não podem ser prevenidas em um modelo de processos de negócio informal. Para resolver esse problema foi desenvolvida a formalização do Modelo de Processos de Negócio do EKD.

Foram desenvolvidos um procedimento de mapeamento formal do Modelo de Processos de Negócio em redes de Petri e a primeira versão do método de avaliação do Modelo de Processos de Negócio. No próximo capítulo são apresentados testes da primeira versão do método de avaliação do Modelo de Processos de Negócio.

CAPÍTULO 7

Testes do Método de Avaliação do Modelo de Processos de Negócio

7.1. Considerações Iniciais

Os testes da primeira versão do método de avaliação do Modelo de Processos de Negócio do EKD são apresentados neste capítulo. Os modelos foram simulados na ferramenta Petri Net Tools. As árvores de alcançabilidade foram desenvolvidas e analisadas.

7.2. Considerações sobre os testes

Os modelos que foram utilizados neste capítulo foram alterados para que fosse possível a realização dos testes do método de avaliação do modelo de processos de negócio.. Como os conectores foram criados neste trabalho na formalização do MPN-EKD, foram adicionados conectores nos modelos retirados da literatura.

Não serão apresentados todos os possíveis relacionamentos entre Modelo de Processos de Negócio e outros submodelos do Modelo Organizacional, apenas as ligações mais importantes, para reduzir a complexidade do modelo.

Esses modelos são mapeados em Redes de Petri, seguindo o procedimento de mapeamento desenvolvido neste trabalho.

Atualmente existem diversas ferramentas para edição, simulação e análise de modelos em rede de Petri. Uma destas, chamada *Petri Net Tools*, foi desenvolvida e implementada no Laboratório de Simulação e Controle de Sistemas Discretos, da USP

de São Carlos (Soares, 2001). Todos os modelos desta seção foram simulados na ferramenta *Petri Net Tools*.

O editor possui seis módulos. Quatro módulos estão em funcionamento: rede de Petri L/T (lugar-transição), MFG (*Mark Flow Graph*), SFC (*Sequential Flow Chart*) e PN Estocásticas. Dois módulos estão em fase de implantação: PLC, Virtual Petri Net (Redes de Petri Virtual). O editor permite os seguintes tipos de análises:

1. árvore de alcançabilidade;
2. matriz de incidência;
3. limitação;
4. vivacidade;
5. verificação do estado final;
6. transições e lugares invariantes.

Nos modelos mapeados em redes de Petri, os lugares e transições, que estão em vermelho representam o mapeamento dos conectores. Esses lugares e transições são nomeados neste trabalho como *IS adic* e *Pr adic*. Esses lugares e transições não estão apresentados nos quadros que apresentam os elementos de redes de Petri correspondentes aos elementos do MPN-EKD. Também são apresentados em vermelhos os processos e *inf-sets* adicionais utilizados como estratégia para tratar as construções que não são permitidas de acordo com as definições de AND-Split, AND-Join, OR-Split e OR-Join apresentadas no capítulo 6. Os lugares e transições em azul representam o início e o fim do processo.

As árvores de alcançabilidade são desenvolvidas em cada teste realizado. Por meio da árvore de alcançabilidade é possível verificar vários erros que podem ocorrer na definição de processo, mesmo sem conhecimento específico do processo de negócio. Para verificar *soundness*, a inspeção da árvore de alcançabilidade da rede de Petri, que corresponde ao MPN-EKD é suficiente.

Os testes estão divididos em casos. No primeiro caso, o método de avaliação foi testado em um modelo desenvolvido neste trabalho: modelo de processo de seleção de alunos para o Programa de Pós-Graduação na Engenharia Mecânica da Escola de Engenharia de São Carlos – USP. O segundo caso, o método foi testado no modelo de processo de empréstimo de livros de Bubenko *et al* (1998). O terceiro caso, o método foi testado em

cinco modelos de processos de negócio do projeto ESPRIT ELEKTRA (Electrical Enterprise Knowledge for Transforming Applications). O quarto caso, o método foi testado em seis modelos que fazem parte do estudo de caso de Pinotti (2004).

7.3. Caso 1 – Modelo de Seleção de Alunos

O primeiro modelo apresentado é o processo de seleção de alunos para o Programa de Pós-Graduação na Engenharia Mecânica da Escola de Engenharia de São Carlos – USP. No processo de seleção, o aluno solicita informações na secretaria, preenche o formulário e entrega os documentos. A secretária verifica o formulário e documentos, se a documentação estiver correta, pontuação do aluno é verificada. O aluno participará do processo seletivo se a pontuação atingir o mínimo exigido. Caso ele tenha a pontuação mínima, a secretária disponibiliza documentos e formulário para orientadores. O aluno será selecionado para entrevista se um ou mais orientadores manifestarem interesse pelo aluno. O aluno faz a entrevista. Após a entrevista, a comissão avalia o aluno e na sequência divulga o resultado. Na figura 37 é apresentado o modelo de processo de seleção de acordo com o método EKD. Na figura 38 é apresentado o modelo mapeado em redes de Petri de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 5. Os conectores do Modelo de Seleção de Alunos e suas respectivas regras de mapeamento são apresentados no quadro 6.

Resultado do teste

A simulação do modelo apresentada na figura 39 demonstra quatro caminhos diferentes para completar o processo seleção de aluno.

A árvore de alcançabilidade da figura 40 demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja o modelo é *sound*.

Figura 37 - Modelo de Processo de Seleção de alunos

Figura 38- Modelo de Seleção de Alunos Mapeado em Redes de Petri

PN	MPN-EKD	
Lugar	<i>Inf-set</i>	Descrição
Is1	1	Aluno solicita matrícula
Is2	2	Formulário e documentos dos interessados
Is3	3	Documentação correta
Is4	4	Documentação incorreta
Is5	5	Inscrição efetuada
Is6	6	Pontuação insuficiente
Is7	7	Pontuação suficiente
Is8	8	Orientador não tem interesse no aluno
Is9	9	Orientador tem interesse no aluno
Is10	10	Lista de alunos convocados para entrevista
Is11	11	Anotações da entrevista
Is12	12	Resultado da avaliação
Is13	13	Resultado divulgado
Transição	Processo	Descrição
Pr1	1	Preencher formulário e entregar documentos
Pr2	2	Verificar formulários e documentos
Pr3	3	Efetuar inscrição para seleção
Pr4	4	Verificar status do candidato
Pr5	5	Disponibilizar formulários para orientadores
Pr6	6	Chamar para entrevista
Pr7	7	Fazer entrevista com aluno
Pr8	8	Avaliar aluno
Pr9	9	Divulgar resultado

Quadro 5 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de seleção de alunos

Conector	Regra correspondente
$C_{PI} C_S C_{OR}$	Regra 8
$C_{IP} C_J C_{OR}$	Regra 3

Quadro 6 - Conectores do Modelo de Seleção de Alunos e a regra de mapeamento correspondente

Figura 40 - Árvore de Alcançabilidade do Modelo de Seleção de Alunos

7.4. Caso 2 – Modelo de Empréstimo

Na figura 41 é demonstrado um modelo para empréstimo de livros original de Bubenko *et al.* (1998). O processo é iniciado com o pedido de um cliente. O pedido pode ser rejeitado ou ser aceito. Se o pedido for rejeitado o processo é finalizado. Se o pedido é aceito, o próximo passo é consultar se o livro existe na biblioteca e se existe verificar se está emprestado. O resultado dessa consulta pode ser: livro não está disponível, livro não existe e livro está emprestado para outro cliente. Se o livro está disponível, o livro é entregue ao cliente e é registrada a transação de empréstimo. Se o livro está emprestado, o cliente tem a opção de ficar ou não na fila de espera do livro. Na figura 42 é apresentado o modelo com alterações. Na figura 43 é apresentado o modelo de empréstimo de livros mapeado em redes de Petri de acordo com o procedimento de mapeamento desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 7. Os conectores do Modelo de Empréstimos de Biblioteca e suas respectivas regras de mapeamento são apresentados no quadro 8.

Resultado do teste

Na figura 44 é apresentado o modelo na ferramenta *Petri Net Tools*. Não foi possível realizar a simulação porque o modelo apresenta problemas. Na figura 45 é apresentada a árvore de alcançabilidade com a marcação inicial no lugar início. A árvore apresentada na figura 45 demonstra que há um *deadlock* no Pr2, neste caso, o modelo não é *sound*. Além disso, é necessário ter marcação inicial no IS15 para que Pr9 dispare e a realimentação, caso o Pr10 fosse realizado, deixaria uma marca não desejada no IS15.

Figura 41 - Modelo de Processos de empréstimos de biblioteca original

Fonte: Adaptado de Bubenko *et al.* (1998) p.50

Figura 42 - Modelo de Processos de empréstimos de biblioteca

Figura 43 - Modelo de Processo de biblioteca mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>inf-set</i>	Descrição
Is1	1	Cliente pede um livro
Is2	2	Pedido rejeitado
Is3	3	Biblioteca aceita o pedido
Is4	4	Catálogo de livros
Is5	5	Empréstimo existente
Is6	6	Livro está disponível
Is7	7	Livro não existe
Is8	8	Livro está emprestado para outro cliente
Is9	9	Livro entregue ao cliente
Is10	10	Livro
Is11	11	Livro não está disponível
Is12	12	Cliente recusa esperar na fila
Is13	13	Cliente aceita esperar na fila
Is14	14	Aceitação na fila
Is15	15	Fila
Transição	Processo	Descrição
Pr1	1	Receber pedido
Pr2	2	Pesquisar todas as cópias da biblioteca
Pr5	5	Biblioteca responde ao cliente
Pr8	8	Entregar livro ao cliente
Pr9	9	Negociar com cliente
Pr10	10	Atualizar fila
Pr11	11	Registrar transação de empréstimo

Quadro 7 - Elementos de Redes de Petri correspondentes aos elementos do MPN- EKD de Empréstimos de livros

Conector	Regra correspondente
$C_{PI} C_S C_{OR}$	Regra 8
$C_{IP} C_J C_{AND}$	Regra 1
$C_{PI} C_J C_{AND}$	Regra 7
$C_{IP} C_J C_{OR}$	Regra 3
$C_{PI} C_S C_{AND}$	Regra 6

Quadro 8 - Conectores utilizados no Modelo de Empréstimo de Biblioteca e a regra de mapeamento correspondente

Figura 44 - Simulação do Modelo de Empréstimo de Biblioteca na ferramenta *Petri Net Tools*

Figura 45 - Árvore de Alcançabilidade do Modelo de Empréstimo de Biblioteca

7.4.1. Proposta de Modificação do Modelo de Empréstimo de Livros

Nessa seção será apresentada uma proposta de modificação do modelo de Empréstimo de biblioteca. O modelo apresentado na figura 40 é original de Bubenko *et al* (1998), sendo assim não foi desenvolvido de acordo com o item 2 do método de avaliação do modelo de processos de negócio do EKD desenvolvido neste trabalho. No item 2 do método de avaliação, é afirmado que o *inf-set* deve ter a entrada igual ou menor que 1. Menor que 1 no caso de ser o *inf-set* inicial. O *inf-set* 4 não tem entrada e não é o *inf-set* inicial do processo. Nesse caso, o *inf-set* 4 não está habilitado ao iniciar o processo e haverá um *deadlock*.

O *inf-set* 5 e o *inf-set* 15 não foram gerados pelo processo e por isso não estão habilitados a disparar o processo. Essa situação também provoca *deadlock*. No item 2 do método de avaliação é apresentado que no modelo de processos de negócio do EKD formal, os *inf-sets* que não foram gerados pelo processo deverão ser habilitados para que o modelo seja *sound*.

Como afirmado anteriormente, para um modelo ser considerado *sound* deve atender os seguintes requisitos:

4. Não deve existir tarefa que não contribui para o processamento dos casos.
5. Para qualquer caso, o procedimento terminará eventualmente.
6. No momento em que o procedimento termina para casos específicos, todas as referências a esse caso devem ser removidas.

No modelo da figura 40, o processo 11 deixa o *inf-set* 5 habilitado mesmo quando o processo é finalizado. Nesse caso, o modelo também não atende ao último requisito para ser considerado *sound*. Essa situação é claramente percebida na simulação e na árvore de alcançabilidade.

Para solucionar esses problemas foram realizadas as seguintes modificações:

- Os *inf-sets* 4, 5 e 15 foram habilitados.
- Foi adicionado o *inf-set* 16 (transação registrada) que é a saída do processo 11 (registrar transação de empréstimo).
- O *inf-set* 16 foi conectado ao processo 12 (finalizar).
- Foi eliminado o *inf-set* 10 por ser redundante e não contribuir para o caso.

Na figura 46 é apresentado o modelo para empréstimo de livros de Bubenko *et al.* (1998) com a proposta de modificação para corrigir os problemas apresentados. Na figura 47 é apresentado o modelo proposto de empréstimo de livros mapeado em redes de Petri de acordo com o procedimento de mapeamento desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 9. Os conectores do Modelo de Empréstimos de Biblioteca e suas respectivas regras de mapeamento são apresentados no quadro 10.

Resultado do teste

Na figura 48 é apresentada a simulação do modelo na ferramenta *Petri Net Tools*. A simulação demonstra que o modelo é *sound*. A árvore apresentada na figura 49 também demonstra que o modelo é seguro, não apresenta tarefas que não contribuem para o processamento do caso, não tem *deadlock*, *livelock*, tem lugar início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 46 - Modelo de Processos de empréstimos de biblioteca com modificações na estrutura do modelo

Figura 47 - Modelo de Processo de biblioteca com modificações na estrutura mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>inf-set</i>	Descrição
Is1	1	Cliente pede um livro
Is2	2	Pedido rejeitado
Is3	3	Biblioteca aceita o pedido
Is4	4	Catálogo de livros
Is5	5	Empréstimo existente
Is6	6	Livro está disponível
Is7	7	Livro não existe
Is8	8	Livro está emprestado para outro cliente
Is9	9	Livro entregue ao cliente
Is11	11	Livro não está disponível
Is12	12	Cliente recusa esperar na fila
Is13	13	Cliente aceita esperar na fila
Is14	14	Aceitação na fila
Is15	15	Fila
Is16	16	Empréstimo registrado
Transição	Processo	Descrição
Pr1	1	Receber pedido
Pr2	2	Pesquisar todas as cópias da biblioteca
Pr2.1	2.1	Habilitar catálogo e empréstimos
Pr5	5	Biblioteca responde ao cliente
Pr8	8	Entregar livro ao cliente
Pr9	9	Negociar com cliente
Pr9.1	9.1	Habilitar fila
Pr10	10	Atualizar fila
Pr11	11	Registrar transação de empréstimo

Quadro 9 - Elementos de Redes de Petri correspondentes aos elementos do MPN- EKD de Empréstimos de livros com a modificações na estrutura

Conector	Regra correspondente
$C_{PI} C_S C_{OR}$	Regra 8
$C_{IP} C_J C_{AND}$	Regra 1
$C_{IP} C_J C_{OR}$	Regra 3
$C_{PI} C_S C_{AND}$	Regra 6

Quadro 10 - Conectores utilizados no Modelo de Empréstimo de Biblioteca com modificações na estrutura e a regra de mapeamento correspondente

Figura 49 - Árvore de Alcançabilidade do Modelo de Empréstimo de Biblioteca

7.5. Caso 3 – ELEKTRA

Os próximos modelos foram retirados do projeto ESPRIT ELEKTRA (*Electrical Enterprise Knowledge for Transforming Applications*). O projeto ELEKTRA concentra-se principalmente na aplicação do método EKD para problemas de gerenciamento de mudanças dentro de organizações da Grécia e Suécia, gerando um conjunto de práticas genéricas de forma a aplicá-las em outras companhias. Dentro do projeto ELEKTRA, o caso Vattenfall foi escolhido.

O projeto ELEKTRA foi baseado em uma cuidadosa análise não apenas das práticas correntes e processo, mas também de problemas, das necessidades, das oportunidades e objetivos futuros percebidos.

7.5.1. Modelo de Planejamento estratégico de Recursos Humanos

O primeiro modelo de projeto ELEKTRA que será apresentado neste trabalho é o processo de planejamento estratégico de recursos humanos. O processo de planejamento é realizado em um nível estratégico. Envolve a formulação de políticas e objetivos para o planejamento de recursos humanos para Vattenfall tratando as métricas sobre a realização dos objetivos formulados.

O modelo descreve como o planejamento de recursos humanos deveria ser integrado com o planejamento estratégico de negócio. Inicialmente, partindo dos objetivos do planejamento do negócio, é formulado o grupo de objetivos e indicadores dentro do domínio de competência, em seguida são comunicados os objetivos e indicadores para área de negócio nos documentos: pré-condições para planejamento de negócio/diretrizes de orçamento. Paralelamente ou não, são formuladas as políticas, as diretrizes e as instruções a partir do termo dos objetivos particulares e políticos. Em seguida, são feitas comunicações sobre essas políticas. Os processos seguintes são: executar atividades dentro do domínio do suplente na competência; retornar trimestralmente os objetivos e indicadores com a ajuda do Sistema de Revisão do Grupo; apresentar um sumário dos objetivos alcançados e indicadores; comparar objetivos realizados e tendências com objetivos propostos de acordo com as políticas particulares e planejamento de negócio e revisar objetivos. Na figura 50 é apresentado o modelo descrito original. Na figura 51 é apresentado o modelo alterado. As entradas do processo 1 foram modificadas por não atender ao requisito do método que afirma que o processo só poderá ter uma entrada.

Embora o método afirma que o deve ser colocado um *inf-set* inicial e final, não foi possível colocar *inf set* final por não estar claro o fim do procedimento. Foram adicionados os conectores.

Na figura 52 é apresentado o mesmo modelo mapeado em redes de Petri de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 11. Os conectores do Modelo de Processo de planejamento estratégico de recursos humanos e suas respectivas regras de mapeamento são apresentados no quadro 12.

Resultado do teste

Na figura 53 é apresentado o modelo simulado na ferramenta Petri Net Tools. A árvore de alcançabilidade apresentada na figura 54 demonstra que a transição Pr6 será disparada apenas se houver uma marca no lugar IS8. Como não existe essa marca antes do disparo da transição Pr7 haverá um *deadlock*. Além disso, não existe uma condição fim clara. O modelo não é *sound*.

Figura 50 - Modelo de Processo de planejamento estratégico de recursos humanos original

Figura 51 - Modelo de Processo de planejamento estratégico de recursos humanos modificado

Figura 52 - Modelo de Planejamento estratégico mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>inf-set</i>	Descrição
Is1	1	Objetivos do planejamento de negócio
Is2	2	Termo dos objetivos particulares e políticos
Is3	3	Objetivos do grupo e indicadores dentro do domínio de competência
Is4	4	Políticas, diretrizes e instruções dentro do domínio de suprimentos de competências
Is5	5	Proposta para indicadores e objetivos revisados
Is6	6	Pré-condições para planos de negócio/diretrizes de orçamento
Is7	7	Políticas, diretrizes e instruções dentro do domínio de suplentes na competência
Is8	8	Objetivos revisados e indicadores dentro do domínio de suplentes na competência
Is9	9	Resultados das atividades para suplentes na competência
Is10	10	Informações sobre a área de negócio
Is11	11	Informações sobre a área do negócio para atingir objetivos
Is12	12	Objetivos e indicadores agregados
Transição	Processo	Descrição
Pr1	1	Formular Grupo de Objetivos e indicadores dentro do domínio de competência
Pr2	2	Formular políticas
Pr3	3	Comunicar objetivos e indicadores para área de negócio nos documentos: pré-condições para planejamento de negócio/diretrizes para orçamento
Pr4	4	Comunicar políticas via folder ou intranet
Pr5	5	Revisar objetivos e indicadores
Pr6	6	Executar atividades dentro do domínio do suplente na competência
Pr7	7	Revisão dos objetivos e indicadores
Pr8	8	Retornar trimestralmente os objetivos e indicadores com a ajuda do Sistema de Revisão do Grupo
Pr9	9	Resumo dos objetivos alcançados e indicadores
Pr10	10	Comparar objetivos realizados e tendências com objetivos propostos de acordo com as políticas particulares e planejamento de negócio

Quadro 11 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Planejamento estratégico

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{OR}$	Regra 3
$C_{IP} C_J C_{AND}$	Regra 1
$C_{IP} C_S C_{OR}$	Regra 7

Quadro 12 - Conectores utilizados no Modelo de Planejamento estratégico de recursos humanos e a regra de mapeamento correspondente

Figura 53 - Simulação do Modelo de Planejamento estratégico na ferramenta Petri Net Tools

Figura 54 – Árvore de alcançabilidade do Modelo de Planejamento estratégico mapeado em redes de Petri

7.5.2. Proposta de Modificação do Modelo de Planejamento estratégico de Recursos Humanos

Nessa seção será apresentada uma proposta de modificação do modelo de planejamento estratégico de recursos humanos. O modelo apresentado na figura 50 é original do projeto ELEKTRA, sendo assim não foi desenvolvido de acordo com o item 2 do método de avaliação do modelo de processos de negócio do EKD desenvolvido neste trabalho.

O modelo apresentado na figura 51, não é *sound* por que o processo 6 será disparado apenas se o *inf-set* 8 estiver habilitado. Esse é um caso de realimentação. A realimentação é sempre dependente das condições geradas pelo próprio processo. Como o *inf-set* 8 não foi ainda gerado existirá um *deadlock*.

No item 2 do método de avaliação é apresentado que no modelo de processos de negócio do EKD formal os *inf-sets* que não foram gerados pelo processo deverão ser habilitados para que o modelo seja *sound*.

No item 2 do método de avaliação é afirmado que deve haver condição de entrada e saída. No modelo não existe uma condição fim clara. Para resolver esses problemas foram realizadas as seguintes mudanças:

- Foi adicionado o Processo 6A para evitar o *deadlock*.
- Foi retirado o conector que liga o *inf-set* 5 ao *inf-set* 2 para que o processo tenha um fim claro.
- Foi adicionado *inf-set* fim e processo fim.

Na figura 55 é apresentado o modelo de planejamento estratégico de recursos humanos com a proposta de modificação para resolver os problemas apresentados. Na figura 56 é apresentado o mesmo modelo mapeado em redes de Petri de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 13. Os conectores do Modelo de Processo de planejamento estratégico de recursos humanos e suas respectivas regras de mapeamento são apresentados no quadro 14.

Resultado do teste

Na figura 57 é apresentado o modelo simulado na ferramenta *Petri Net Tools*. A simulação demonstra que o modelo é seguro, não apresenta tarefas que não contribuem

para o processamento do caso, não tem *deadlock*, *livelock*, tem lugar início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*. A árvore de alcançabilidade apresentada na figura 58 também demonstra que o modelo proposto é *sound*.

Figura 55 - Modelo de Processo de planejamento estratégico de recursos humanos com proposta de modificação

Figura 56 - Modelo de Planejamento estratégico com a proposta de modificação mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>inf-set</i>	Descrição
Is1	1	Objetivos do planejamento de negócio
Is2	2	Termo dos objetivos particulares e políticos
Is3	3	Objetivos do grupo e indicadores dentro do domínio de competência
Is4	4	Políticas, diretrizes e instruções dentro do domínio de suprimentos de competências
Is5	5	Proposta para indicadores e objetivos revisados
Is6	6	Pré-condições para planos de negócio/diretrizes de orçamento
Is7	7	Políticas, diretrizes e instruções dentro do domínio de suplentes na competência
Is8	8	Objetivos revisados e indicadores dentro do domínio de suplentes na competência
Is9	9	Resultados das atividades para suplentes na competência
Is10	10	Informações sobre a área de negócio
Is11	11	Informações sobre a área do negócio para atingir objetivos
Is12	12	Objetivos e indicadores agregados
Transição	Processo	Descrição
Pr1	1	Formular Grupo de Objetivos e indicadores dentro do domínio de competência
Pr2	2	Formular políticas
Pr3	3	Comunicar objetivos e indicadores para área de negócio nos documentos: pré-condições para planejamento de negócio/diretrizes para orçamento
Pr4	4	Comunicar políticas via folder ou intranet
Pr5	5	Revisar objetivos e indicadores
Pr6	6	Executar atividades dentro do domínio do suplente na competência
Pr6A	6A	Executar atividades dentro do domínio do suplente na competência
Pr7	7	Revisão dos objetivos e indicadores
Pr8	8	Retornar trimestralmente os objetivos e indicadores com a ajuda do Sistema de Revisão do Grupo
Pr9	9	Resumo dos objetivos alcançados e indicadores
Pr10	10	Comparar objetivos realizados e tendências com objetivos propostos de acordo com as políticas particulares e planejamento de negócio

Quadro 13 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Planejamento estratégico com proposta de modificação

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1
$C_{IP} C_S C_{OR}$	Regra 7

Quadro 14 - Conectores utilizados no Modelo de Planejamento estratégico com proposta de modificação de recursos humanos e a regra de mapeamento correspondente

Figura 57 - Simulação do Modelo com a proposta de modificação de Planejamento estratégico na ferramenta Petri Net Tools

Figura 58 - Árvore de Alcançabilidade do Modelo de Planejamento de Recursos humanos com a proposta de modificação

7.5.3. Modelo de Auditoria

O modelo da figura 59 descreve o processo de auditoria. As ações executadas de áreas do negócio relacionadas à competência são enviadas para a execução do processo realizar auditoria. Paralelamente ou não, são formulados os relatórios de auditoria e avaliação dos resultados. Em seguida, o plano de ação relacionado à melhoria dentro das áreas de negócio, as propostas de melhorias no grupo de suprimento de competência do processo e o plano de ação para atividades do grupo Vatterfal. Os processos desse modelo são: realizar auditoria anual, retornar para áreas de negócio para melhoria de processo, retornar sobre como o processo está funcionando, colecionar todas as auditorias por área de negócio e planejar ações para coordenar atividades relacionadas a competência.

Na figura 59 é apresentado o modelo original descrito. Na figura 60 é apresentado o modelo alterado. O modelo original também não atende ao requisito que afirma que o modelo deve ter um lugar início e um lugar fim. No modelo alterado foram adicionados os conectores.

Na figura 61 é apresentado o mesmo modelo mapeado em redes de Petri de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 15. Os conectores do Modelo de Processo de Auditoria e suas respectivas regras de mapeamento são apresentados no quadro 16.

Resultado do teste

Na figura 62 é apresentado o modelo simulado na ferramenta Petri Net Tools. A árvore de alcançabilidade apresentada na figura 63, demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 59 - Modelo de Processos de Negócio de Auditoria (Vatterfall) original

Figura 60 - Modelo de Processos de Negócio de Auditoria (*Vatterfall*) alterado

Figura 61 - Modelo de Processos de Negócio de Auditoria mapeando em redes de Petri

PN	MPN-EKD	
Lugar	Inf-set	Descrição
Is1	1	Ações executadas de áreas do negócio relacionadas a competência
Is1.1	1	Pré-condições para o trabalho planejamento de negócio
Is1.2	1.2	Políticas/diretrizes para o apoio de competências
Is2	2	Relatório da auditoria (retorno da avaliação dos processos)
Is3	3	Avaliação dos resultados
Is4	4	Plano de ação relacionado a melhoria dentro das áreas de negócio
Is5	5	Propostas para melhoria no grupo de suprimento da competência do processo
Is6	6	Compilando áreas do negócio/companhia excedente/necessidade de competência
Is7	7	Plano de ação para atividades do grupo Vatterfall relacionada à competência
Transição	Processo	Descrição
Pr1	1	Realizar auditoria anual para respectivas áreas de negócio
Pr2	2	Retornar para áreas de negócio para melhoria de processo
Pr3	3	Retornar sobre como o processo está funcionando
Pr4	4	Colecionar todas as auditorias por áreas de negócio
Pr5	5	Planejar ações para coordenar atividades relacionadas a competência.

Quadro 15 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Auditoria.

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_S C_{AND}$	Regra 5

Quadro 16 - Conectores do Modelo de Auditoria e as regras de mapeamento correspondente

Figura 62 - Simulação do Modelo de Auditoria na ferramenta Petri Net Tools

Figura 63 - Árvore de alcançabilidade do modelo de Auditoria

7.5.3. Modelo de Planejamento de Competências

O modelo da figura 64 descreve o planejamento de competências. Os processos do referido modelo são: continuar ações de áreas de negócio relacionadas à competência, formular objetivos do negócio para áreas do negócio, cumprir uma SWOT análise entre outras, formular estratégias, identificar as áreas de negócio de controle, identificar necessidades de competências de todas as áreas, escolher fatores indicadores chaves de métricas e formular objetivos para a área de controle. Para que esses processos sejam executados é necessário conhecer as propostas para o planejamento estratégico dentro do campo de competência, a análise do ambiente, a situação do mercado, relatório de ações, situações correntes considerando a disponibilidade de competência, análise dos competidores e objetivos da Vatterfal. Serão elaborados os objetivos de negócio por áreas, análise para competência, estratégias de alto nível para atingir objetivos do negócio, áreas de controle identificadas, todas as necessidades de competências como negociantes, líderes de projetos e desenvolvedores de produto, métricas e, finalmente, a situação corrente em áreas de negócio responsáveis pela seção de competência.

Na figura 64 é apresentado o modelo original. Na figura 65 é apresentado o modelo alterado para que ser possível testar o método criado neste trabalho. O modelo foi alterado por que o Pr6 e o Pr7 não atende ao requisito que afirma que a saída do processo deve ser igual a 1. Outra alteração foi no inf set 6.1 por que está conectado inf set 6 e essa conexão não é permitida. O modelo original também não atende ao requisito que afirma que o modelo deve ter um lugar início e um lugar fim. No modelo alterado foram adicionados os conectores. A alteração não modifica o fluxo do processo.

Na figura 66 é apresentado o mesmo modelo mapeado em redes de Petri de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 17. Os conectores do Modelo de Processo de Planejamento de Competências e suas respectivas regras de mapeamento são apresentados no quadro 18.

Resultado do teste

Na figura 67 é apresentado o modelo simulado na ferramenta Petri Net Tools. A árvore de alcançabilidade apresentada na figura 68 demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do

caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 64 - Modelo de Processos de Negócio Planejamento de Competências (Vatterfall) original

Figura 65 - Modelo de Processos de Negócio Planejamento de Competências (Vatterfall) alterado

Figura 66 - Modelo de Processos de Negócio de Planejamento de Competências mapeado em redes de Petri

PN	MPN-EKD	
Lugar	Inf-set	Descrição
Is1	1	Propostas para planejamento estratégico dentro do campo de competência
Is2	2	Análise do Ambiente
Is3	3	Situação do mercado
Is4	4	Situação corrente condiderando atitude
Is5	5	Relatório do ações
Is6	6	Objetivos Vatterfall
Is6.1	6.1	Pré-condições para o trabalho de planejamento de negócio
Is7	7	Análise dos competidores
Is8	8	Situação corrente considerando a disponibilidade de competência
Is9	9	Objetivos de negócio por áreas
Is10	10	SWOT análise para competência
Is11	11	Estratégias de alto nível para atingir objetivos de negócio
Is12	12	Áreas de controle identificada
Is13	13	Todas as necessidades de competências como negociantes, lideres de projetos e desenvolvedores de produtos
Is14	14	Métricas
Is15	15	Situação corrente em áreas de negócio responsáveis pela seção de competência
Transição	Processo	Descrição
Pr1	1	Continuar ações de áreas de negócio relacionadas a competência
Pr2	2	Formular objetivos do negócio para Áreas do Negócio
Pr3	3	Cumprir uma análise SWOT entre outros tipos
Pr4	4	Formular estratégias para atingirem objetivos
Pr5	5	Identificar as áreas de negócio de controle
Pr6	6	Identificar necessidades de competências de todas as áreas
Pr7	7	Escolher fatores indicadores chaves de métricas
Pr8	8	Criar objetivos para áreas de controle x% e Y itens

Quadro 17 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Planejamento de Competências

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1

Quadro 18 - Conectores do Modelo Planejamento de Competências e a regra de mapeamento correspondente

Figura 67 - Simulação do Modelo de Planejamento de Competências na ferramenta Petri Net Tools

Figura 68 - Árvore de Alcançabilidade do Modelo de Processos de Negócio de Planejamento de Competências

7.5.4. Modelo de Finalização de Scorecard

O modelo da figura 69 descreve a finalização do *scorecard*. Os processos do referido modelo são: ajustar *scorecard*, finalizar *scorecard* balanceado para o grupo, compilar base de dados para o quadro de diretores e finalizar *scorecard* e financiar previsão financeira para áreas de negócio e para o grupo. Para que o processo ajustar *scorecard* para o grupo seja iniciado é necessário ter: a iniciativa para *scorecard* para anos 1-5, o plano de negócio finalizado e a previsão econômica, e, finalmente, o *scorecard* balanceado das áreas do negócio finalizado.

Na figura 69 é apresentado o modelo original e na figura 70 é apresentado o modelo alterado. O modelo foi alterado por que o *inf-set3* não atende ao requisito que afirma que a saída do *inf-set* deve ser igual a 1. O modelo original também não atende ao requisito que afirma que o modelo deve ter um lugar início e um lugar fim. No modelo alterado foram adicionados os conectores. A alteração não modifica o fluxo do processo.

Na figura 71 é apresentado o modelo alterado mapeado em redes de Petri, de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 19. Os conectores do Modelo de Finalização de *Scorecard* e suas respectivas regras de mapeamento são apresentados no quadro 20.

Resultado do teste

Na figura 72 é apresentado o modelo simulado na ferramenta Petri Net Tools. A árvore de alcançabilidade apresentada na figura 73 demonstra que o modelo (depois de alterado) é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 69 - Modelo de Processos de Negócio Finalizar *scorecard* (Vatterfall) original com acrescimo dos conectores

Figura 70 - Modelo de Processos de Negócio Finalizar *scorecard* (Vatterfall) alterado

Figura 71 - Modelo de Processos de Negócio Finalizar *scorecard* mapeado em redes de Petri

PN	MPN-EKD	
Lugar	Inf-set	Descrição
Is1	1	Iniciativa para scorecard para anos 1-5
Is2	2	Plano de negócio finalizado e previsão econômica
Is3	3	Scorecard balanceado das áreas de negócio finalizado
Is4	4	Scorecard do grupo ajustado
Is5	5	Scorecard balanceado do grupo finalizado
Is6	6	Base de dados compilada para quadro de diretores
Is7	7	Scorecard finalizado pelo quadro de diretores e previsão econômica para as respectivas áreas de negócio e para o grupo, incluindo seção sobre empregado e competência
Transição	Processo	Descrição
Pr1	1	Ajustar scorecard para o grupo
Pr2	2	Finalizar scorecard balanceado para o grupo
Pr3	3	Compilar base de dados para o quadro de diretores
Pr4	4	Finalizar scorecard e financiar previsão financeira para as áreas de negócio e para o grupo

Quadro 19 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Finalizar scorecard.

Conector	Regra correspondente
$C_{IP} C_J C_{AND}$	Regra 1
$C_{PI} C_S C_{AND}$	Regra 6

Quadro 20 - Conectores do modelo Finalizar scorecard e a regra de mapeamento correspondente

Figura 72 - Simulação do Modelo Finalizar *scorecard* na ferramenta Petri Net Tools

Figura 73 - Árvore do Modelo de Processos de Negócio Finalizar *scorecard*

7.5.5. Modelo de Planejamento Estratégico

O modelo da figura 74 descreve o planejamento estratégico. Para iniciar o processo é necessário conhecer: a descrição da situação corrente de quantidade de competência, o plano estratégico dentro do campo personalizado e os resultados de competências implementadas relacionadas a ações. Os processos desse modelo são: projetar uma proposta para o Vatterfal, propor plano estratégico na reunião e planejamento, finalizar plano estratégico e *scorecard*, finalizar plano estratégico com quadro de diretores, projetar pré-condições para trabalho de planejamento de negócio, negociar com uniões sobre pré-condições para o trabalho de planejamento de negócio e enviar as pré-condições para BA/companhia.

Na figura 74 é apresentado o modelo original. Na figura 75 é apresentado o modelo alterado. O modelo foi modificado por que o modelo não atende ao requisito que afirma

que o modelo deve ter um lugar início e um lugar fim. No modelo alterado foram adicionados os conectores. Essa alteração não modifica o fluxo do processo.

Na figura 76 é apresentado o modelo alterado mapeado em redes de Petri, de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 21. Os conectores do Modelo de planejamento estratégico e as regras de mapeamento utilizadas são apresentados no quadro 22.

Resultado do teste

Na figura 77 é apresentado o modelo simulado na ferramenta Petri Net Tools. Na figura 78 é apresentada a árvore de alcançabilidade. A árvore demonstra que o modelo (depois da alteração) é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 75 - Modelo de Processos de Negócio Projeto de Plano Estratégico (*Vattenfall*) alterado

Figura 76 - Mapeamento do Modelo de Processos de Negócio Projeto de Plano Estratégico em redes de Petri

PN	MPN-EKD	
Lugar	Inf-set	Descrição
Is1	1	Descrições da situação corrente de quantidade de competência
Is2	2	Plano estratégico dentro do campo personalalizado
Is3	3	Resultados de competências implementadas relacionadas a ações
Is4	4	Outras entradas
Is5	5	Proposta para planejamento estratégico incluindo Scorecard balanceado
Is6	6	Proposta para planejamento estratégico incluindo scorecard
Is7	7	Plano estratégico finalizado
Is8	8	Iniciativa para scorecard para anos 1-5
Is9	9	Plano estratégico finalizado
Is10	10	Pré-condições para o trabalho de Planejamento de Negócio
Is11	11	Pré-condições para o trabalho de planejamento de negócio
Is12	12	Pré-condições enviadas
Transição	Processo	Descrição
Pr1	1	Projetar uma proposta para Vatterfall Group Design para scorecard ano 1-5
Pr2	2	Propor plano estratégico na reunião de planejamento
Pr3	3	Finalizar plano estratégico e scorecard
Pr4	4	Finalizar plano estratégico com o quadro de diretores
Pr5	5	Projetar pré-condições para trabalho de planejamento de negócio
Pr6	6	Negociar com uniões sobre pré-condições para o trabalho de planejamento de negócio
Pr7	7	Enviar as pré-condições para o BA/companhia

Quadro 21 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Projeto de Plano Estratégico

Conector	Regra correspondente
$C_{IP} C_J C_{AND}$	Regra 1
$C_{PI} C_S C_{AND}$	Regra 6

Quadro 22 - Conectores do Modelo Projeto Plano Estratégico e a regra de mapeamento correspondente

Figura 77 - Simulação do Modelo Projeto de Plano estratégico na ferramenta Petri Net Tools

Figura 78 - Árvore de alcançabilidade Modelo de Processos de Negócio Projeto de Plano Estratégico

7.6. Caso 4 – Delegacia Regional Tributária de Araraquara

Os modelos que serão apresentados a seguir fazem parte do estudo de caso de Pinotti (2004). Os modelos representam um estudo de caso no âmbito da Delegacia Regional Tributária de Araraquara, da Secretaria da Fazenda do Estado de São Paulo. O propósito dos modelos é especificar um sistema para o gerenciamento das ausências de funcionários devido a férias, licenças ou outros fatores, e das respectivas substituições

desse funcionários, no período em que estão ausentes, caso seja necessário. O resumo do contexto do estudo de caso contido em Pinotti (2004) será apresentado a seguir.

No ambiente da delegacia existem dois tipos de funcionários: os que executam funções externas (chamados de “externos”) e os que executam funções internas (chamados de “internos”). Os externos são organizados em equipes de fiscalização e executam tarefas relativas à fiscalização de empresas da região. Os internos trabalham nas diversas unidades administrativas da delegacia e seu trabalho refere-se ao andamento dos processos na secretaria.

Cada funcionário tem direito a 30 dias de férias a cada ano, que podem ser usufruídos num único período ou em dois períodos de 15 dias. Dessa forma, ao final de cada ano, os chefes das unidades devem elaborar uma escala de férias dos funcionários de sua unidade para o ano seguinte. Na elaboração dessa escala, deve-se considerar os cursos eventualmente programados para o ano seguinte e as previsões de licenças que serão tiradas pelos funcionários da unidade. Assim, são elaboradas duas escalas: a de férias e a de licenças.

A escala de férias é formal, sendo que os períodos informados na escala são fixos e existe um procedimento para alterar esses períodos, caso seja necessário. Já a escala de licenças é informal, constituindo-se apenas numa previsão. O funcionário que for tirar licença faz a solicitação no período em que for usufruir a mesma, ainda que esse período seja diferente daquele informado na escala. Assim, para amenizar os problemas decorrentes de solicitações de licenças, relativos à necessidade de substituição do respectivo funcionário, elabora-se, juntamente com a escala de férias, a escala de licenças para o ano seguinte. Porém, nada impede que o funcionário solicite uma licença, desde que tenha direito à mesma, fora do prazo previsto na escala.

Existem funções internas que precisam de substituição, ou seja, quando o funcionário ocupante da função está em férias, ou em licença, ou ausente por qualquer outro motivo, é necessário que outro funcionário o substitua na função. São funções dessa natureza, por exemplo, todas as funções de chefia de unidade e a de delegado. Existem, também, funções que não precisam de substituição. Nesses casos, a unidade fica sem o ocupante para a função enquanto o funcionário está ausente.

O funcionário substituto pode ser tanto interno quanto externo. Porém, na maioria das vezes é externo, pois para substituir um funcionário por outro também interno, outra

função (a do funcionário substituto) ficaria desocupada. Dessa forma, todas as substituições nos postos fiscais são feitas por funcionários externos, ou em alguns casos, por um funcionário interno do próprio posto. Já nas outras unidades, a substituição varia. No Núcleo de Informações, por exemplo, na ausência do chefe, um dos assistentes passa a substituir o chefe e a unidade fica com um funcionário a menos.

O fator substituição deve ser considerado na elaboração das escalas de férias das equipes. É necessário haver funcionários nas equipes para substituir as funções internas, além da necessidade de se manter um número mínimo de funcionários na equipe para o andamento dos trabalhos.

No caso de pedido de alteração de férias, o procedimento é similar ao de elaboração das escalas de férias. Caso seja um funcionário interno, o superior hierárquico é responsável pela aprovação da alteração.

Quando o delegado optar por aprovar as escalas de férias e alterações em algum ano, determina que todas as escalas de férias sejam aglutinadas numa única planilha, que usa para fazer consultas para escolher funcionários substitutos para as funções internas ou para tomar decisões sobre as alterações de férias solicitadas. A cada alteração de férias aprovada, essa planilha deve ser atualizada. Isso gera um problema, pois essas alterações não são atualizadas na planilha imediatamente, pois dependem de serem enviadas ao funcionário responsável por essa tarefa e também da disponibilidade do funcionário. Com isso, freqüentemente o delegado não tem informações atualizadas para tomar decisões.

Todo esse processo é regulado anualmente. Assim, no final de cada ano, o delegado emite um memorando circular contendo as instruções para a elaboração das escalas de férias e alterações posteriores para o ano seguinte.

7.6.1. Modelo de Elaboração de Escalas

Na figura 79 é apresentado o modelo de elaboração de escalas original do trabalho de Pinotti (2004). O modelo da figura 79 apresenta muitas situações de travamento e por isso foi dividido em dois modelos (figura 80 e 84) e foram realizadas algumas modificações em cada modelo.

O modelo foi modificado por vários motivos:

- Os processos 1 e 2 não atendem ao requisito que afirma que todo conector do tipo *join* deve ter a saída igual a 1. Para resolver esse problema, os processos foram agrupados (figura 80).
- O *inf-set escala de férias* da unidade tem duas saídas e não atende ao requisito que afirma que a saída do *inf-set* deve ser igual ou menor que 1.
- O *inf-set Escala de licenças* tem três saídas: processo 3, processo 4 e para o processo 14 (atualizar planilha de ausência), por isso não atende ao requisito que afirma que a saída do *inf-set* deve ser igual ou menor que 1. Neste caso, foi direcionado ao processo 4 (Enviar ao Gabinete) e quando a escala estiver aprovada será enviada ao processo 3 e 14.
- O processo 13 está com a saída para *inf-set escala de férias* que tem mais de uma entrada e cinco saídas, não atendendo aos requisitos de nenhum tipo de conector. Dessa forma, a saída do processo 13 foi direcionada para o *inf-set 11 escala de férias enviada* (figura 84).
- O processo 14 tem três entradas, o que contraria o requisito que afirma que a entrada do processo deve ser igual a 1. Esse problema foi resolvido na divisão dos modelos.
- Os processos 12 e 11 foram agrupados por que eles têm duas saídas em comum e uma saída apenas do processo 11, não sendo permitido esse tipo de construção com o conector *join* e *split* (figura 84).
- O processo 11 tinha as duas saídas do processo 12 e mais uma saída que é o *inf-set* função interna não ocupada. Como os processos 12 e 11 foram agrupados, foi criado um novo processo: analisar se a função interna é ocupada ou não. Esse processo tem dois *inf-sets* de saída: *função interna não ocupada* ou *função interna ocupada* (figura 84).
- O *inf-set substituições planejadas* tem duas saídas: processo 2 e 12. Não é permitida essa construção por que esse *inf-set* conecta-se ao conector AND (que conecta ao processo 1 e 2) e ao processo 12. Na divisão do modelo o *inf-set substituições planejadas* apenas conecta-se ao processo 12, uma vez que esse *inf-set* vai sempre causar *deadlock* por que necessita da execução do processo 3 *planejar substituições de funções* que será executado depois do processo 1 e 2.

7.6.2. Modelo de Elaboração de Escalas modificado (parte 1)

Para iniciar o processo do modelo da figura 79 é necessário conhecer a previsão de licenças, o modelo de escalas de férias, a programação de cursos, preferências dos funcionários e as substituições planejadas. Os processos do modelo modificado (parte 1) apresentados na figura 80 são: alterar as escalas da unidade e da equipe, enviar ao gabinete, analisar escala de férias, refazer escala de férias, arquivar escala de férias no gabinete, enviar escala aprovada à unidade, arquivar escala de férias na unidade, enviar ao departamento de RH e atualizar planilha de ausência. O modelo mapeado em redes de Petri é apresentado na figura 81.

Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 23. Os conectores do Modelo de Processo de Elaboração de Escalas e suas respectivas regras de mapeamento são apresentados no quadro 24.

Resultado do teste

Na figura 82 é apresentado o modelo simulado na ferramenta Petri Net Tools. A árvore de alcançabilidade é apresentada na figura 83. A árvore demonstra que o modelo (depois de modificado) é seguro, não tem *deadlock*, *livelock*, tem lugar início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, assim, o modelo é *sound*.

Figura 79- Modelo de Processos de Negócio de Elaboração de Escalas original

Fonte: Pinotti (2004), p. 61

Figura 80 -Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 1

Figura 81 -Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 1 mapeado em redes de Petri

PN	MPN-EKD	
Lugar	Inf-set	Descrição
Is1	1	Previsão de licença
Is2	2	Modelo de Escala de Férias
Is3	3	Programação de cursos
Is4	4	Preferência dos funcionários
Is5	5	Substituições Planejadas
Is8	8	Escala de férias da Unidade
Is9	9	Escala de licenças
Is10	10	Escala de férias da equipe
Is11	11	Escala de férias enviada
Is12	12	Alterações Necessárias
Is13	13	Escala de férias
Is14	14	Escala de férias arquivadas
Is15	15	Escala de férias enviada a unidade
Is16	16	Planilha de ausências
Is17	17	Função interna não-ocupada
Is18	18	Função interna ocupada
Is19	19	Pedido de alteração negado
Transição	Processo	Descrição
Pr1 e 2	1 e 2	Elaborar escalas da unidade e Elaborar escala de férias
Pr4	4	Enviar ao gabinete
Pr5	5	Analisar escala de férias
Pr6	6	Enviar escala aprovada a unidade
Pr7	7	Arquivar escala de férias no gabinete
Pr8	8	Enviar ao departamento de RH
Pr9	9	Arquivar escala de férias na unidade
Pr10	10	Afixar escala de férias no gabinete
Pr14	14	Atualizar planilha de ausências

Quadro 23 - Elementos de redes de Petri correspondentes aos elementos do MPN_EKD de Elaboração de Escalas

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1
$C_{PI} C_J C_{OR}$	Regra 4
$C_{PI} C_S C_{OR}$	Regra 8
$C_{IP} C_S C_{AND}$	Regra 5

Quadro 24 - Conectores do Modelo de Elaboração de Escalas e as regras de mapeamento correspondente

Figura 82 - Simulação do Modelo de Processos de Negócio de Elaboração de Escalas na ferramenta Petri Net Tools

Figura 83 - Árvore de Alcançabilidade do Modelo de Elaboração de Escalas parte 1

7.6.3. Modelo de Elaboração de Escalas modificado (parte 2)

O modelo de Processos de Negócio de Elaboração de Escalas modificado (parte 2) é apresentado na figura 84. Para iniciar é necessário ter a escala de férias original, relação de aposentadorias e ausências, pedidos e necessidade de alteração de férias.

Os processos do modelo modificado (parte 2) apresentado na figura 84 são: planejar substituições de funções internas, analisar se a função interna é ocupada, analisar pedido/necessidade de alteração de equipe e função interna, enviar ao departamento de RH e atualizar planilha de ausência.

O modelo de processos de negócio mapeado em redes de Petri é apresentado na figura 85. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 25. Os conectores do Modelo de Processo de Elaboração de Escalas modificado (parte 2) e suas respectivas regras de mapeamento são apresentados no quadro 26.

Resultado do teste

Na figura 86 é apresentado o modelo na ferramenta Petri Net Tools. A simulação não foi possível de ser realizada por causa do *deadlock* na Pr3. O Pr3 só poderá ser executado quando houver marca no IS17. Para que o processo planejar substituições de funções internas seja realizado é necessário conhecer se a função interna não é ocupada. A árvore de alcançabilidade é apresentada na figura 87 e demonstra que existe um *deadlock*. O modelo não é *sound*. Os erros encontrados no modelo estão relacionados com o planejamento e ordem de execução das atividades da organização.

Figura 84 -Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 2

Figura 85 - Modelo de Processos de Negócio de Elaboração de Escalas modificado parte 2 mapeado em redes de Petri

PN	MPN-EKD	
Lugar	Inf-set	Descrição
Is5	5	Substituições Planejadas
Is6	6	Aposentadoria e outras ausências
Is7	7	Pedido/Necessidade de alteração de férias
Is14	14	Escala de férias arquivadas
Is16	16	Planilha de ausências
Is17	17	Função interna não-ocupada
Is18	18	Função interna ocupada
Is19	19	Pedido de alteração negado
Transição	Processo	Descrição
Pr3	3	Planejar substituições de funções internas
Pr8	8	Enviar ao departamento de RH
Pr14	14	Atualizar planilha de ausência
Pr11.1	11.1	Analisar se a função interna é ocupada
Pr12 e 11	12 e 11	Analisar pedido/necessidade de alteração de equipes e função interna

Quadro 25 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Elaboração de Escalas modificado parte 2

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1
$C_{IP} C_S C_{AND}$	Regra 5
$C_{PI} C_S C_{OR}$	Regra 8
$C_{PI} C_J C_{AND}$	Regra 7
$C_{PI} C_J C_{OR}$	Regra 4

Quadro 26 - Conectores do Modelo de Elaboração de Escalas modificado parte 2 e as regras de mapeamento correspondentes

Figura 86 - Simulação do Modelo de Processos de Negócio de Elaboração de Escalas na ferramenta Petri Net Tools

Figura 87 - Árvore de Alcançabilidade do Modelo de Escalas modificado parte 2

7.6.4. Modelo de Planejamento de Escalas da Unidade

O modelo da figura 88 trata o processo de Planejamento das Escalas da Unidade original de Pinotti (2004). Na figura 89 é apresentado o modelo alterado. Os processos desse modelo são priorizar escolhas, negociar com funcionários, preencher escalas e visar escala de unidade subordinada. Para iniciar é necessário ter as preferências dos funcionários, previsão de licenças e programação de cursos. Na figura 90 é apresentado o modelo alterado mapeado em redes de Petri, de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-

EKD são apresentados no quadro 27. Os conectores do Modelo de Processo de Planejamento de Escalas da Unidade e suas respectivas regras de mapeamento são apresentados no quadro 28.

Resultado do teste

Na figura 91 é apresentado o modelo simulado na ferramenta Petri Net Tools. Na figura 92 é apresentada a árvore de alcançabilidade. A árvore de alcançabilidade demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 88 - Modelo de Processos de Negócio Planejamento das Escalas da Unidade original

Figura 89 - Modelo de Processos de Negócio Planejamento das Escalas da Unidade

Fonte: Pinotti (2004), p. 62.

Figura 90 - Modelo de Planejamento de Escalas da unidade mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>Inf-set</i>	Descrição
Is1	1	Preferência dos funcionários
Is2	2	Previsão de licenças
Is3	3	Programação de cursos
Is4	4	Preferências não atendidas
Is5	5	Escala de licenças planejadas
Is6	6	Escala de licenças
Is7	7	Modelo de Escala de férias
Is8	8	Escala de férias da unidade subordinada
Is9	9	Escala de licenças
Is10	10	Escala de férias da unidade
Transição	Processo	Descrição
Pr1	1	Priorizar escolhas
Pr2	2	Negociar com funcionários
Pr3	3	Preencher escalas
Pr4	4	Visar escala de unidade subordinada

Quadro 27 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Planejamento das Escalas da Unidade

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1

Quadro 28 - Conectores do Modelo de Elaboração de Escalas da Unidade e as regras de mapeamento correspondente

Figura 91 - Simulação do Modelo de Planejamento de Escalas da Unidade na ferramenta Petri Net Tools

Figura 92 - Árvore de Alcançabilidade do modelo Planejamento de Escalas da Unidade

7.6.5. Planejamento das Escalas das Equipes

O modelo da figura 93 trata o processo de Planejamento das Escalas das Equipes original. Os processos desse modelo são: priorizar escolhas, negociar com funcionários e preencher escalas. Para iniciar é necessário conhecer as substituições planejadas, as preferências dos funcionários, a previsão de licenças e a programação de cursos. Na figura 94 é apresentado alterado. Na figura 95 é apresentado o modelo alterado mapeado em redes de Petri, de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 29. Os conectores do Modelo de Processo de Planejamento de Escalas das Equipes e suas respectivas regras de mapeamento são apresentados no quadro 30.

Resultado do teste

Na figura 96 é apresentado o modelo simulado na ferramenta Petri Net Tools. Na figura 97 é apresentada a árvore de alcançabilidade.

A árvore de alcançabilidade demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 93 - Modelo de Processos de Negócio de Planejamento das Escalas das equipes original
 Fonte: Pinotti (2004), p. 63.

Figura 94 - Modelo de Processos de Negócio de Planejamento das Escalas das equipes alterado

Figura 95 - Modelo de Planejamento das Escalas das equipes mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>Inf-set</i>	Descrição
Is1	1	Substituições planejadas
Is2	2	Preferências dos funcionários
Is3	3	Previsão de licenças
Is4	4	Programação de cursos
Is5	5	Modelo de Escala de Férias
Is6	6	Preferências não atendidas
Is7	7	Escala de férias planejadas
Is8	8	Escala de licenças planejadas
Is9	9	Escala de licenças
Is10	10	Escala de férias da equipe
Transição	Processo	Descrição
Pr1	1	Priorizar escolhas
Pr2	2	Negociar com funcionários
Pr3	3	Preencher escalas

Quadro 29 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Planejamento das Escalas das equipes

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1

Quadro 30 - Conectores do Modelo de Planejamento das Escalas das equipes e as regras de mapeamento correspondente

Figura 96 – Simulação do Modelo de Processos de Negócio de Planejamento das Escalas das equipes na ferramenta Petri Net Tools

Figura 97 - Arvore do Modelo de Planejamento das Escalas das equipes

7.6.6. Modelo de Planejamento das substituições de funções internas

O modelo da figura 98 trata o processo de Planejamento das substituições de funções internas original. Os processos desse modelo são: verificar se a função vaga precisa de substituição e escolher funcionário para substituir. Para iniciar é necessário conhecer as escalas de licenças, escala de férias da unidade, função interna não ocupada e aposentadorias e outras ausências. Na figura 99 é apresentado o modelo alterado para ser realizado o teste do método de avaliação desenvolvido neste trabalho. Na figura 100 é apresentado o modelo alterado mapeado em redes de Petri, de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 31. Os conectores do Modelo de Processo de Planejamento das substituições de funções internas e suas respectivas regras de mapeamento são apresentados no quadro 32.

Resultado do teste

Na figura 101 é apresentado o modelo simulado na ferramenta Petri Net Tools. Na figura 102 é apresentada a árvore de alcançabilidade. A árvore de alcançabilidade demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 98 - Modelo de Processos de Negócio de Planejamento das substituições de funções internas original

Fonte: Pinotti (2004), p. 64.

Figura 99 - Modelo de Processos de Negócio de Planejamento das substituições de funções internas alterado

Figura 100 - Modelo de Processos de Negócio de Planejamento das substituições de funções internas mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>Inf-set</i>	Descrição
Is1	1	Escala de licenças
Is2	2	Escala de férias da unidade
Is3	3	Função interna não ocupada
Is4	4	Aposentadoria e outras ausências
Is5	5	Substituição necessária
Is6	6	Substituição não-necessária
Is7	7	Substituições planejadas
Transição	Processo	Descrição
Pr1	1	Verificar se a função vaga precisa de substituição
Pr2	2	Escolher funcionário para substituir

Quadro 31 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Planejamento das substituições de funções internas

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1
$C_{IP} C_J C_{OR}$	Regra 3
$C_{PI} C_S C_{OR}$	Regra 8

Quadro 32 - Conectores do Modelo de Planejamento das substituições de funções internas e as regras de mapeamento correspondente

Figura 101 - Simulação do Modelo de Processos de Negócio de Planejamento das substituições de funções internas na ferramenta Petri Net Tools

Figura 102 - Árvore de Alcançabilidade do Modelo de Planejamento das substituições de funções internas

7.6.7. Modelo de Análise de pedido de Alteração de Férias

O modelo da figura 103 trata o processo de análise de pedido/necessidade de alteração interna original. Os processos desse modelo são: elaborar ofício de alteração de férias, decidir sobre pedido de alteração, manifestar-se sobre pedido de alteração, negar pedido de alteração e enviar ao gabinete para conhecimento. Para iniciar é necessário conhecer pedido ou necessidade de alteração de férias e planilha de ausências. Na figura 104 é apresentado o modelo alterado para que seja possível a realização do teste. Os processos 11.2 e 11.4 eram repetidos e por isso o processo 11.2 foi eliminado.

Na figura 105 é apresentado o modelo alterado mapeado em redes de Petri, de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 33. Os conectores do Modelo de Processo de Planejamento das substituições de funções internas e suas respectivas regras de mapeamento são apresentados no quadro 34.

Resultado do teste

Na figura 106 é apresentado o modelo simulado na ferramenta Petri Net Tools. Na figura 107 é apresentada a árvore de alcançabilidade. A árvore de alcançabilidade demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 103 - Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna original
Fonte: Pinotti (2004), p. 65.

Figura 104 - Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna alterado

Figura 105 - Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>Inf-set</i>	Descrição
Is1	1	Pedido/necessidade de alteração de férias
Is2	2	Ofício de alteração de férias
Is3	3	Ofício com manifestação
Is4	4	Planilha de ausências
Is5	5	Alteração autorizada - comunicado de de alteração de férias
Is6	6	Alteração não autorizada
Is7	7	Função interna não ocupada
Is8	8	Pedido de alteração negado
Transição	Processo	Descrição
Pr1	1	Elaborar ofício de alteração de férias
Pr2	2	Manifestar-se sobre pedido de alteração
Pr3	3	Decidir sobre pedido de alteração
Pr4	4	Enviar ao gabinete para conhecimento
Pr5	5	Negar pedido de alteração

Quadro 33 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD Análise de pedido/necessidade de alteração interna

Conector	Regra correspondente
$C_{IP} C_J C_{AND}$	Regra 1
$C_{IP} C_J C_{OR}$	Regra 3
$C_{PI} C_S C_{OR}$	Regra 8

Quadro 34 - Conectores do Modelo de Análise de pedido/necessidade de alteração interna e as regras de mapeamento correspondente

Figura 106 - Simulação do Modelo de Processos de Negócio de Análise de pedido/necessidade de alteração interna

Figura 107 - Árvore de alcançabilidade do Modelo de Análise de pedido/ necessidade interna

7.6.8. Modelo de Análise de Pedido de Alteração de Equipe

O modelo da figura 108 trata o processo de análise de pedido de alteração de equipe original. Os processos desse modelo são: elaborar ofício de alteração de férias, manifestar-se sobre pedido de alteração, decidir sobre pedido de alteração de férias, negar pedido de alteração e enviar ao gabinete. Para iniciar é necessário conhecer: o pedido ou necessidade de alteração de férias, as substituições planejadas e a planilha de ausências. Na figura 109 é apresentado o modelo alterado para que o teste seja realizado.

Na figura 110 é apresentado o modelo alterado mapeado em redes de Petri, de acordo com o método desenvolvido neste trabalho. Os elementos de Redes de Petri correspondentes aos elementos do MPN-EKD são apresentados no quadro 35. Os

conectores do Modelo de Processo de Planejamento das substituições de funções internas e suas respectivas regras de mapeamento são apresentados no quadro 36.

Na figura 111 é apresentado o modelo simulado na ferramenta Petri Net Tools. Na figura 112 é apresentada a árvore de alcançabilidade. A árvore de alcançabilidade demonstra que o modelo é seguro, não tem *deadlock*, *livelock*, não apresenta tarefas que não contribuem para o processamento do caso, tem lugar de início e fim bem definidos, no final do caso existe apenas uma marca no lugar fim, ou seja, o modelo é *sound*.

Figura 108 - Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe original
 Fonte: Pinotti (2004), p. 66.

Figura 109 - Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe alterado

Figura 110 - Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe mapeado em redes de Petri

PN	MPN-EKD	
Lugar	<i>Inf-set</i>	Descrição
Is1	1	Pedido/Necessidade de alteração de férias
Is2	2	Substituições planejadas
Is3	3	Planilha de ausências
Is4	4	Ofício de alteração de férias
Is5	5	Ofício com manifestação do inspetor
Is6	6	Alteração não autorizada
Is7	7	Alteração autorizada
Is8	8	Comunicado de alteração de férias
Is9	9	Pedido de alteração negado
Transição	Processo	Descrição
Pr1	1	Elaborar ofício de alteração de férias
Pr2	2	Manifestar-se sobre pedido de alteração
Pr3	3	Decidir sobre pedido de alteração de férias
Pr4	4	Negar pedido de alteração
Pr5	5	Enviar ao gabinete

Quadro 35 - Elementos de Redes de Petri correspondentes aos elementos do MPN-EKD de Análise de Pedido de alteração de férias de equipe

Conector	Regra correspondente
$C_{PI} C_S C_{AND}$	Regra 6
$C_{IP} C_J C_{AND}$	Regra 1
$C_{IP} C_J C_{OR}$	Regra 3
$C_{PI} C_S C_{OR}$	Regra 8

Quadro 36 - Conectores do Modelo de Análise de Pedido de alteração de férias de equipe e as regras de mapeamento correspondente

Figura 111 - Simulação do Modelo de Processos de Negócio de Análise de Pedido de alteração de equipe na ferramenta Petri Net Tools

Figura 112 - Árvore de Alcançabilidade do Modelo de Análise de Pedido de alteração de equipe

7.7. Considerações Finais

Foram apresentados testes do método de avaliação do Modelo de Processos de Negócio. Os modelos foram simulados na ferramenta Petri Net Tools. As árvores de alcançabilidade foram desenvolvidas e analisadas. A primeira versão do método de avaliação do Modelo de Processos de Negócio do EKD mostrou-se importante para evitar ambigüidades e confusão nos modelos de processos de negócio.

CAPÍTULO 8

Conclusão

8.1. Considerações Iniciais

Neste trabalho, a modelagem organizacional é considerada como um passo importante para se conhecer a estrutura do negócio, para se discutir mudanças e estruturação de suas regras. Estas representam requisitos que governam o sistema operacional da organização e determinam como o negócio é executado. As regras do negócio são originadas dos objetivos organizacionais.

Foi enfatizado neste trabalho que as organizações devem criar novas formas de trabalhar para que sobrevivam no atual ambiente competitivo. Nesse contexto, considerou-se que modelar o conhecimento organizacional contribui para o entendimento da situação corrente e se estabelece, assim, uma visão do futuro sistema. Portanto, modelar a organização torna-se um pré-requisito muito importante para o desenvolvimento de sistemas de informação. Este trabalho mostrou que poucas abordagens investigam o aspecto dinâmico da modelagem do conhecimento, focando na automação dos processos existentes. Buscando suprir tal deficiência, o método EKD foi escolhido para este trabalho porque dentre todas as abordagens encontradas na literatura, foi o único a abordar gerenciamento de mudanças e tratar regras do negócio explicitamente em um modelo e disponibilizar um conjunto de diretrizes para sua aplicação. O desenvolvimento de cada um dos sub-modelos é apoiado por um conjunto de questões. Foi apresentado que o método EKD é independente de uma ferramenta de modo que os modelos podem ser desenvolvidos em qualquer editor de texto ou manualmente. Este trabalho mostrou que a abordagem EKD proporciona uma forma estruturada de

descrever o conhecimento organizacional e auxilia os desenvolvedores de sistemas de informação e *stakeholders* na determinação dos requisitos e objetivos do sistema.

8.2. Conclusão

Neste trabalho, foi destacado que o principal problema das abordagens de Modelagem Organizacional, incluindo-se o EKD, é a ausência de técnicas de análise objetivas. As técnicas de análise com rigor matemático não são usuais para profissionais da área de negócio. Este trabalho mostrou que as redes de Petri resolvem esse problema, uma vez que elas possuem representação gráfica, são de fácil aprendizado, funcionam como linguagem de comunicação entre especialistas de diversas áreas, permitem a descrição dos aspectos estáticos e dinâmicos do sistema a ser representado, e ainda possuem o formalismo matemático que permite a utilização de importantes métodos de análise.

Os processos de negócio, normalmente, têm uma estrutura simples antes de serem introduzidos nos sistemas de informação avançados, tais como sistema de *workflow*. Essa simplicidade é devida principalmente ao fato de que um documento (papel) pode estar apenas em um lugar em um mesmo momento. O documento atua como um conjunto de marcas que asseguram a execução seqüencial das tarefas. Atualmente, após vários anos de desenvolvimento de sistemas de forma seqüencial, é possível modelar processos de forma completamente diferente, uma vez que as informações e os dados podem ser compartilhados. Várias pessoas podem trabalhar ao mesmo tempo no mesmo caso. Por essa razão nem sempre é possível que as tarefas sejam realizadas seqüencialmente. Graças à utilização de processos de negócio paralelos é possível conseguir enormes reduções no tempo de execução. O ambiente de negócio é propício para executar as tarefas em paralelo de acordo com a necessidade. Contudo o uso de rotas seqüencial, paralela, seletiva e iterativa no mesmo processo torna difícil a avaliação dos processos definidos.

Neste sentido, foi enfatizado neste trabalho que o Modelo de Processos de Negócio deve ser desenvolvido com muito cuidado, pois, além de os problemas resultantes de erros no projeto serem difíceis de detectar, os custos da correção dos erros são altos. As ambigüidades e conflitos devem ser eliminados dos modelos.

Este trabalho mostrou que ambigüidades e confusões não podem ser prevenidas em um modelo de Processos de Negócio informal. Para resolver este problema, foi

desenvolvido o Modelo de Processos de Negócio com uma semântica formal. Para desenvolver essa semântica foi criado um conjunto de conectores para o Modelo de Processos de Negócio do EKD. O conjunto de conectores é representado por C e é composto por C_{AND} , C_{OR} , C_J , C_S , C_{IP} e C_{PI} . Os conectores C_{OR} e C_{AND} são importantes para identificar escolha (exclusiva) e paralelismo para que os casos de paralelismo e escolha não sejam modelados exatamente da mesma forma, evitando ambigüidades e dificuldades de compreensão. Os conectores C_J e C_S definem conectores do tipo *join* e *split*. Os conectores C_{IP} e C_{PI} demonstram que um conector c é um caminho de um *inf-set* para um processo ou um caminho de um processo para um *inf-set*.

Foram incluídos os estados inicial e final para possibilitar que a formalização fosse efetivamente realizada. Esses estados não são especificados no Modelo de Processos de Negócio do EKD original.

Foi desenvolvido neste trabalho um procedimento de mapeamento formal do Modelo de Processos de Negócio em redes de Petri. O procedimento de mapeamento foi desenvolvido baseado em redes de Petri lugar/transição. Por meio de um modelo de processos de negócio mapeado em redes de Petri de acordo com este procedimento, foi possível verificar alguns requisitos que garantem se o processo foi modelado corretamente e outros requisitos que permitem uma análise do processo.

Nesse sentido, a partir do procedimento de mapeamento formal do Modelo de Processos de Negócio em redes de Petri foi criada a primeira versão do Método de Avaliação do Modelo de Processos de Negócio do EKD. O método consiste em uma seqüência de passos que vai desde o desenvolvimento do modelo organizacional até a construção da árvore de alcançabilidade e simulação do modelo em ferramenta.

A aplicação da primeira versão do Método de Avaliação do Modelo de Processos de Negócio de redes de Petri permite verificar a presença de alguma(s) dessas situações indesejáveis:

- 1 – Existência de marcas nos outros lugares depois que a condição fim foi completada.
Uma vez que a marca aparece no lugar fim, todas as outras marcas devem ter desaparecido.
- 2 – Processos sem condição de entrada e saída. Quando um processo não tem nenhuma condição de entrada, não fica claro quando poderá ser realizado. Quando um

processo não tem condições de saída, não contribui para o sucesso do caso e pode ser omitido.

3 – Processos Inativos (*deadtasks*). O processo nunca poderá ser realizado.

4 – *Deadlock*. Trava o caso antes que seja completado.

5 – *Livelock*. Entra em *loop* infinito, não existindo nenhuma oportunidade de escapar.

6 – Processos que continuam acontecendo mesmo quando a condição fim é atingida.

Uma boa definição de processo tem um início claro (condição início) e fim (condição fim). Uma vez que a condição fim é atingida, nenhuma tarefa deveria ser executada.

Foram realizados testes da primeira versão do Método de Avaliação do Modelo de Processos de Negócio do EKD. Os modelos testados são de outros trabalhos e por não serem desenvolvidos seguindo a primeira versão do Método de Avaliação apresentaram alguns problemas de modelagem. Os testes realizados permitiram a avaliação dos quatro principais problemas de modelagem de processos de negócio que impossibilita que o modelo seja *sound*. Os principais problemas foram:

1. Não habilitar *inf-sets* que não foram ainda gerados pelo processo.
2. Realimentar o processo com *inf-set* que não está habilitado.
3. Não ter as condições início e fim claras.
4. Ter *inf-sets* habilitados depois que a condição fim é atingida.

Com base nesses problemas, pode-se afirmar que o cuidado no processo de modelagem é fundamental para que o modelo represente fielmente como o processo é realizado e que esses problemas são minimizados quando o modelo é desenvolvido de acordo com o método desenvolvido neste trabalho.

Os testes comprovaram que por meio da aplicação da primeira versão do método de avaliação são identificados muitos dos erros possíveis em uma definição de processo, mesmo sem conhecimento específico do processo de negócio. Conclui-se que muitos erros estão relacionados com o planejamento e ordem de execução de atividades e que deve haver uma atenção maior para os possíveis erros da definição do processo.

Neste trabalho, foi tratado o fluxo de informações e materiais do processo e a correteza lógica do processo definido embora o procedimento de mapeamento

desenvolvido neste trabalho possa ser usado para tratar recursos e tempo. Nesse último usa-se redes de Petri estocásticas. No capítulo 5 foi apresentado um exemplo de redes de Petri estocásticas.

A grande conveniência, como afirmado anteriormente, no uso de redes de Petri na modelagem de processos de negócios é a possibilidade de um rastreamento minucioso e não-ambíguo de cada etapa da operação.

Além disso, este trabalho mostra que as redes de Petri possibilitam uma representação matemática formal e disponibiliza mecanismos de análise que tornam possíveis a verificação da correção do modelo e a checagem de suas propriedades.

Foi destacado no capítulo 5 que a questão crítica de verificação de modelos de processos de negócio é negligenciada. Essa ausência de apoio pode ser explicada pelo fato de que a verificação de processos de negócio é complicada do ponto de vista computacional e algorítmico. As consequências, de qualquer forma, são que poucos modelos de processos de negócio são checados a fundo antes que eles estejam em execução, o que freqüentemente resulta em erros que serão corrigidos com altos custos.

O fato de algumas construções não serem permitidas pode ser considerado uma desvantagem da formalização do MPN-EKD. Porém, durante o processo de modelagem essas construções devem ser analisadas cuidadosamente, sendo importante o discernimento da equipe ou pessoa que está modelando para que o modelo seja desenvolvido de acordo com as definições criadas neste trabalho.

Uma desvantagem da formalização do MPN-EKD observada é que ao incorporar conceitos de *AND-Split*, *AND-Join*, *OR-Split* e *OR-Join*, algumas construções do modelo de processos de negócio não são permitidas. Para resolver esse problema, foi desenvolvida uma estratégia, apresentada no capítulo 6, para tratar essas situações antes de aplicar o método.

É importante ressaltar que a aplicação do método de avaliação em muitos Modelos de Processos de Negócio poderá ser inviável sem uma ferramenta computacional que apóie todos os passos do método.

Resumindo, como resultados finais deste trabalho, pode-se destacar:

- Uma compilação atualizada da literatura de *workflow* e redes de Petri.
- Formalização inédita do Modelo de Processos de Negócio do EKD.

- Criação de uma estratégia para tratar as situações em que as construções não são permitidas de acordo com os conceitos de AND-Split, AND-Join, OR-Split e OR-Join.
- Criação de um procedimento inédito de mapeamento do Modelo de Processos de Negócio do EKD em redes de Petri.
- Criação da primeira versão do Método de Avaliação de Processos de Negócios.
- Realização de testes de aplicação da primeira versão do Método de Avaliação do Modelo de Processos de Negócio.

8.3. Trabalhos Futuros

Espera-se incentivar a continuação de um esforço de pesquisa nesta área com o desenvolvimento deste trabalho. Algumas sugestões de trabalhos futuros são:

- Desenvolver uma ferramenta que funcione como editor de Modelo de Processos de Negócio do EKD formal. Essa ferramenta facilitará o processo de modelagem e possíveis modificações no modelo. É importante que essa ferramenta tenha uma interface simples para que não seja necessário muito treinamento para utilizá-la. Dessa forma, os usuários se sentem mais à vontade para trabalhar e modificar o modelo. É conveniente que essa ferramenta siga as definições criadas neste trabalho do MPN-EKD formal para que não permita que o modelo seja desenvolvido de forma incorreta. É relevante que a ferramenta avise que a construção não é permitida quando o usuário tentar modelar de forma incorreta.
- Desenvolver ferramenta que faça o mapeamento do Modelo de Processos de Negócio do EKD formal em redes de Petri para que o procedimento possa ser utilizado em grandes projetos. Nesse caso, o método pode ser aplicado corretamente sem que o usuário tenha conhecimento em redes de Petri. É interessante que a ferramenta apresente um relatório de simulação do modelo indicando se o modelo é *sound* ou não. Se não for apresentar todos os motivos.
- Realizar mais testes com o procedimento de mapeamento para que a primeira versão do Método de Avaliação do Modelo de Processos de Negócio do EKD seja aprimorada.

- Testar a formalização do Modelo de Processos de Negócio do EKD em outros casos e buscar alternativas para as situações que a formalização impõem limitações. Apresentar uma lista de possíveis soluções e estratégias para os problemas comuns encontrados no processo de modelagem, sempre seguindo as definições do MPN-EKD formal.
- Aprimorar a primeira versão do método de avaliação do Modelo de Processos de Negócio do EKD para que sejam utilizados recursos no Modelo de Processos de Negócio do EKD formal. Com recursos será possível controlar a disponibilidade de pessoas e máquinas na execução dos processos. Recursos também são fontes de *deadlocks* (travamentos) nos processos. Com esse aprimoramento será possível avaliar os principais problemas na execução das tarefas na ausência de um ou mais funcionários ou na falha de uma ou mais máquinas. É importante que o método aprimorado para utilizar recursos seja testado em outros casos.
- Desenvolver ferramenta que faça o mapeamento do Modelo de Processos de Negócio aprimorado para tratar recursos. Aplicar o método aprimorado para recursos. É conveniente que recursos de simulação seja fornecido para que seja possível avaliar o processo de negócio com diferentes números de funcionários e máquinas.
- Aprimorar o método para tratar tempo e custo dos processos de negócio. No fim do processo o tempo total será fornecido pela soma de todos os fatores de tempo e o custo pela soma dos fatores de custo. Para tratar tempo e custos, redes de Petri estocásticas deverão ser utilizadas. É importante que o método aprimorado seja validado e testado em outros casos.
- Desenvolver ferramenta que faça o mapeamento do Modelo de Processos de Negócio com adição de tempo e custos. Aplicar o método aprimorado para tempo e custos. Fornecer recursos de simulação para que seja possível avaliar o processo de negócio executado de formas diferentes buscando a melhor relação custo e tempo.
- Integrar todas essas ferramentas sugeridas anteriormente em uma única ferramenta. O usuário fará a opção entre aplicar o método de avaliação do Modelo de Processos de Negócio do EKD em um modelo simples, com recursos ou tratando tempo e custos.

Finalmente, espera-se, que o desenvolvimento dos temas acima apresentados possa tornar esse procedimento de mapeamento do Modelo de Processos de Negócio do EKD em redes de Petri mais aprimorado, obtendo-se a melhor eficácia e eficiência possível na sua aplicação e revelando de forma mais clara ainda suas vantagens e limitações.

Referências Bibliográficas

- AALST, W.M.P.V.D. (1995). **A Class of petri net for modeling and analyzing business processes**. Computing Science Reports 95/26, Eindhoven University of Technology, Eindhoven, The Netherlands.
- _____. (1999). Formalization and verification of event-driven process chains. *Information and Software Technology*, London, v.41, n.10, p.639-650, July.
- _____. (1996). Business process redesign: a petri-net-based approach. **Computers In Industry**, Amsterdam, v.29, n.1-2, p.15-26.
- _____. (1996). Three good reasons for using a petri-net based workflow management system. In: INTERNATIONAL WORKING CONFERENCE ON INTEGRATION IN ENTERPRISES, 1996, Cambridge, Massachusetts. **Proceedings...** Cambridge. p.179-201.
- _____. (1997). Verification of workflow nets. In: AZEMA, P.; BALBO, G. (Ed.). **Application and theory of petri nets**. Berlin: Springer-Verlag. (Lectures Notes in Computer Science).
- _____. (1998). The Application of petri nets to workflow management. **The Journal of Circuits, Systems and Computers**, Singapore, v 8, n. 1, p.21-68.
- _____. (1999). Woflan: a petri-net-based workflow analyser. **System Analysis – Modeling–Simulação**, v.35, n.3, p.345-357.
- _____. (2000). Workflow Verification: Finding control-flow errors petri-net-based techniques. In: AALST, V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management: model, techniques, and empirical studies**. Berlin: Springer-Verlag. p. 161-183. (Lecture Notes in Computer Science, 1806).
- AALST, W.M.P.V. et al. (2000). Advanced workflow patterns. In: ETZION, O.; SCHEUERMANN, V. International conference on cooperative information systems. 7th ed. Berlin: Springer-Verlag. p.198-209. (Lectures Notes in Computer Science, 1901).
- AALST, W.M.P.V.D.; HEE, K.M.V.D. (1995). Framework for business process redesign. In: WORKSHOP ON ENABLING TECHNOLOGIES: infrastructure for collaborative enterprises, 4., 1995, Berkeley. **Proceedings...** New York: IEEE Computer Society. P.36-45.
- AALST, W.M.P.V.D; HOFSTEDE, A.H.M.T. (2000). Verification of workflow task structures a petri-net-based approach. **Information Systems**, Oxford, v.25, n.1, p.43-69.
- AALST, W.V.D.; HEE, V.K. (2002). **Workflow management: models, methods and systems**. Cambridge: MIT Press.
- ACHOUR, B.; SOUVEYET, C; TAWBI, M. (1999). Bridging the gap between users and requirements engineering: the scenario-based approach. **International Journal of Computer Systems Science & Engineering**, v.14, n.6, p. 112-122, Nov.
- ADAM, N.R.; ATLURI, V; HUANG, W.K. (1998). Modeling and analysis of workflows using petri nets. **Journal of Intelligent Information Systems**, Norwell, v.10, n.2, p.1-29. Special Issue on Workflow and Process Management.

ALENCAR, F.M.R. (1999). **Mapeando a modelagem organizacional em especificações precisas**. 304 p. Tese (Doutorado) - Centro de Informática, Universidade Federal de Pernambuco, Recife, 1999. 21/12/1999.

ATLURI, V.; HUANG, W.K. (1996). **Na Authorization model for workflows**. Berlin: Springer-Verlag. (Lectures Notes in Computer Science, 1146). p.44-64. 1996.

BADOUEL, E.; OLIVER, J. (1998). **Reconfigurable nets, a class of high level petri nets supporting dynamic changes within workflow systems**. Publication Interne IRISA PI 1163.

BASSIL, S. et al. (2002). Addressing dynamism in e-negotiations by workflow management systems. In: INTERNATIONAL WORKSHOP ON DATABASE AND EXPERT SYSTEMS APPLICATIONS, 13./ INTERNATIONAL WORKSHOP ON NEGOTIATIONS IN E-MARKETS-BEYOND PRICE DISCOVERY, 3., 2002, Aix-em-Provence. **Proceedings...** New York: IEEE. p.655-659.

BERIO, G.; PETIT, M. (2003). Enterprise modelling and UML: (sometimes) a conflict without a case. In: ISPE INTERNATIONAL CONFERENCE ON CONCURRENT ENGINEERING : research and applications, 10., 2003, Madeira Island. *Proceedings...*

BERNUS, P.; NEMES, L. (1997). Requirements of the generic enterprise reference architecture and methodology. **Annual Reviews in Control**, v.21, p.125-136

BERZTISS, A.T.; BUBENKO JR., J.A. (1995). A Software process model for business reengineering. In: INFORMATION SYSTEMS DEVELOPMENT FOR DECENTRALIZED ORGANIZATIONS / IFIP WORKING CONFERENCE, 1995, Trondheim. **Proceedings...** Norway: Chapman & Hall.

BRYMAN, A. (1989). **Research methods and organization studies**. Routledge: [s.n.].

BUBENKO JR., J.A. (1993). Extending the scope of information modelling. In: INTERNATIONAL WORKSHOP ON THE DEDUCTIVE APPROACH TO INFORMATION SYSTEMS AND DATABASE, 4., 1993, Lloret. **Proceedings...** Barcelona: Departament de Llenguatges i Sistemes Informatics of the Universitat Politecnica de Catalunya. p.73-98.

BUBENKO JR., J.A.; KIRIKOVA, M. (1994). Enterprise modelling: improving the quality of requirements specification. In: IRIS-17 INFORMATION SYSTEMS RESEARCH SEMINAR IN SCANDINAVIA, 1994, Syote. **Proceedings...**

BUBENKO JR., J.A.; STIRNA, J.; BRASH, D. (1998). **EKD user guide, Dpt of computer and systems sciences**. Stockholm: Royal Institute of Technology.

BUBENKO JR., J.A.; WANGLER, B. (1993). Objectives driven capture of business rules and information systems requirements. In: INTERNATIONAL CONFERENCE ON SYSTEMS, MAN AND CYBERNETICS, 1993, Le Touquet. **Proceedings...** [S.l.:s.n.]. p.670-677.

BUCKHOUT, S. et al. (1999). Por um ERP eficaz. *HSM Management*, São Paulo, ano 3, n.16, p.30-36, set./out.

CASATI, F. et al. (1995). Conceptual modeling of workflows. In: INTERNATIONAL OBJECT-ORIENTED AND ENTILY-RELATIONSHIP MODELLING CONFERENCE, 14., 1995, Brisbane. **Proceedings...** Berlin: Springer-Verlag. p. 438-455.

- CERVO, A.; BERVIAN, P.A. (1983). **Método científico**. São Paulo: MacGraw Hill.
- CHEN-BURGER, Y. (2001). Knowledge sharing and inconsistency checking on multiple enterprise models. In: IJCAR INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE, 2001, Sienna. *Automated reasoning: proceedings...* Berlin: Springer. (Lecture Notes in Computer Science, 2083).
- CHEN-BURGER, Y.; STADER, J. (2003). Formal support for adaptive workflow systems in a distributed environment. In: FISCHER, I.; LAYNA, M. (Ed.). *Workflow handbook 2003*. Lighthouse Point, FL, USA: Future Strategies. (Workflow Management Coalition).
- CHEN-BURGER, Y.; TATE, A.; ROBERTSON, D. (2002). Enterprise modelling: a declarative approach for FBPM. In: EUROPEAN CONFERENCE OF ARTIFICIAL INTELLIGENCE, KNOWLEDGE MANAGEMENT AND ORGANISATIONAL MEMORIES WORKSHOP, Lyon. *Proceedings...*
- CHIZZOTTI, A. (1995). **Pesquisa em ciências humanas**. 2.ed. São Paulo: Cortez.
- CHYZASTOWSKI-WACHTEL, P. et al. (2003). A top-down petri net-based approach for dynamic workflow modeling. In: INTERNATIONAL CONFERENCE BPM 2003, 2003, Eindhoven. *Proceedings...* Berlin: Springer. p.336-353. (Lecture Notes in Computer Science, 2678).
- CINDIO, F. et al. (1988). **CHAOS**: a knowledge-based system for conversing within offices, office knowledge representation, management and utilization. Lammersdorf: W.Elsevier Science; B.V.North-Holland. p.257-275.
- D'ONOFRIO, S. (1999). **Metodologia do trabalho intelectual**. São Paulo: Atlas.
- DALLAVALLE, S.I. CAZARINI, E.W. (2001a). Modelagem Organizacional Desenvolvimento do Conhecimento Organizacional, Facilitador de Desenvolvimento de Sistemas de Informação. In: Encontro Nacional de Engenharia de Produção, 11, 2001. Salvador, Anais... Salvador: [s.n.]. CD-ROM.
- DALLAVALLE, S.I. CAZARINI, E.W. (2001b). Modelagem Organizacional para capturar os requisitos organizacionais. [online]. In: Simpósio de Engenharia de Produção, 8, Bauru, UNESP, 2001. Anais. Available from WWW: URL: <http://www.bauru.unesp.br/acontece/simpep.html> [30 Nov. 2001].
- DALLAVALLE, S.I. CAZARINI, E.W. (2001c). Desenvolvimento do Conhecimento Organizacional como Base para o Desenvolvimento de Sistemas de Informação. [online]. In: Simpósio de Engenharia de Produção, 8, Bauru, UNESP, 2001. Anais. Available from WWW: URL: <http://www.bauru.unesp.br/acontece/simpep.html> [30 Nov. 2001].
- DALLAVALLE, S.I.; CAZARINI E.W. (2000). Regras do negócio: fator chave de sucesso no processo de desenvolvimento de sistemas de informação. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 10., 2000, São Paulo. **Anais...** São Paulo: [s.n.]. CD-ROM.
- DELLAROCAS, C.; KLEINS, M. (2000). A knowledge-based approach for designing robust business processes. In: VAN DER AALST, W.; DESEL, J.; OBERWEIS, A. (Ed.). *Business process management: models, techniques, and empirical studies*. Berlin: Springer. p.50-65. (Lecture Notes in Computer Sciences, 1806).
- DEMO, P. (1987). **Introdução à metodologia da ciência**. São Paulo: Atlas.

DELEN, D.; BENJAMIN, P.C. (2003). Towards a truly integrated enterprise modeling and analysis environment. *Computers in Industry*, Amsterdam, v.51, n.3, p.257-268, Aug.

DERHNERT, J. (2003). Four steps towards sound business process models. In: EHRIG., H. et al. (Ed.). *Petri net technology for communication-based systems-advances in petri nets*. Berlin: Springer. p.66-82. (Lecture Notes in Computer Science, 2472).

DESEL, J.; ERWIN, T. (2000). Modeling simulation and analysis of business process. In: AALST,V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management: models, techniques, and empirical studies**. Berlin: Springer. p. 129-141. (Lectures Notes in Computer Sciences, 1806).

DESEL, J.; ESPARZA, J. (1995). **Free choice petri nets Cambridge tracts in theoretical computer science**. Cambridge: Cambridge University Press. v.40.

DESROCHERS, A.A.; AL-JAAR, R.Y. (1994). **Applications of petri nets in manufacturing systems: modeling, control and performance analysis**. New Jersey: IEEE Press.

DI FEBBRARO, A. et al. (2002). Optimization of manufacturing systems modelled by timed petri nets. In: INTERNATIONAL WORKSHOP ON DISCRETE EVENT SYSTEMS, 6., 2002, Zaragoza. **Anais eletrônicos...** Disponível em: <<http://computer.org/proceeding/wodes/1683/1683toc.htm>>. Disponível em: 10/08/2004.

DOBSON, J.E.; STREMS, R. (1994). Organizational requirements definition for information technology systems. In: IEEE INTERNATIONAL CONFERENCE ON REQUERIMENTS ENGINEERING, 1994, Los Alamitos. **Proceedings...** Los Alamitos: IEEE Computer Society. p.158-165.

ELECTRICAL enterprise knowledge for transforming applications.(2000). The ELEKTRA project programme. Disponível em: <www.singular.gr/elektra.ekd.htm>. Acesso em: 27 Nov. 2000.

ELLIS, C.A.; NUTT, G.J. (1996). Workflow: the process spectrum. In: NSF WORKSHOP ON WORKFLOW AND PROCESS AUTOMATION IN INFORMATION SYSTEM: state-of-the-art and future directions, 1996, Athens. **Proceedings...** Arlington: NSP. p.140-145.

ESHUIS, R.; DEHNERT, J. (2003). Reactive petri nets for workflow modeling. In: INTERNATIONAL CONFERENCE ON THE APPLICATIONS AND THEORY OF PETRI NETS, 24. Proceedings... Berlin: Springer. p.296-315. (Lecture Notes in Computer Science, 2679).

FENG, C.X.; KUSIAK, A.; HUANG, C.C. (1996). Cost evaluation in design with form features. **Computer-Aided Design**, London, v.28, n.11, p.879-885, Nov.

FERRAZ, P.; INAMASU, R.Y.; PORTO, A.J.V. (2001). Modelo de implementação de um template para Rede de Petri Virtual aplicada a sistemas de manufatura automatizados. In: CONGRESSO NACIONAL DE ENGENHARIA MECÂNICA, 16., 2001, Uberlândia. **Anais...** Rio de Janeiro: ABCM. v.1, p.567-574.

FERSCHA, A. (1994). Qualitative and quantitative analysis of business workflows using generalized stochastic petri nets. In: WORKFLOW MANAGEMENT -

challenges, paradigms and products, 1994, Linz. **Proceedings...** Oldenbourg: Verlag. p.222-234.

FEYERABEND, P. (1989). **Contra o método**. Rio de Janeiro: Francisco Alves.

FURLAN, J.D. (1997). **Modelagem de negócio** - uma abordagem integrada de modelagem estratégica funcional, de dados e a orientação a objetos. São Paulo: Makron Books.

GANG, X.; ZHIMING, W. (2002). A Kind of deadlock-free scheduling method based on petri net. In: IEEE INTERNATIONAL SYMPOSIUM ON HIGH ASSURANCE SYSTEMS ENGINEERING, 7., 2002, Tokyo. **Proceedings...** Disponível em:<<http://computer.org/proceedings/hase/1769/1769toc.htm>>. Disponível em: 05/04/2004.

GIL, A.C. (1991). **Como elaborar projetos de pesquisa**. São Paulo: Atlas.

_____. (1999). **Métodos e técnicas da pesquisa social**. São Paulo: Atlas.

GOTTESDIENER, E. (1999). Capturing business rules. **Software Development**, San Francisco, v.7, n.12, p.72-76, Dec.

GRIGOROVA, K. (2004). Process modelling using petri nets. In: INTERNATIONAL CONFERENCE ON COMPUTER SYSTEMS AND TECHNOLOGIES: e-learning, 4., 2003, Rousse. p.95-100. Disponível em:<<http://doi.acm.org/10.1145/973620.973636>>. Acesso em: 14 June.

GROVER, V. et al. (1998). The influence of information technology diffusion and business process change on perceived productivity: the IS executive's perspective. *Information and Management*, Amsterdam, v.34, n.3, p.141-159, Oct.

GUTH, V.; LENZ, K.; OBERWEIS, A. (1998). Distributed workflow execution based on fragmentation of Petri Nets. In: IFIP WORD COMPUTER CONGRESS TELECOOPERATION, THE GLOBAL OFFICE, TELEWORKING AND COMMUNICATION TOOLS, 15., 1998, Budapest. **Proceedings...** Vienna: Ostareichische Computer Gesellschaft.

HAN, Y. et al. (1994). Management of workflow resource to support runtime adaptability and system evolution. In: PRACTICAL ASPECT OF KNOWLEDGE MANAGEMENT INTERNATIONAL CONFERENCE, 1994, **Proceedings...** p.30-31.

HEE, K.M.; REIJERS, H.A. (2000). Using formal analysis techniques in business process redesign. In: VAN DER AALST, W.; DESEL, J.; OBERWEIS, A. (Ed.). *Business process management: models, techniques, and empirical studies*. Berlin: Springer. p.142-160. (Lectures Notes in Computer Sciences, 1806).

HEE, K.M.V.; REIJERS, H.M. (2000). Using formal analysis techniques in business process redesign. In: AALST, V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management: models, techniques, and empirical studies**. Berlin: Springer. p.142-160. (Lectures Notes in Computer Sciences, 1806).

HERBST, H. (1996). Business rules in system analysis: a meta-model and repository system. **Information Systems**, Oxford, v.21, n.2, p.147-166.

HOLT, A.P. (2000). Management-oriented models of business process. In: AALST, V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management:**

models, techniques, and empirical studies. Berlin: Springer. p. 99-109. (Lectures Notes in Computer Sciences, 1806).

INAMASU, R.Y. (1995). **Modelo de FMS: uma plataforma para simulação e planejamento**. 134 p. Tese (Doutorado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 1995.

JACOBSON, I.; BOOCH, G.; RUMBAUGH, J. (1999). *The unified software development process*. Reading: Addison-Wesley.

JANSSENS, G.K.; VERELST, J.; WEYN, B. (2000). Techniques for modeling workflows their support of reuse. In: AALST,V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management: models, techniques, and empirical studies**. Berlin: Springer. p. 1-15. (Lectures Notes in Computer Sciences, 1806).

JONKERS, H. et al. (2003a). Towards a language for coherent enterprise architecture descriptions. In: IEEE INTERNATIONAL ENTERPRISE DISTRIBUTED OBJECT COMPUTING CONFERENCE, 7., 2003, Brisbane. *Proceedings...* Los Alamitos: IEEE Computer Society.

_____. (2003). Concepts for modelling enterprise architectures. In: LANDELIJK ARCHITECTUR CONGRESS, 2003. Disponível em:<http://www.serc.nl/lac/2003/ontvangen_papers.htm>. Acesso em: 16 Apr.

JUNGINGER, S. et al. (2004). Building complex workflow applications: how to overcome the limitations for the waterfall model. In: FISCHER, L. (Ed.). *Workflow Management Coalition: the workflow handbook 2001*. Disponível em:<http://www.boc-eu.com/english/papers/Complex_Workflow_Appl.pdf>. Acesso em: 16 Apr.

KILOV, H.; SIMMONDS, I. (1997). Business rules: from business specification to design. In: BOSCH, J.; MITCHELL, S. **Object oriented technology**. Berlin: Springer. p. 188-194. (Lectures Notes in Computer Sciences, 1357).

KINDLER, E.; MARTENS, A.; REISIG, W. (2000). Inter-operability of workflow applications: local criteria for global soundness. In: AALST,V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management: models, techniques, and empirical studies**. Berlin: Springer. p.235-253. (Lectures Notes in Computer Sciences, 1806).

KIRIKOVA, M. (2000). Explanatory capability of enterprise models. **Data & Knowledge Engineering**, Amsterdam, v.33, n.2, p.119-136, May.

KIRITSIS, D.; NEUENDORF, K.; XIROUCHAKIS, P. (1999). Petri net techniques for process planning and cost estimation. **Advances in Engineering Software**, Oxford, v.30, p.375-387, June.

KIRITSIS, D.; PORCHET, M. (1996). A Generic petri net model for dynamic process planning and sequence optimization. **Advances in Engineering Software**, Oxford, v.25, p.61-71, Jan.

KNOLMAYER, G. et al. (2000). Modeling processes and workflows by business rules. In: VAN DER AALST, W.; DESEL, J.; OBERWEIS, A. (Ed.). *Business process management: models, techniques, and empirical studies*. Berlin: Springer. p.16-29. (Lectures Notes in Computer Sciences, 1806).

KOUBARAKIS, M.; PLEXOUSAKIS, D. (2002). A formal framework for business process modelling and design. *Information Systems*, Oxford, v.27, n.5, p.299-319, July.

- KOZLENKOV, A.; ZISMAN, A. (2002). Are their design specification consistent with our requirements?. In: IEEE JOINT INTERNATIONAL REQUIREMENTS ENGINEERING CONFERENCE, 2002, Germany. *Proceedings...* Germany: University of Essen. p.145-154.
- KRUCHTEN, P. (2000). *The rational unified process*. 2nded. Harlow: Addison-Wesley.
- LAKATOS, E.M.; MARCONI, M.A. (1995). **Fundamentos de metodologia científica**. São Paulo: Atlas
- _____. (1993). **Fundamentos de pesquisa científica**. São Paulo: Atlas.
- LAUDON, K.C.; LAUDON, J.P. (2003). **Sistemas de Informação (com Internet)**. Pearson Brasil. 5^a. ed.
- LEITE, J.C.S.P.; LEONARDI, M.C. (1998). Business rules as organizational policies. In: INTERNATIONAL WORKSHOP ON SOFTWARE SPECIFICATION & DESIGN, 9., 1998, Japan. **Proceedings...** Los Almitos, IEEE CSP. p.68-76.
- LEYMANN, F.; ROLLER, D. (1997). Workflow-based applications. **IBM Systems Journal**, v.36, n.1. Disponível em:<<http://researchweb.watson.ibm.com/journal/sj/361/leymann.html>>. Acesso em: 01 Oct. 2002.
- LI, J. et al. (1994). MAS: a declarative formalism for hierarchical representation of procedural knowledge. **IEEE Transaction on Knowledge and Data Engineering**, New York, v.6, n.4, p.639-643.
- LIEBERS, A.; KALS, H.J.J. (1997). Cost decision support in product design. **Annals of the CIRP**, Paris, v. 46, n.1, p.107-112.
- LIMA, P. et al. (1998). Petri nets for modeling and coordination of robotic tasks. In: IEEE INTERNATIONAL CONFERENCE ON SYSTEMS, MAN AND CYBERNETICS, 1998, San Diego, California, USA. **Proceedings...** New York: IEEE. v.1, p.190-195.
- LOWE, M.; WIKARSKI, D.; HAN, Y. (1995). **Higher-order objects nets and their application to workflow modeling**. Berlin: Technische Universitat Berlin, Forschungsberichte der FB Informatik. p. 34-95.
- MACIÀ, H. et al (2003). A New Synchronization in Finite Stochastic Petri Box Calculus. In: INTERNATIONAL CONFERENCE ON APPLICATION OF CONCURRENCY TO SYSTEM DESIGN, 3., 2003, Guimarães. **Proceedings...** Disponível em:< <http://computer.org/proceedings/acsd/1887/1887toc.htm>>. Acesso em: 10/05/2004.
- McDERMID, D.C. (2003). Integrated business process management: using state-based business rules to communicate between disparate stakeholders. In: INTERNATIONAL CONFERENCE BPM 2003, 2003, Eindhoven. *Proceedings...* Berlin: Springer. p.58-71. (Lecture Notes in Computer Science, 2678).
- MERZ, M. et al. (1995). Workflow modeling and execution with coloured petri net in COSM. In: WORKSHOP ON APPLICATIONS OF PETRI NET TO PROTOCOLS WITHIN THE INTERNATIONAL CONFERENCE ON APPLICATION AND THEORY OF PETRI NET, 16., 1995, Turin. **Proceedings...** Berlin: Springer. p.1-12.
- MOLD, D.; VALK, R. (2000). Object oriented petri net in business process modeling. In: AALST, V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management:**

models, techniques, and empirical studies. Berlin: Springer. p.254-273. (Lectures Notes in Computer Sciences, 1806).

MURATA, T. (1989). Petri net: properties, analysis and applications. **Proceedings of the IEEE**, v.77, n.4, p.541-579.

NEIGER, D.; CHURILOV, L. (2003). Structuring business objectives: a business process modeling perspective. In: INTERNATIONAL CONFERENCE BPM 2003, 2003, Eindhoven. *Proceedings...* Berlin: Springer. p.72-87. (Lecture Notes in Computer Science, 2678).

NICOLAO, M. (1998). **Modelagem de workflow utilizando um modelo de dados temporal orientado a objetos com papéis**. 99 p. Dissertação (Mestrado) - Universidade Federal do Rio Grande do Sul, Porto Alegre.

NURCAN, A.; ROLLAND, C. (2003). A multi-method for defining the organizational change. *Journal of Information and Software Technology*, London, v.45, n.2, p.61-82, Feb.

NURCAN, S. (1998). Analysis and design of co-operative work process a framework. **Information and Software Technology**, London, v.40, n.3, p.143-156, June.

NURCAN, S.; BARRIOS, J. (2003). Enterprise knowledge and information system modelling in na evolving enviroment. In INTERNATIONAL WORKSHOP ON ENGINEERING METHODS TO SUPPRTED INFORMATION SYSTEMS EVOLUTION IN CONJUNCTION WITH, 2003, Geneva. *Proceedings...*

NUTT, G..J. (1996). The Evolution towards. **Flexible workflow systems, Distributed Systems Engineering**, v.3, n.4 , p.276-294.

OBERWEIS, A. (1996). An Integrated approach for the specification of processes and related complex structured objects in business applications. **Decision Support Systems**, Amsterdam, v.17, n.1, p.31-53, Apr.

OBERWEIS, A. et al. (1997). INCOME/WF-A petri-net based aproach to workflow Management. In: KRALLMANN, H. (Ed.). **Wirtschaftsinformatik'97**. Berlin : Springer-Verlag. p.557-580.

PÁDUA, E.M.M. (1996). **Metodologia de pesquisa: abordagem teórico-prática**. Campinas: Papirus.

PÁDUA, S. I. D.; CAZARINI, E.W.; INAMASU, R. Y. (2004). Modelagem Organizacional: captura dos requisitos organizacionais no desenvolvimento de sistemas de informação.. **Revista Gestão e Produção**. v.11, n.2, p.1-20. mai-ago.

PÁDUA, S. I. D.; SILVA, A. R. Y.; INAMASU, R. Y.; PORTO, A. J. V. (2003b). Aplicações e Potencial das Redes de Petri na Engenharia de Produção. In: Simpósio de Engenharia de Produção, 10, Bauru, UNESP, 2003. **Anais**. Avaliable from WWW: URL: <http://www.bauru.unesp.br/acontece/simpep.html> [30 Nov. 2003].

PÁDUA, S. I. D.; SILVA, A. R. Y.; INAMASU, R. Y.; PORTO, A. J. V. (2004). O potencial das redes de Petri em modelagem e análise de processos de negócios. **Revista Gestão e Produção**. v.11, n.1, p.1-11 abril

PÁDUA, S.I.D, BISPO, C.A.F. (2003). Sistema de Gerenciamento de Workflow: um overview e um estudo de caso. In: Encontro Nacional de Engenharia de Produção, 13, 2003. Ouro Preto, **Anais...** Ouro Preto: [s.n.]. CD-ROM.

- PÁDUA, S.I.D, SILVA, A, R., Y., INAMASU, R.Y., PORTO, A, J. V. (2002). Redes de petri aplicadas aos sistemas de gerenciamento de Workflow (CD-ROM). In: Encontro Nacional de Engenharia de Produção, 12, 2002. Curitiba, **Anais...** Curitiba: [s.n.]. CD-ROM.
- PÁDUA, S.I.D. (2001). **Investigação do processo de desenvolvimento de software a partir da modelagem organizacional, enfatizando regras do negócio.** 145 p. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 27/03/2001.
- PÁDUA, S.I.D; PÁDUA, F. S.M.; CAZARINI, E. W.; INAMASU, R, Y. (2003a). Modelo de Atores e Recursos, seu potencial e importância no modelo organizacional. In: Simpósio de Engenharia de Produção, 10, Bauru, UNESP, 2003. **Anais.** Available from WWW: URL: <http://www.bauru.unesp.br/acontece/simpep.html> [30 Nov. 2003].
- PERSSON, A. (2000). The utility of participative enterprise modelling in IS development - challenges and research issues. In: INTERNATIONAL WORKSHOP ON THE REQUIREMENTS ENGINEERING PROCESS, 2., 2000, Greenwich. *Proceedings...* Berlin: Springer. p.978-982.
- PETERSON, J.L. (1981). **Petri nets; theory and modelling of systems.** Englewood Cliffs: Prentice-Hall.
- PETRI net tools database. (2003). Disponível em:<<http://www.daimi.au.dk/PetriNets/tools/db.html>>. Acesso em: 24 Abr. 2003.
- PINOTTI, M.M. (2004). **Diretrizes para guiar o processo de Engenharia de Requisitos a partir de modelos Organizacionais desenvolvidos em EKD.** Dissertação (mestrado) - Instituto de Ciências Matemáticas e Computação, Universidade de São Paulo, São Carlos, Brasil, 97p
- RINDERLE, S.; REICHERT, M.; DADAM, P. (2003). Evaluation of correctness criteria for dynamic workflow changes. In: INTERNATIONAL CONFERENCE BPM 2003, 2003, Eindhoven. *Proceedings...* Berlin: Springer. p.41-57. (Lecture Notes in Computer Science, 2678).
- ROLLAND, C.; GROSZ, G.; KLA, R. (1999). Experience with goal-scenario coupling in requirements engineering. In: IEEE INTERNATIONAL SYMPOSIUM ON REQUIREMENTS ENGINEERING, 4., 1999, Limerich. **Proceedings...** Limerich: University of Limerich. p.74-81.
- ROLLAND, C.; NURCAN, S.; GROSZ, G. (2000). A Decision making pattern for guiding the enterprise knowledge development process. **Journal of Information and Software Technology**, v.42, n. 5, p.313-331, April.
- ROSCA, D. et al. (1997). A Decision making methodology in support of business rules lifecycle. In: IEEE INTERNATIONAL SYMPOSIUM ON REQUIREMENTS ENGINEERING, 1997, Annapolis. **Proceedings...** Los Alamitos: IEEE Computer Society. p.236 -246.
- SALIMIFARD, S.; WRIGHT M. (2001). Petri net based modelling of workflow systems: an overview. **European Journal of Operational Research**, Amsterdam, v.134, n.3, p.664-676, Nov.
- SALOMON, D.V. (1991). **Como fazer uma monografia.** São Paulo: Martins Fontes.

- SANTANDER, V.F.A. (2002). *Integrando modelagem organizacional com modelagem funcional*. 21p. Tese (Doutorado) - Universidade Federal de Pernambuco, Recife, 16/12/2002.
- SCHÖMIG, A.K.; RAU, H. (1995). **A Petri net approach for the performance analysis of business processes**. Würzburg: University of Würzburg. Report n. 116 Seminar at IBFI, Schloss Dagstuhl, May 22-26.
- SILVA, A.R.Y. (2002). **Modelagem de custos em sistemas de manufatura utilizando redes de Petri**. 95 p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.
- SIMON, C. (2001). Developing software controllers with petri nets and a logic of actions. In: IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, 2001, Seoul. **Proceedings...** New York: IEEE. p.58-63.
- SOARES, J.B. (2001). **Editor de modelos de sistemas de eventos discretos, baseado em redes de Petri interpretadas**. Dissertação (mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, Brasil, 56p
- SOMMERVILLE, I. (2001). Software Engineering. Addison-Wesley. 6a. ed.
- STERGIOU, M.; JOHNSON, L. (1998). The Importance of business rules in the organizational transformation process. In: INTERNATIONAL CONFERENCE ON INFORMATION SYSTEMS, ANALYSIS AND SYNTHESIS, 4., 1998, Orlando. **Proceedings...** Orlando: International Institute of Informatics and Systemics.
- TAYLOR, R.; TOFTS, C. (2004). *Modelling, myth vs reality, map vs territory*. Technical Reports HPL-2003-246 20031211 External. Disponível em: <<http://www.hpl.hp.com/techreports/2003/HPL-2003-246.html>>. Acesso em: 16 Apr.
- THIOLLENT, M. (1995). **Metodologias da pesquisa-ação**. São Paulo: Cortez.
- VERBEEK, H.M.W.; BASTEN, T.; AALST, W.M.P. **Diagnosing workflow using woflan**. Eindhoven: Eindhoven University of Technology. BETA Working Paper Series, WP 48.
- VOORHOEVE, M. (2000). Compositional modeling and verification of workflow process. In: AALST, V.D.W.; DESEL, J.; OBERWEIS, A. **Business process management: models, techniques, and empirical studies**. Berlin: Springer. p.184-200. (Lectures Notes in Computer Sciences, 1806).
- WIKARSKI, D. (1996). **An Introduction to modular process net**. Berkeley: International Computer Science Institute. Technical Report TR-96-019.
- WORKFLOW Management coliation: reference model. (1996). Document Number WPMC-TC00-1003 Document status – Issue, Novembro.
- WORKFLOW Management coliation: terminology and glossary. (1996). Document Number WPMC-TC-1011 Document status – Issue, Fevereiro.
- YIN, R.K. (1989). **Case study research design and methods**. 2.ed. Washington: Sage.
- YU, E. (1993). Modelling organizations for informations systems requirements engineering. In: IEEE INTERNATIONAL SYMPOSIUM ON REQUERIMENTS ENGINEERING, 1993, San Diego. **Proceedings...** Los Alamitos: IEEE Computer

Society. p.34-41.

ZISMAN, M.D. (1977). **Representation, specification and automation of office procedures.** Thesis (PhD.) - University of Pennsylvania, Wharton School of Business.