

UNIVERSIDADE DE SÃO PAULO

ANDERSON VALTRIANI SIQUEIRA

MODELAGEM DE PROCESSOS DE NEGÓCIO

São Paulo
Dezembro / 2006

ANDERSON VALTRIANI SIQUEIRA

MODELAGEM DE PROCESSOS DE NEGÓCIO

Monografia apresentada ao Instituto de Matemática e Estatística da Universidade de São Paulo para a conclusão do curso de graduação em Ciência da Computação.

Área de Concentração: Ciência da Computação

Orientador: Prof. Dr. João Eduardo Ferreira

São Paulo
Dezembro / 2006

LISTA DE ILUSTRAÇÕES

Figura 2.1: Elementos da BPMN (Fonte: [12])	11
Figura 2.2: Diretrizes para o projeto conceitual de workflow (Fonte: [13])	20
Figura 3.1: Necessidades de um Analista de Negócio (Baseado em [13])	24
Figura 3.2: Padrão Seqüência (Fonte: [1])	27
Figura 3.3: Seqüência (BPMN)	27
Figura 3.4: Padrão Divisão Paralela (Fonte: [1])	27
Figura 3.5: Divisão Paralela (BPMN)	28
Figura 3.6: Padrão Sincronização (Fonte: [1])	28
Figura 3.7: Sincronização (BPMN)	29
Figura 3.8: Padrão Escolha Exclusiva (Fonte: [1])	29
Figura 3.9: Escolha Exclusiva (BPMN)	30
Figura 3.10: Padrão Junção Simples (Fonte: [1])	30
Figura 3.11: Junção Simples (BPMN)	31
Figura 3.12: Padrão Escolha Múltipla (Fonte [1])	31
Figura 3.13: Escolha Múltipla (BPMN)	32
Figura 3.14: Padrão Junção Sincronizada (Fonte [1])	32
Figura 3.15: Junção Sincronizada (BPMN)	33
Figura 3.16: Padrão Junção Múltipla (Fonte [1])	34
Figura 3.17: Junção Múltipla (BPMN)	34
Figura 3.18: Padrão Discriminador (Fonte [1])	35
Figura 3.19: Discriminador (BPMN)	35
Figura 3.20: Padrão N-Discriminador (Fonte [1])	36
Figura 3.21: N-Discriminador (BPMN) (Fonte: [15])	36
Figura 3.22: Padrão Ciclo Arbitrário	37
Figura 3.23: Ciclo Arbitrário (BPMN)	38
Figura 3.24: Padrão Terminação Implícita (Fonte [1])	38
Figura 3.25: Terminação Implícita (BPMN)	39
Figura 3.26: MI Sem Sincronização (BPMN)	40
Figura 3.27: MI Com Conhecimento Prévio em Tempo de Projeto (BPMN) (Fonte: [15]) ...	40
Figura 3.28: MI Com Conhecimento Prévio em Tempo de Execução (BPMN) (Fonte: [15])	41
Figura 3.29: MI Sem Conhecimento Prévio (BPMN) (Fonte [15])	42

Figura 3.30: Padrão Escolha Postergada (Fonte [1]).....	43
Figura 3.31: Escolha Postergada (BPMN) (Fonte [15])	43
Figura 3.32: Padrão Roteamento Paralelo Entrelaçado (Fonte [1]).....	44
Figura 3.33: Roteamento Paralelo Entrelaçado (BPMN)	44
Figura 3.34: Padrão Marco (Fonte [1]).....	45
Figura 3.35: Marco (BPMN)	45
Figura 3.36: Atividade Cancelável (BPMN) (Fonte: [15]).....	46
Figura 3.37: Caso Cancelável (BPMN).....	47

LISTA DE ABREVIATURAS E SIGLAS

BPD	Business Process Diagram
BPM	Business Process Management
BPMI	Business Process Management Initiative
BPMN	Business Process Modeling Notation
NPDL	Navigation Plan Definition Language
PN	Plano de Navegação
SQL	Structured Query Language
WfMC	Workflow Management Coalition

SUMÁRIO

1 INTRODUÇÃO	6
1.1 Objetivos	7
1.2 Contribuições	7
1.3 Organização do texto	8
2 FUNDAMENTAÇÃO TEÓRICA	9
2.1 Processo de negócio	9
2.2 Modelagem de processo de negócio utilizando a BPMN	10
2.3 Arquitetura <i>RiverFish</i>	14
2.3.1 DEFINIÇÃO FORMAL DE PLANO DE NAVEGAÇÃO.....	14
2.3.2 PRINCIPAIS CARACTERÍSTICAS E PROPRIEDADES DO <i>RIVERFISH</i>	15
2.3.3 <i>RIVERFISH</i> ESTENDIDO.....	16
2.4 <i>Navigation Plan Definition Language (NPDL)</i>	16
2.4.1 A <i>NAVIGATIONPLAN TOOL</i>	19
2.5 Diretrizes para projeto conceitual de <i>workflows</i>	19
3 RESULTADOS OBTIDOS	23
3.1 Necessidades de um Analista de Negócio	24
3.2 Especificação dos Padrões de Controle de Fluxo em NPDL e BPMN	26
3.2.1 PADRÕES BÁSICOS DE CONTROLE DE FLUXO.....	26
3.2.2 Padrões Avançados de Ramificação e Sincronização	31
3.2.3 PADRÕES ESTRUTURAIS	37
3.2.4 PADRÕES DE MÚLTIPLA INSTÂNCIA.....	39
3.2.5 PADRÕES BASEADOS EM ESTADOS	42
3.2.6 PADRÕES DE CANCELAMENTO	46
4 CONCLUSÃO	48
4.1 Trabalhos Futuros	48
REFERÊNCIAS BIBLIOGRÁFICAS	50

Capítulo 1

INTRODUÇÃO

Modelagem de negócio é uma ferramenta conceitual que utiliza um conjunto de elementos e relações entre estes para expressar a maneira como uma determinada atividade é realizada, especificando dados de entrada, saída e o fluxo de informação que ocorre durante a realização da atividade. Assim, a modelagem é usada não apenas para especificar os passos de negócio, mas também para prover uma maneira uma visão simplificada do negócio.

Na modelagem de negócio temos o mapeamento dos processos existentes. Este mapeamento é de grande importância uma vez que, permitindo a visão ampla e simplificada do negócio, possibilita o entendimento e a melhoria dos serviços internos e externos da empresa.

Dessa maneira, a modelagem também é usada para auxiliar as organizações na compreensão de seu próprio negócio, permitindo a compreensão e a identificação de problemas e possíveis melhorias.

De acordo com [10], a modelagem de processos de negócio intenta criar uma definição simples de um processo de tal maneira que profissionais de diferentes áreas possam ver, compreender e manipular, de acordo com suas competências, o processo através de uma notação adequada.

A BPMN é um conjunto de convenções gráficas para descrever processos de negócio. Esta notação foi especificamente projetada para coordenar seqüências de processos e a troca de mensagens existente entre processos. A intenção de um modelo de processo em BPMN é capturar detalhes suficientes para permitir que ele seja utilizado como fonte de uma descrição executável de um processo.

Existem muitas ferramentas para modelagem de processos de negócio. Muitas características podem ser levadas em consideração ao se escolher uma ferramenta, entre elas, podemos destacar a utilização da BPMN para modelagem do processo e a possibilidade de simular e de gerar estatísticas do processo. Para que seja possível a implementação destas duas últimas, é necessário que exista uma maneira de especificar formalmente os processos de negócio.

Há uma grande variedade de linguagens para a definição e execução de processos de negócio. O ideal seria que estas linguagens pudessem descrever completamente qualquer processo de negócio mas, como pode ser visto em uma pesquisa feita por [14], esta não é a realidade. Dessa forma, deve-se escolher a linguagem que atenda as necessidades existentes. Então, visando estabelecer parâmetros para comparação entre as linguagens de definição de processos de negócio, foram definidos padrões de controle de fluxo em [1]. Assim, podemos classificar as linguagens de acordo com os padrões que ela é capaz de expressar.

Existem duas importantes características que as ferramentas para modelagem devem possuir: uma notação que seja completa o suficiente para permitir a modelagem do processo de negócio assim como a utilização de uma linguagem para a definição do processo que possa expressar corretamente o processo de negócio.

1.1 Objetivos

O principal objetivo deste trabalho é identificar como a notação BPMN pode, graficamente, representar os padrões de controle de fluxo. Fazendo o mapeamento dos padrões de controle de fluxo para esta notação.

1.2 Contribuições

A principal contribuição é a representação dos padrões de controle de fluxo através da notação BPMN.

Uma segunda contribuição deste trabalho é a definição de passos que podem ser seguidos para que algumas necessidades que os analistas de negócio possuem possam ser atendidas.

1.3 Organização do texto

No capítulo 2, são apresentados os fundamentos utilizados para desenvolver e entender este trabalho, tais como: o conceito de processos e gerenciamento de negócios, uma descrição da BPMN e de seus elementos que foram utilizados neste texto, a definição do plano de navegação e da arquitetura *RiverFish*, a descrição da NPDL e de seus operadores e a descrição das diretrizes para modelagem de projeto conceitual de workflows.

No capítulo 3, é apresentada a descrição dos padrões de controle de fluxo, assim como um exemplo, sua representação em NPDL e em BPMN para cada um dos padrões. Além disso, apresentamos quais as principais necessidades de um profissional de negócio, e mostramos passos que podem ser seguidos para que elas sejam facilmente atendidas.

Por fim, o capítulo 4 traz a conclusão do trabalho, sua contribuição e apresenta propostas para trabalhos futuros.

Capítulo 2

FUNDAMENTAÇÃO TEÓRICA

2.1 Processo de negócio

Segundo [5], processo de negócio pode ser visto como a ordenação de atividades de trabalho utilizando tempo e espaço, com um início, um fim e um conjunto claramente definido de entradas e saídas. Assim, podemos entender processo de negócio como um conjunto de passos (ou atividades) que devem ser executados em uma determinada ordem. Estas atividades podem ser executadas sequencialmente ou paralelamente.

O gerenciamento de processos de negócio, ou *Business Process Management* (BPM), é um novo método e uma nova tecnologia para manipulação de processos. Além das características relativas ao fluxo de controle, envolve métodos, técnicas e ferramentas para apoiar o projeto, execução, gerenciamento e análise operacional dos processos de negócio.

Com o BPM, houve a possibilidade da abstração do negócio da lógica de aplicativos, pois, segundo [6], ofereceu uma das primeiras oportunidades reais para a separação de gerenciamento de negócios do gerenciamento de sistemas.

Um sistema de BPM é constituído dos seguintes módulos: *Workflow* (ambiente de execução ou motor de execução); Modelagem de Processos (definição e projeto do processo); Simulação, Monitoramento e de Atividade e Interface de usuário. A modelagem de processos pode ser feita utilizando a *Business Process Management Notation* (BPMN), que será descrita a seguir.

2.2 Modelagem de processo de negócio utilizando a BPMN

BPMN foi desenvolvida pela *Business Process Management Initiative* (BPMI) após mais de 2 anos de pesquisa para fornecer uma notação que fosse facilmente entendida pelos usuários de negócio, desde os analistas de negócios, que começam a modelar os processos, até os desenvolvedores, que os implementam. Esta notação visa permitir a comunicação, entre diferentes audiências, de uma grande variedade de informações. Uma descrição detalhada pode ser encontrada em [12].

A BPMN define o *Business Process Diagram* (BPD) que é um conjunto de elementos gráficos que permitem a criação de um modelo de Processo de Negócio.

Com a BPMN temos uma notação gráfica padronizada que descreve os passos de um processo de negócio em um *workflow*.

Os elementos da BPMN foram organizados em quatro categorias básicas.

1. **Objetos de Fluxo:** São os principais elementos gráficos para definir o comportamento de um processo do negócio. Há três objetos do fluxo:
 - a. **Evento** – É algo que ocorre no decorrer do processo de negócio. Eles afetam o fluxo do processo e podem ter um gatilho, ou um resultado associado. Há três tipos de eventos: Inicial, intermediário e final.
 - b. **Atividade** – A atividade é uma tarefa genérica executada pela organização; ela pode ser atômica ou complexa.
 - c. **Gateway** – *Gateways* são usados como estrutura de controle na seqüência do fluxo.
2. **Objetos de conexão:** Os objetos de conexão conectam eventos, atividades e *gateways*. Há três objetos de conexão utilizados para conectar os objetos do fluxo às outras informações:
 - a. **Fluxo de seqüência** – É utilizado para mostrar a ordem que as atividades serão executadas no fluxo.
 - b. **Fluxos de mensagem** – É utilizado para mostrar o tráfego de mensagens entre as entidades do negócio.
 - c. **Associação** – Uma associação conecta informação adicional aos elementos básicos como, por exemplo, uma descrição.

3. **Raia:** Raias são utilizadas para agrupar os elementos de modelagem, possibilitando a organização das atividades em categorias. Há dois tipos de raias:
 - a. **Piscinas** – Uma piscina é um contêiner que agrupa um conjunto de atividades de uma organização. As piscinas podem ser do tipo caixa preta ou caixa branca. A piscina do tipo caixa preta esconde seus detalhes; a comunicação só pode ocorrer com a borda da piscina enquanto a piscina do tipo caixa branca exibe seus detalhes e permite a comunicação com elementos em seu interior.
 - b. **Pistas** – Para decompor uma organização em unidades específicas divide-se longitudinalmente o interior da piscina por meio de pistas.
4. **Artefato:** Artefatos são utilizados para agregar informações ao modelo permitindo uma maior flexibilidade da notação. Atualmente, existem três tipos de artefatos definidos:
 - a. **Objeto de Dados** – São mecanismos utilizados para mostrar como dados são produzidos, ou requeridos pelas atividades.
 - b. **Grupo** – Grupos são utilizados para documentação e não afetam o fluxo do negócio.
 - c. **Anotação** – É utilizado para oferecer um texto com informações adicionais para quem estiver lendo o diagrama BPMN.

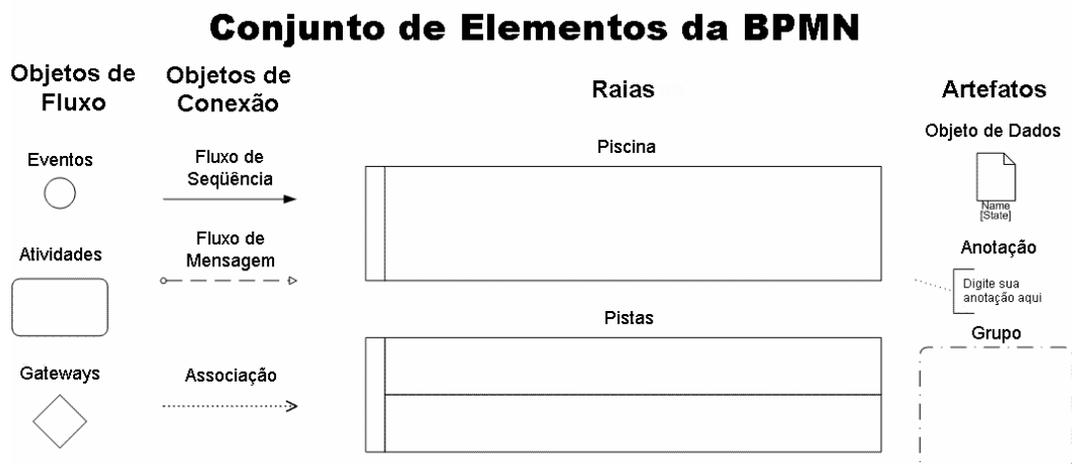


Figura 2.1: Elementos da BPMN (Fonte: [12])

A BPMN possui dezenas de elementos de modelagem. Os elementos citados acima são apenas uma generalização dos elementos existentes. A seguir, serão apresentados os elementos utilizados neste trabalho.

1. **Evento:** Existem 9 tipos de eventos: mensagem, tempo, erro, regra, cancelamento, compensação, ligação, múltiplo e término. A seguir, uma descrição de 3 deles.

- a. **Erro:** É utilizado para o controle de exceção. 
- b. **Ligação:** É utilizado para ligar eventos. Pode ser visto como um “go to”. 
- c. **Término:** Indica que todas as atividades do processo devem ser terminadas. 

2. **Gateway:** Gateways podem ser usados tanto para a união quanto para a divisão do fluxo. Na divisão, temos o gateway como sendo um local onde o fluxo pode seguir dois ou mais caminhos. Na união, o gateway é um local que espera uma ou mais linhas de execução do fluxo antes de habilitar a atividade seguinte que pode, inclusive, ser habilitada mais de uma vez.

- a. **Exclusivo (XOR):** Decisão e união exclusiva.  ou



Divisão: Apenas um caminho será seguido.

União: Aguarda a linha do fluxo que foi ativada antes de habilitar a atividade seguinte.

- b. **Inclusivo (OR):** Decisão e união inclusiva. 

Divisão: Qualquer quantidade de caminho pode ser seguido.

União: Aguarda todas as linhas de fluxos que foram ativadas e só depois habilita a atividade seguinte.

c. **Paralelo (AND):**

Divisão: Todos os caminhos devem ser seguidos.

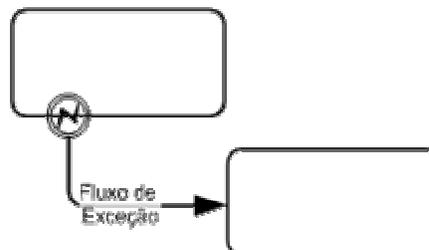
União: Aguarda todas as linhas de fluxo antes de habilitar a atividade seguinte.

d. **Complexo:**

Divisão: Uma expressão determina quais caminhos devem ser seguidos.

União: Uma expressão determina por quais fluxos de seqüência é necessário esperar antes de habilitar a atividade seguinte.

3. **Fluxo de exceção:** Ocorre fora do fluxo normal do processo e é baseado em um evento intermediário que ocorre durante a execução de uma atividade.



4. **Atividade de repetição:** Determina se a atividade é executada zero, uma ou mais vezes. Este tipo de repetição é semelhante às construções “while” e “repeat”.



5. **Atividade de múltiplas instâncias:** Determina a quantidade de vezes que a atividade será executada. Este tipo de repetição é semelhante à construção “for”.



2.3 Arquitetura *RiverFish*

Segundo [7], *RiverFish* é um modelo arquitetônico para a representação, controle e execução de processos de negócio. Esta arquitetura possui uma linguagem de modelagem descrita pelo conceito de *Navigation Plan* (Plano de Navegação). Dessa forma, podemos ver *RiverFish* como uma arquitetura para gerenciamento de processos de negócios.

Um conceito utilizado nesta arquitetura é o conceito de plano de navegação. Este conceito foi definido em [7] e estendido em [8] como um conjunto de todos os processos de negócio exigidos em uma aplicação para se atingirem os objetivos de negócio. O plano de navegação mapeia todas as regras de consistência em função dos dados pertencentes às requisições dos serviços eletrônicos de todos os objetivos de negócio. Um plano de navegação conecta uma requisição ao seu respectivo passo de negócio, e uma instância do PN pode ser entendida como um vínculo entre uma requisição de um determinado usuário e os passos de negócio que responderão por ela.

De acordo com [7], a definição formal de plano de navegação utiliza fundamentos da álgebra de processos. Os fundamentos da Álgebra de Processos não serão descritos neste trabalho, mas podem ser vistos em [2].

2.3.1 DEFINIÇÃO FORMAL DE PLANO DE NAVEGAÇÃO

Definição 1: Ação Simples. Uma ação simples é um conjunto de ações atômicas compostas usando os operadores de seqüência e composição alternativa.

Definição 2: Ponto de verificação. Um ponto de verificação é um conjunto de ações atômicas compostas usando regras restritivas e condicionais.

Definição 3: Passo de Negócio. Um passo de negocio ou é uma ação simples ou um ponto de verificação.

Definição 4: Processo de Negócio. Um processo de negocio é um conjunto de passos de negocio compostos usando operadores básicos da álgebra de processos e suas extensões.

Definição 5: Plano de Navegação. Um plano de navegação é um conjunto de todos os processos de negocio exigidos em uma aplicação para atingir o objetivo de negocio.

Passos de negocio podem ser especializados em ações simples, pontos de verificação, regras e tipo de regras. Cada passo especializado implementa suas ações atômicas internas sobre um dado específico. Quando um plano de navegação é instanciado, um vinculo é criado entre os dados e os passos de negocio.

2.3.2 PRINCIPAIS CARACTERÍSTICAS E PROPRIEDADES DO RIVERFISH

Segundo [7], o modelo arquitetônico *RiverFish* apresenta duas principais características:

1. A reutilização de um determinado passo de negócio na definição de outros processos de negócio:
2. O controle de execução das instancias dos planos de navegação em bancos de dados relacionais.

Para os autores, estas duas características aliadas à definição formal do Plano de Navegação, ratificam as seguintes propriedades da arquitetura *RiverFish*:

1. Reaproveitamento
2. Formalização
3. Instanciação
4. Precisão na localização da interação

Estas quatro propriedades do RF-NP estão fundamentadas nas definições da Álgebra de Processos, relacionadas à representação do formalismo do ciclo de regras. Suas definições podem ser encontradas em [2] e [7].

2.3.3 RIVERFISH ESTENDIDO

Segundo a arquitetura de *RiverFish*, processos podem ser divididos em duas categorias: atômicos e compostos. Processos de negócio atômicos são descritos em termos de passos de negócio, sendo que passos de negócio podem ser ações simples ou pontos de verificação. Processos de negócios compostos são aqueles obtidos a partir de outros processos de negócio. Ações e pontos de verificação são utilizados para manipulação de dados.

Em [9], o metamodelo do *RiverFish* foi estendido, com o objetivo de utilizar ontologias para representar o processo de negócio. Dessa forma, inseriu-se a classificação de regras de negócio, possibilitando a representação do encadeamento seqüencial do fluxo de execução de um processo de negócio. E a arquitetura foi adaptada para representar os conceitos de classificação de regras de negócio propostos em [11].

As regras de negócio foram agrupadas em duas categorias: restritivas e condicionais. Regras restritiva não admitem respostas falsas, ou seja, o fluxo só pode continuar sendo executado se uma regra deste tipo for atendida. Regras condicionais avaliam a respostas e, com base nela, decidem por qual caminho seguir.

2.4 Navigation Plan Definition Language (NPD)

A NPD é uma extensão da linguagem SQL. Ela foi definida por [4] com o objetivo de viabilizar o controle de processos de negócio para um modelo relacional de dados. Esta linguagem é baseada no conceito do plano de navegação da arquitetura *RiverFish* e em operadores da álgebra de processos.

Processos em NPD são definidos por expressões algébricas, que são constituídas por ações atômicas, processos e operadores NPD.

A NPD contém os operadores mais comuns da álgebra de processos, além de também definir operadores adicionais, com o objetivo de atender a todas as categorias de padrões de controle de fluxo.

Os operadores da NPD L são:

1. **Composição Seqüencial** (operador "."): o termo de processo A . B significa que inicialmente somente o termo A está habilitado para execução. O termo B será habilitado para execução somente após o fim da execução do termo A;
2. **Composição Alternativa** (operador "+"): o termo de processo A + B significa que ambos termos A e B estarão habilitados para a execução;
3. **Composição Paralela** (operador "||"): o termo de processo A || B significa que os termos A e B podem ser executados paralelamente; Este operador representa o operador de entrelaçamento da álgebra de processos, com a ressalva de que na NPD L não há o conceito de comunicação da ACP.
4. **Composição Paralela Entrelaçada** (operador "|*"): o termo de processo A |* B significa que os termos A e B podem ser executados em qualquer ordem, mas não paralelamente;
5. **Composição Multi-Convergente** (operador "&"): o termo de processo A & B significa que o termo B será habilitado para execução após o término da execução de cada linha de execução do termo A. Este operador foi criado para facilitar o uso do padrão junção múltipla.
6. **Composição Discriminatória** (operador "^"): o termo de processo A ^ B significa que o termo B será habilitado para execução após o primeiro término de uma linha de execução do termo A; Este operador foi criado para facilitar o uso do padrão discriminador.
7. **Repetição Ilimitada** (operador "?*"): o termo de processo A?* significa que o termo A pode ser executado, de forma paralela, uma ou mais vezes; Este operador foi criado para facilitar a representação dos padrões de múltipla instâncias. Este padrão reproduz o comportamento que define a criação de subprocessos recursivos para a representação de múltiplas instâncias.
8. **Repetição Numericamente Limitada** (operador "?n"): o termo de processo A?5 significa que o termo A deve ser executado, de forma paralela, 5 vezes; Este operador é uma variação do operador ?* que permite indicar o número de vezes que o termo deve ser instanciado e

executado. O número associado ao ? deve ser um número inteiro, positivo e não nulo.

9. **Repetição Limitada por Função** (operador "?f"): o termo de processo $A?f$ significa que o termo A deve ser executado, de forma paralela, o número de vezes retornado pela função f em tempo de execução; Este operador é uma variação do operador $?^*$ que permite a associação de uma função para determinar em tempo de execução quantas instâncias da atividade serão criadas. A função que pode ser associada ao "?" é uma função atômica cujo valor de retorno é um inteiro positivo e não nulo. A função será avaliada em tempo de execução do processo.
10. **Execução Condicional** (operador "%r", sendo r uma regra de negócio, que retorna um valor *booleano*): o termo de processo $\%r1 A$ significa que o termo A será habilitado para execução se o valor de retorno da regra $r1$ for verdadeiro. Sendo assim, uma regra na NPDL pode ser entendida como uma condição para a execução de uma atividade;
11. **Execução Condicional Negativa** (operador "%!r", sendo r uma regra de negócio, que retorna um valor *booleano*): o termo de processo $\%!r1 A$ significa que o termo A será habilitado para execução se o valor de retorno da regra $r1$ for falso.

O comportamento dos operadores de execução condicional (10) e (11) não são representadas em álgebra de processos. Os operadores (4), (5), (6), (7), (8) e (9) também não existem na álgebra de processos, entretanto, seus comportamentos podem ser representados com a combinação de outros operadores de álgebra de processos.

2.4.1 A NAVIGATIONPLAN TOOL

A *NavigationPlan Tool* foi implementada na forma de uma biblioteca de funções e foi desenvolvida para atender às características especificadas pela *Workflow Management Coalition (WfMC)* para um sistema de *workflow*. Ela é composta por três serviços: o **Interpretador NPD**L, o **Serviço de Instanciação de Processos** e o **Serviço Monitor de Execução de Instâncias de Processos**.

A função do Interpretador NPD L é receber um comando de entrada e efetuar as análises léxica, sintática e semântica. O Serviço de Instanciação de Processos disponibiliza funções para a criação de instâncias de processos. O Serviço Monitor de Execução de Instâncias de Processos é responsável por ligar uma instância de processo aos seus dados de execução (plano de navegação). Não entraremos muito em detalhes sobre a *NavigationPlan Tool*, mais informações podem ser obtidas em [4] e em [3].

O nosso interesse na *NavigationPlan Tool* é que, dada uma expressão algébrica em NPD L, podemos criar uma árvore de expressão (árvore de navegação), que é uma árvore binária completa, que representa a expressão algébrica.

Com esta árvore de navegação, é possível determinar a ordem de execução dos passos de um plano de navegação. E assim, tem-se o controle do estado de cada passo.

2.5 Diretrizes para projeto conceitual de *workflows*

A idéia principal destas diretrizes é propor uma seqüência de fases que conduza a ordenação e encadeamento das tarefas que compõe o fluxo de informações de um processo de negócio, possibilitando a construção de um projeto conceitual de *workflow*, que deverá ser um artefato, grafo orientado, apto a ser representado, simulado e executado em ferramentas de BPM.

A concepção destas diretrizes se apóia na arquitetura para gerenciamento de processos de negócio, denominada *RiverFish*, na classificação de regras de negócio para *RiverFish*, no conceito de precedência de atividades, nos mecanismos básicos de seqüência, paralelismo, iteração e seleção para modelagem de fluxo de controle de um processo de negócio e nos padrões do *Workflow*.

Estas diretrizes são um conjunto de etapas de modelagem, denominadas fases, e são descritas a seguir:

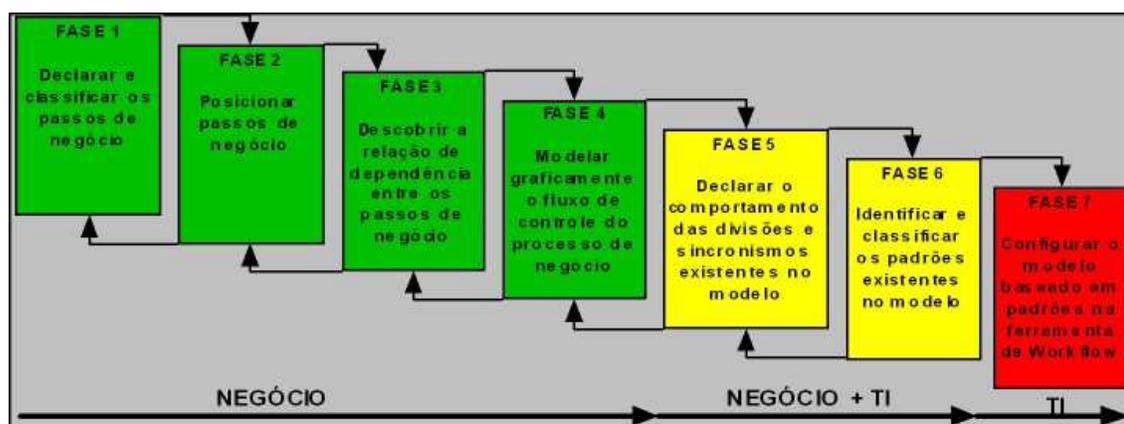


Figura 2.2: Diretrizes para o projeto conceitual de workflow (Fonte: [13])

As principais características das diretrizes são:

1. Delimitação e separação das fases de construção do “Projeto Conceitual de *workflows*” para atuação de especialistas de negócio e TI;
2. Aplicação das fases de forma iterativa e incremental, exceto a sétima fase que é apenas incremental;
3. Recuperação do processo de modelagem por uso de tabelas e diagramas gerados em cada etapa;

A seguir, cada uma das sete fases será detalhada:

Fase 1 – Declaração e Classificação dos passos de negócio

Esta primeira etapa é destinada a auxiliar pessoas de negócio na modelagem de processos de negócio.

A declaração dos passos de negócio tem o objetivo de tornar explícito todos os passos que compõe um processo de negócio, sendo assim, com a declaração

queremos dizer, expor ou manifestar a essência de um passo de negócio da maneira mais clara possível.

A classificação dos passos de negócio tem o objetivo de auxiliar na ordenação do fluxo de controle de processos de negócio. Esta classificação é feita de acordo com a estrutura proposta pela arquitetura *RiverFish* estendida por [9]. Nesta classificação devemos dividir os passos de negócio em duas categorias: Restritivos e Condicionais.

Fase 2 – Posicionar os passos do processo de negócio classificados como restritivos

Nesta fase, visando uma melhor performance, deseja-se posicionar os passos de negócio classificados como restritivos, sempre que possível, no início do fluxo de controle. Isto é necessário para minimizar os efeitos causados por esses passos caso eles não sejam atendidos, pois passos de negócios restritivos podem causar interrupção na execução do fluxo de controle, uma vez que sua condição só admite resposta verdadeira para prosseguir.

Fase 3 – Descobrir a relação de dependência entre os passos do processo de negócio

Esta etapa consiste em identificar para cada passo de negócio declarado na Fase 1, a sua relação de dependência com os outros passos do processo.

Fase 4 – Modelar graficamente o fluxo de controle do processo de negócio

Nesta fase é feita a modelagem gráfica. Para isto, conectamos por meio de uma seta orientada os passos de negócio de acordo com a relação de dependência descrita na fase anterior. Quando um passo possui mais de uma seta saindo dele, dizemos que existe um ponto de divisão. Quando existe mais de uma seta entrando em um único passo de negócio, dizemos que existe um ponto de união.

Fase 5 – Descrever o comportamento dos pontos de divisão e união existentes no modelo construído

Nesta fase, o comportamento dos pontos de divisão e união identificados no passo anterior devem ser descritos para permitir a identificação dos padrões de

controle de fluxo. É proposto que esta fase seja desenvolvida juntamente com o profissional de TI, objetivando um entendimento comum do negócio.

Fase 6 – Identificar, classificar os padrões de controle de fluxo e finalizar o projeto conceitual de workflow

A partir da descrição do comportamento dos pontos de divisão e união, os especialistas de negócio em conjunto com os especialistas de TI deverão classificar o padrão que melhor atenda ao comportamento declarado na fase anterior.

Fase 7 – Configuração do modelo baseado em padrões na ferramenta workflow

Nesta última fase, deseja-se configurar o modelo obtido pela execução dos passos anteriores em uma ferramenta *workflow* que melhor atenda às necessidades dos especialistas.

Capítulo 3

RESULTADOS OBTIDOS

Existem diversas linguagens para a definição de processos de negócio. Neste trabalho foi utilizada a NPDL por ela conseguir atender a todas as categorias de padrões de controle de fluxo e por se apoiar no conceito do plano de navegação da arquitetura *RiverFish* e em operadores da álgebra de processos, além de possuir a *NavigationPlan Tool*, que dispõe de bibliotecas que possibilitam o controle da ordem de execução de uma expressão em NPDL.

Como esta facilidade já está implementada e descrita em [4], partiremos do pressuposto de que dada uma expressão em NPDL, que representará um processo de negócio, sabemos qual o comportamento e como simular este processo, tendo como controlar o estado de cada atividade em qualquer instante de, por exemplo, uma simulação.

Assim, para se ter o controle completo de um processo de negócio, basta representá-lo em NPDL. Mas esta linguagem não é muito amigável ao usuário, visto se tratar de expressões algébricas.

A BPMN, porém, é uma notação já conhecida entre os profissionais de negócio e com elementos que possuem o comportamento bem definido. No entanto, esta notação não nos fornece nenhuma maneira de controlar a execução do processo.

Desta maneira, deseja-se uma linguagem que seja facilmente utilizável, como a BPMN, e que possua fundamentação matemática, como a NPDL. Neste capítulo, isto é apresentado, não criando uma linguagem nova, que seria desconhecida aos profissionais, mas criando um vínculo entre a BPMN e a NPDL. Este vínculo é feito mostrando como estas notações representam os padrões de controle de fluxo.

3.1 Necessidades de um Analista de Negócio

Na seção 2.5, vimos uma seqüência de fases para o desenvolvimento do projeto conceitual de *workflows*. Destas diretrizes podemos definir algumas necessidades que um analista de negócio apresenta ao modelar um processo de negócio.



Figura 3.1: Necessidades de um Analista de Negócio (Baseado em [13])

Da figura 3.1, temos sete fases, que serão descritas e detalhadas a seguir:

1. Declarar os passos de negócio

Nesta fase deve-se declarar os passos de negócio, ou seja, as atividades que compõe o processo. A melhor forma de fazer isto é através de uma tabela em que cada linha representa uma atividade.

2. Classificar segundo Arquitetura *RiverFish*

Cada atividade declarada na fase anterior deve ser classificada segundo a arquitetura *RiverFish* estendida, descrita na seção 2.3.3. Assim, ao final desta segunda fase, teremos todas as atividades e suas classificações.

3. Reposicionar visando melhor desempenho

Uma vez que as atividades já foram classificadas, é desejável que as restritivas fiquem o mais próximo possíveis do começo do fluxo. Isso diminui os efeitos causados por uma interrupção gerada por estas atividades e melhora o desempenho do processo.

4. Declarar as dependências entre os passos de negócio

Nesta fase, para cada atividade declarada na primeira fase, deve-se especificar quais são as dependências entre elas, ou seja, quais atividades precisam ser executadas para que uma outra atividade esteja habilitada para a execução.

Uma forma de declarar esta dependência é, para cada atividade declarada na fase 1, especificar quais atividades será habilitadas para a execução após o término de uma determinada atividade.

Esta fase, assim como as próximas, não dependem das fases 2 e 3 pois estas visam apenas o melhor desempenho do processo.

5. Identificar os padrões de controle de fluxo

Com as dependências entre as atividades declaradas na fase anterior, devemos identificar os padrões de controle de fluxo. Mais detalhes sobre estes padrões está descritos na próxima seção.

6. Fazer a diagramação em notação BPMN

Uma vez que as dependências entre as atividades já são conhecidas, deve-se fazer o diagrama em notação BPMN. Mais detalhes sobre como representar os padrões identificados na fase anterior está descrito na próxima seção.

7. Simular e gerar estatísticas do processo de negócio

Com o processo já descrito em BPMN, deve-se ser possível a realização de uma simulação, assim como, gerar estatísticas. Para isto, pode-se utilizar a NPDL, que foi descrita na seção 2.4. Mais detalhes sobre a como representar os padrões de controle de fluxo estão descritos na próxima seção deste texto.

3.2 Especificação dos Padrões de Controle de Fluxo em NPDL e BPMN

Aalst, Hofstede, Kiepuszewski e Barros, em [1], identificaram e definiram 20 padrões de controle de fluxo em workflows. Os Padrões de *Workflow* tentam representar as diversas situações recorrentes em fluxos de controle de processos de negócio. Estes padrões foram descritos inicialmente com o objetivo de estabelecer uma maneira para comparar diversas linguagens de representação de *workflow*. Assim, podemos classificar as linguagens de acordo com os padrões que ela é capaz de expressar por meio de sua sintaxe.

Estes padrões estão divididos em seis categorias:

1. Padrões básicos de controle de fluxo
2. Padrões avançados de ramificação e sincronização
3. Padrões estruturais
4. Padrões de múltiplas instâncias
5. Padrões baseados em estados
6. Padrões de cancelamento

A seguir são apresentados as especificações dos padrões de controle de fluxo usando a notação BPMN e comandos NPDL.

3.2.1 PADRÕES BÁSICOS DE CONTROLE DE FLUXO

Referem-se a construções básicas, presentes na maioria das linguagens de modelagem de workflows.

1) Seqüência

Definição: Uma atividade em um processo de *workflow* é habilitada para a execução após o término de uma outra atividade no mesmo processo.

Exemplo: A atividade B é habilitada para a execução somente após o término da execução da atividade A.

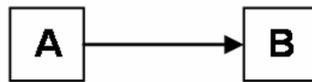


Figura 3.2: Padrão Seqüência (Fonte: [1])

Em NPDL:

SET P = A . B
Seqüência (Fonte: [4])

Em BPMN: Uma seta ordenada, representando o fluxo de seqüência da BPMN, pode ser utilizada para a representação do padrão Seqüência.



Figura 3.3: Seqüência (BPMN)

2) Divisão Paralela

Definição: É um ponto no processo de *workflow* no qual uma única linha de execução se divide em múltiplas linhas de execução permitindo que duas ou mais atividades sejam executadas simultaneamente ou em qualquer ordem.

Exemplo: A atividade A é executada, habilitando a execução de B e C em paralelo (ou seja, em qualquer ordem ou simultaneamente).

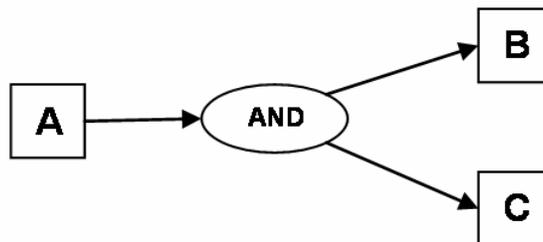


Figura 3.4: Padrão Divisão Paralela (Fonte: [1])

Em NPDL:

SET P = A . (B || C)
Divisão Paralela (Fonte: [4])

Em BPMN: O *gateway* paralelo pode ser usado para a representação do padrão Divisão Paralela.

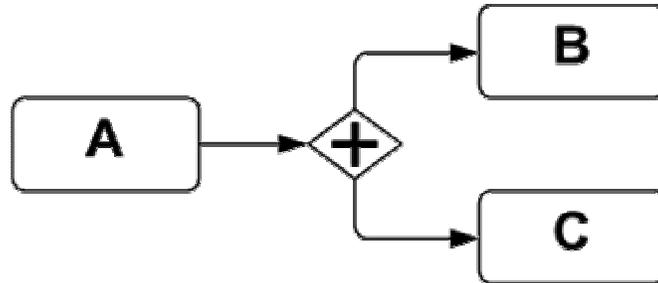


Figura 3.5: Divisão Paralela (BPMN)

3) Sincronização

Definição: É um ponto no processo de *workflow* no qual múltiplas atividades paralelas convergem para uma única linha de execução, sincronizando, assim, as linhas de execução.

Exemplo: A atividade C é habilitada somente depois que A e B tiverem sido executadas.

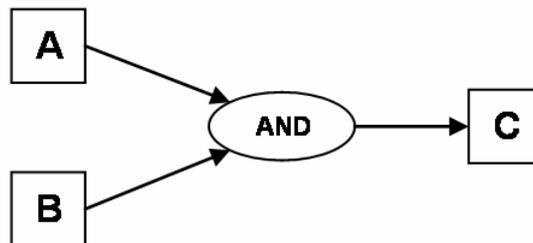


Figura 3.6: Padrão Sincronização (Fonte: [1])

Em NPDL:

SET P = (A || B) . C
Sincronização (Fonte: [4])

Em BPMN: O *gateway* paralelo pode ser usado para a representação do padrão Sincronização.

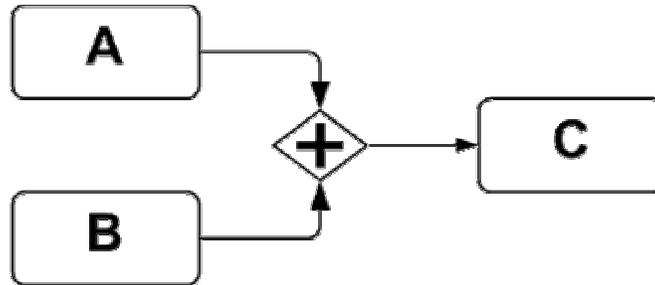


Figura 3.7: Sincronização (BPMN)

4) Escolha Exclusiva

Definição: É um ponto no processo de *workflow* no qual um único caminho é escolhido dentre os possíveis. Essa escolha baseia-se em uma decisão ou em um dado de controle do *workflow*.

Exemplo: As atividades B e C são habilitadas após a execução de A, mas somente uma delas será escolhida implicitamente para execução.

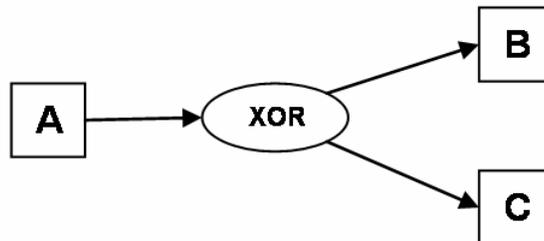


Figura 3.8: Padrão Escolha Exclusiva (Fonte: [1])

Em NPDL:

SET P = A . (%r B + %!r C)
 Sendo r a regra associada à escolha exclusiva
Escolha Exclusiva (Fonte: [4])

Em BPMN: O *gateway* exclusivo pode ser usado para a representação do padrão Escolha Exclusiva.

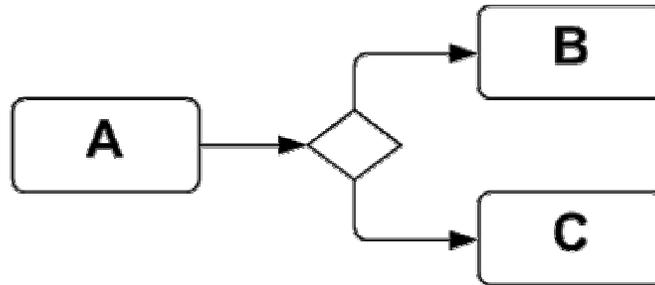


Figura 3.9: Escolha Exclusiva (BPMN)

5) Junção Simples

Definição: É um ponto no processo de *workflow* no qual, dois ou mais caminhos diferentes se unem sem sincronização. Neste caso, assume-se que esses diferentes caminhos não estão sendo executados em paralelo.

Exemplo: Inicialmente, as atividades A e B estão habilitadas. Após a execução de uma delas, C é habilitada.

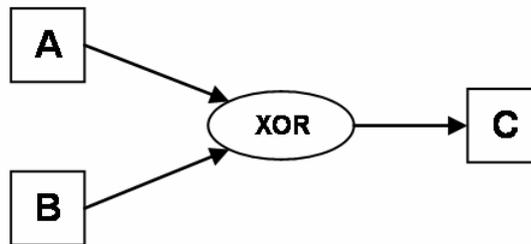


Figura 3.10: Padrão Junção Simples (Fonte: [1])

Em NPDL:

SET P = (%r A + %!r B) . C
 Sendo r a regra associada à escolha exclusiva
 Junção Simples (Fonte: [4])

Em BPMN: O *gateway* exclusivo pode ser usado para a representação do padrão Junção Simples.

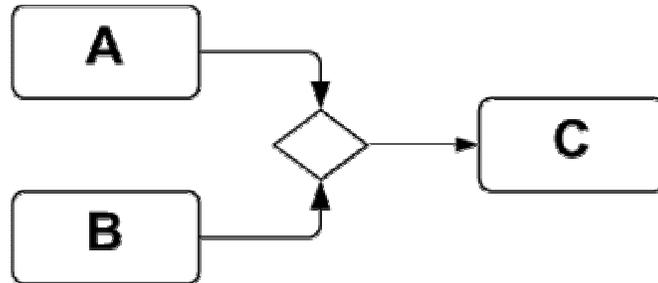


Figura 3.11: Junção Simples (BPMN)

3.2.2 Padrões Avançados de Ramificação e Sincronização

Representam controles de fluxos mais avançados que os padrões básicos

6) Escolha Múltipla

Definição: É um ponto no processo de *workflow* no qual um ou mais caminhos são escolhidos. A escolha baseia-se em uma decisão ou dados de controle do *workflow*.

Exemplo: Inicialmente, uma das três opções estará habilitada: somente a atividade B será executada, somente a atividade C será executada ou a atividade B em paralelo com a atividade C serão executadas.

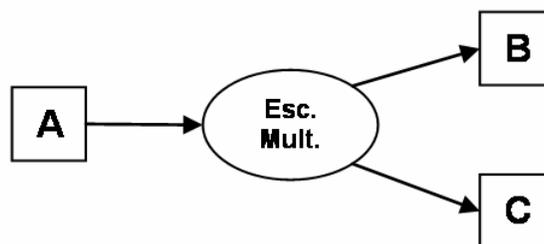


Figura 3.12: Padrão Escolha Múltipla (Fonte [1])

Em NDDL:

SET P = A . (%r1 (B || C) + %!r1 (%r2 B + %!r2 C))
 Sendo r1 e r2 as regras associadas às escolhas exclusivas
Escolha Múltipla (Fonte [4])

Em BPMN: O *gateway* inclusivo pode ser usado para a representação do padrão Escolha Múltipla.

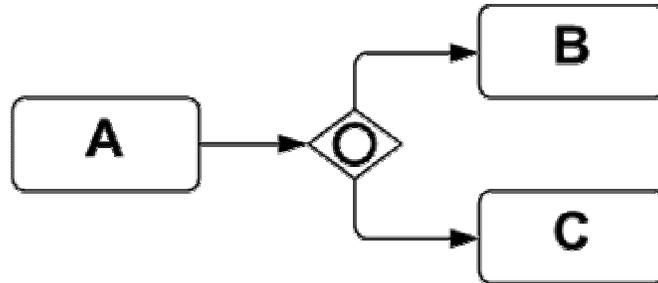


Figura 3.13: Escolha Múltipla (BPMN)

7) Junção Sincronizada

Definição: É um ponto no processo de *workflow* no qual múltiplos caminhos convergem para uma única linha de execução. Se mais de um caminho estiver com atividades ativas, então é feita a sincronização destes caminhos. Se somente um caminho estiver com atividades ativas, então o ramo alternativo de atividade converge sem sincronismo.

Exemplo: Se na escolha múltipla a execução tiver sido encaminhada para a divisão paralela de B e C, então a junção sincronizada aguarda a finalização das ações B e C para habilitar D. Caso a escolha múltipla dispare somente a execução de B, então D será habilitada após a execução de B. Caso a escolha múltipla dispare somente a execução de C, então D será habilitada após a execução de C.

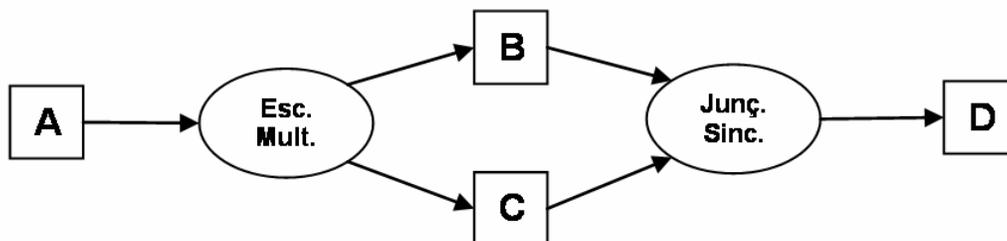


Figura 3.14: Padrão Junção Sincronizada (Fonte [1])

Em NPDL:

$$\text{SET } P = A . (\%r1 (B \parallel C) + \%!r1 (\%r2 B + \%!r2 C)) . D$$

ou

$$\text{SET } P1 = (\%r1 (B \parallel C) + \%!r1 (\%r2 B + \%!r2 C))$$

$$\text{SET } P = A . P1 . D$$

Sendo P1 o mapeamento da escolha múltipla e r1 e r2 as regras associadas às escolhas exclusivas
Junção Sincronizada (Fonte [4])

Em BPMN: O *gateway* inclusivo pode ser usado para a representação do padrão Junção Sincronizada.

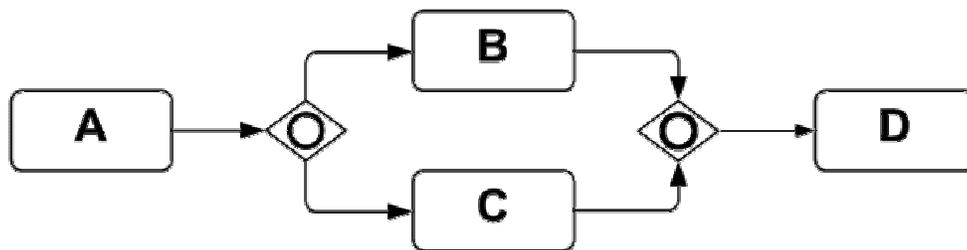


Figura 3.15: Junção Sincronizada (BPMN)

8) Junção Múltipla

Definição: É um ponto no processo de *workflow* no qual, dois ou mais ramos de atividades convergem sem sincronismo. Se mais de um ramo for ativado, de forma concorrente, então a atividade posterior à junção é executada cada vez que é encerrada uma execução de um dos ramos de entrada.

Exemplo: Se na múltipla escolha a execução tiver sido encaminhada para a divisão paralela de B e C, então a junção múltipla executa D, cada vez que uma linha de execução da divisão paralela é finalizada. Caso a escolha múltipla dispare somente a execução de B, então D será habilitada após a execução de B. Caso a escolha múltipla dispare somente a execução de C, então D será habilitada após a execução de C.

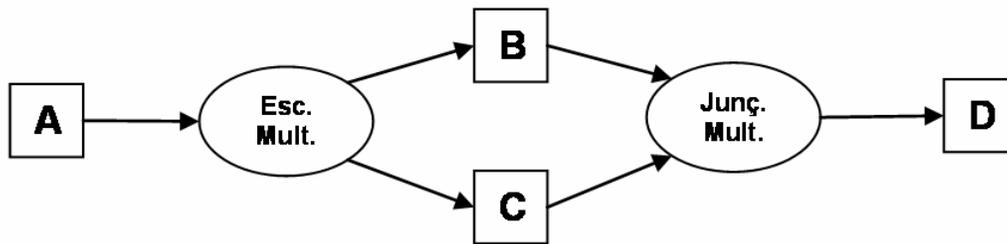


Figura 3.16: Padrão Junção Múltipla (Fonte [1])

Em NPDL:

SET P = A . (%r1 (B || C) + %!r1 (%r2 B + %!r2 C)) & D
ou

SET P1 = (%r1 (B || C) + %!r1 (%r2 B + %!r2 C))

SET P = A . P1 & D

Sendo P1 o mapeamento da escolha múltipla e r1 e r2 as regras associadas às escolhas exclusivas

Junção Múltipla (Fonte [4])

Em BPMN: As setas representando a seqüência de fluxo podem ser usadas para a representação do padrão Junção Múltipla.

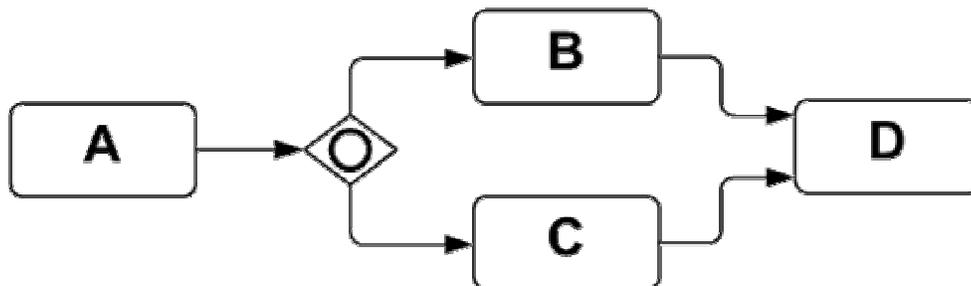


Figura 3.17: Junção Múltipla (BPMN)

9) Discriminador

Definição: é um ponto no processo de *workflow* que espera pelo término da execução de um de seus ramos de entrada antes de habilitar a execução da atividade subsequente. A partir desse momento, ele espera o término da execução dos demais ramos de entrada mas não os leva em consideração. Dessa forma, diferentemente da Junção Múltipla, o número de ramos ativos de entrada do

discriminador não determina o número de vezes que a atividade subsequente será executada.

Exemplo: Se na Escolha Múltipla a execução tiver sido encaminhada para a divisão paralela de B e C, então o Discriminador habilitará a execução de D após o término da linha de execução que for encerrada primeiro. Se B terminar primeiro, então, D será habilitado para execução; após C terminar, diferentemente da Junção Múltipla, D não será habilitado novamente. Caso a escolha múltipla dispare somente a execução de B, então, D será habilitado após a execução de B. Caso a escolha múltipla dispare somente a execução de C, então, D será habilitado após a execução de C

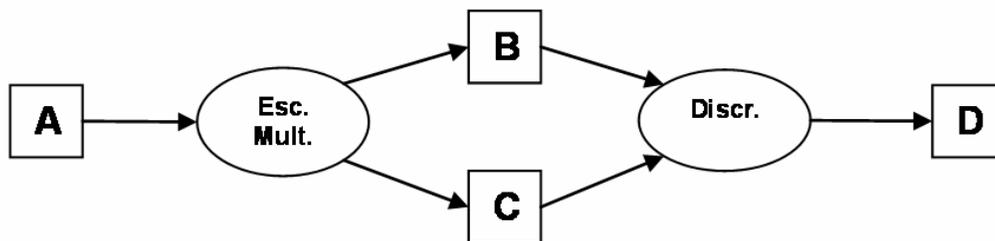


Figura 3.18: Padrão Discriminador (Fonte [1])

Em NPDL:

$$\text{SET } P = A . (\%r1 (B \parallel C) + \%!r1 (\%r2 B + \%!r2 C)) \wedge D$$

ou

$$\text{SET } P1 = (\%r1 (B \parallel C) + \%!r1 (\%r2 B + \%!r2 C))$$

$$\text{SET } P = A . P1 \wedge D$$

Sendo P1 o mapeamento da escolha múltipla e r1 e r2 as regras associadas às escolhas exclusivas
Discriminador (Fonte [4])

Em BPMN: A combinação dos *gateways* inclusivo e exclusivo pode ser usada para a representação do padrão Discriminador.

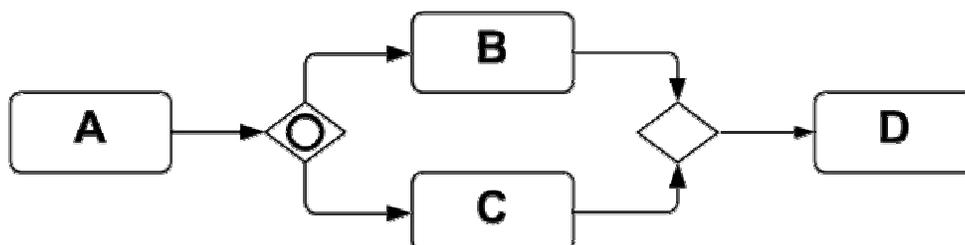


Figura 3.19: Discriminador (BPMN)

9a) N-Discriminador

Definição: É uma generalização do padrão Discriminador. É um ponto no processo de *workflow* que espera pelo término da execução de N de seus ramos de entrada antes de habilitar a execução da atividade subsequente. A partir desse momento, ele espera o término da execução dos demais ramos de entrada mas não os leva em consideração. Dessa forma, ocorre uma sincronização parcial.

Exemplo: Vamos fixar o valor de N como sendo 2. Assim, se na Escolha Múltipla a execução tiver sido encaminhada para a divisão paralela de B, C e D, então o N-Discriminador habilitará a execução de E após o término de quaisquer 2 linhas de execução (B e C ou C e D ou B e D). Quando o terceiro processo terminar sua execução, o N-Discriminador não o levará em consideração.

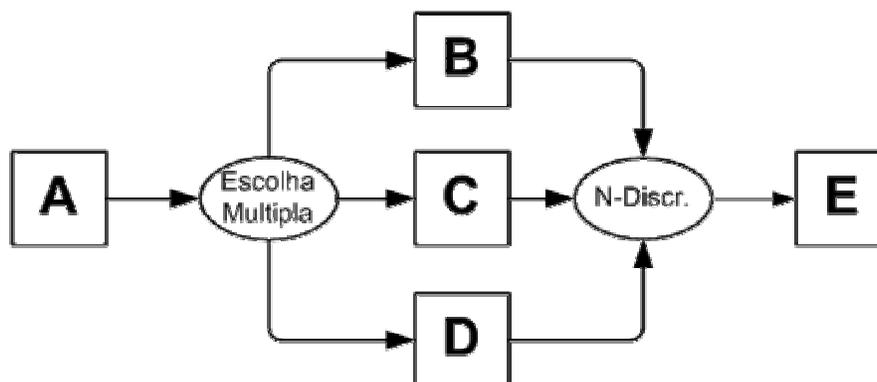


Figura 3.20: Padrão N-Discriminador (Fonte [1])

Em NPD:

$$\text{SET } P = A . P1 \wedge E$$

Sendo P1 o mapeamento da escolha múltipla e as regras associadas a P1 definem o comportamento do N-Discriminador

N-Discriminador

Em BPMN: A combinação dos *gateways* inclusivo e complexo pode ser usada para a representação do padrão N-Discriminador.

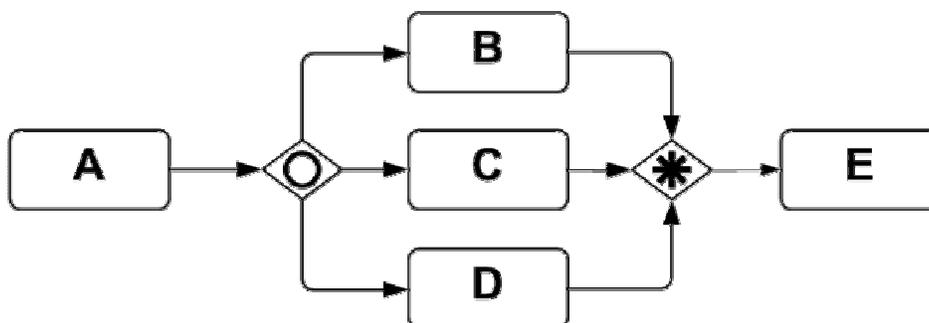


Figura 3.21: N-Discriminador (BPMN) (Fonte: [15])

3.2.3 PADRÕES ESTRUTURAIS

Representam a flexibilidade que as linguagens de modelagem de *workflow* possuem de blocos possuírem paralelismo e múltiplos pontos de entrada e saída.

10) Ciclo Arbitrário

Definição: É um ponto em um processo de *workflow* que permite que uma ou mais atividades sejam feitas repetidamente.

Exemplo: A atividade A é executada seguida pela atividade B. Em seguida, as atividades C e D são habilitadas para a execução. A execução da atividade D representa o ciclo, que possibilita que a atividade B seja executada várias vezes.

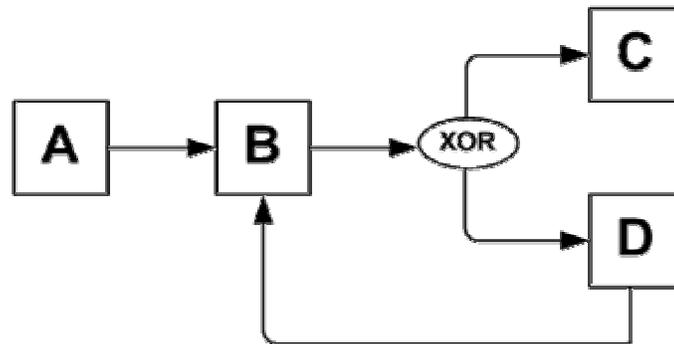


Figura 3.22: Padrão Ciclo Arbitrário

Em NPD:

```

SET P1 = B . (%r C + %!r (D . P1))
SET P = A . P1
  
```

Sendo r a regra associada à escolha exclusiva
Ciclo Arbitrário

Em BPMN: O Ciclo Arbitrário é representado por meio de caminhos cíclicos no fluxo.

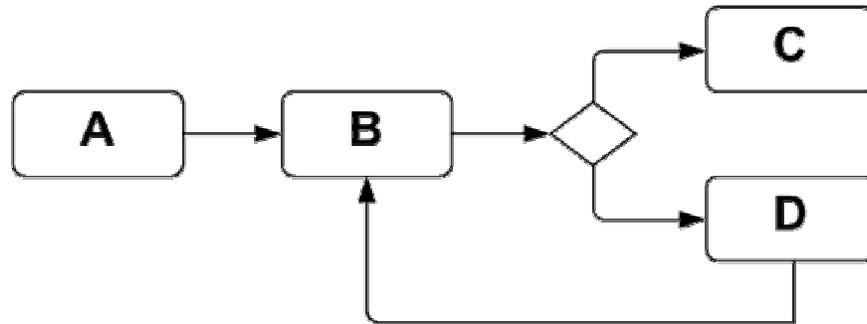


Figura 3.23: Ciclo Arbitrário (BPMN)

11) Terminação Implícita

Definição: Um dado subprocesso pode ser encerrado quando não há mais nada a ser feito, ou seja, um caminho pode ser concluído se não houver mais atividades a serem executadas neste caminho.

Exemplo: Após a execução da atividade A, as atividades B e C são habilitadas. Após a execução da atividade B, a atividade D é habilitada e após a execução desta, este caminho termina. Assim como, após a execução da atividade C, a atividade E é habilitada e após a execução de E, o caminho termina.

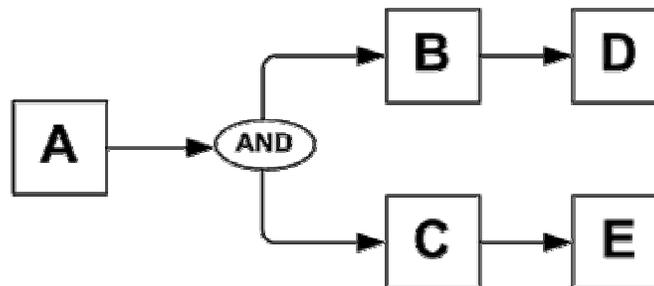


Figura 3.24: Padrão Terminação Implícita (Fonte [1])

Em NPDL: Não existe, em NPDL, um termo que indique o estado final do processo

Terminação Implícita (Fonte [4])

Em BPMN: O evento de Fim pode ser usado para a representação do padrão Terminação Implícita.

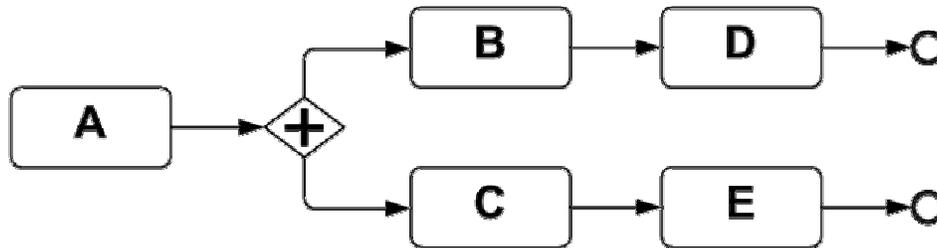


Figura 3.25: Terminação Implícita (BPMN)

3.2.4 PADRÕES DE MÚLTIPLA INSTÂNCIA

Padrões de múltiplas instâncias são utilizados para descreverem a execução de mais de uma instância de uma determinada tarefa atômica.

12) Múltiplas Instâncias Sem Sincronização

Definição: Este padrão representa um ponto no *workflow* no qual múltiplas instâncias de uma atividade podem ser criadas, isto é, há uma facilidade para gerar novas linhas de execução. Cada uma das linhas de execução gerada é independente das demais. Além disso, não há a necessidade de sincronizar estas linhas.

Exemplo: Após a execução da atividade A, múltiplas instâncias da atividade B são criadas, gerando assim, várias linhas de execução independentes.

Em NPD:

SET P = A . (B . C) ?*
 MI Sem Sincronização (Baseado em: [4])

Em BPMN: Os padrões Divisão Paralela e Escolha Exclusiva podem ser usados para representar este padrão. Após a execução da atividade B, a atividade C é executada e outra instância da atividade B pode ser criada ou não, dependendo da regra associada à Escolha Exclusiva, mas sempre que a atividade B for executada, uma nova linha de execução independente será criada.

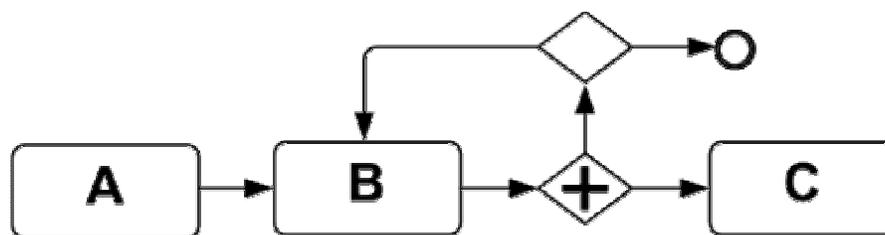


Figura 3.26: MI Sem Sincronização (BPMN)

13) Múltiplas Instâncias com conhecimento prévio em tempo de projeto

Definição: Ocorre quando, em uma instância de processo, uma atividade é habilitada múltiplas vezes, sendo que o número de instâncias a serem executadas, da atividade em questão, é conhecido em tempo de projeto. Após o término da execução de todas as instâncias, uma outra atividade do *workflow* é habilitada para execução

Exemplo: Após a execução da atividade A, a atividade B é executada 3 vezes, e após a execução dessas 3 vezes da atividade B, a atividade C é executada. As múltiplas instâncias da atividade B devem ser executadas paralelamente.

Em NPDL:

SET P = A . B ?3 . C

MI Com Conhecimento Prévio em Tempo de Projeto (Baseado em: [4])

Em BPMN: O símbolo de múltipla instância é adicionado à atividade B, mostrando que esta atividade será executada n vezes e em paralelo.



Figura 3.27: MI Com Conhecimento Prévio em Tempo de Projeto (BPMN) (Fonte: [15])

14) Múltiplas Instâncias com conhecimento prévio em tempo de execução

Definição: Ocorre quando, em uma instância de processo, uma atividade é habilitada múltiplas vezes, sendo que o número de instâncias a serem executadas, da atividade em questão, é conhecido em tempo de execução. Após o término da

execução de todas as instâncias, uma outra atividade do *workflow* é habilitada para execução.

Exemplo: Após a execução da atividade A, a atividade B é executada um número de vezes que será decidido em tempo de execução e, após a execução dessas atividades B, a atividade C é executada.

Em NPDL:

$$\text{SET } P = A . B ?f . C$$

Sendo f uma função que retorna um número inteiro e maior que zero

MI Com Conhecimento Prévio em Tempo de Execução (Baseado em: [4])

Em BPMN: O símbolo de ciclo é adicionado à atividade B, mostrando que esta atividade será executada n vezes.

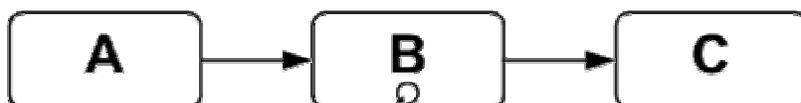


Figura 3.28: MI Com Conhecimento Prévio em Tempo de Execução (BPMN) (Fonte: [15])

15) Múltiplas Instâncias sem conhecimento prévio em tempo de execução

Definição: Ocorre quando, em uma instância de processo, uma atividade é habilitada múltiplas vezes, sendo que o número de instâncias a serem executadas, da atividade em questão, não é conhecido nem em tempo de projeto nem em tempo de execução, ou seja, ele não é conhecido antes do momento de criação das instâncias da atividade. Após o término da execução de todas as instâncias criadas, uma outra atividade subsequente do *workflow* é habilitada para execução.

Exemplo: Após a execução da atividade A, a atividade B é executada várias vezes até que uma condição seja satisfeita e, após estas execuções, a atividade C será executada.

Em NPDL:

$$\text{SET } P = A . B ?* . C$$

MI Sem Conhecimento Prévio (Baseado em: [4])

Em BPMN: Após a execução da atividade A, as atividades B e B' são habilitadas para a execução. A atividade B' determina se mais cópias de B devem ser criadas. Se mais cópias devem ser criadas, a linha de execução que habilitou B será encerrada Através do padrão Terminação Implícita.

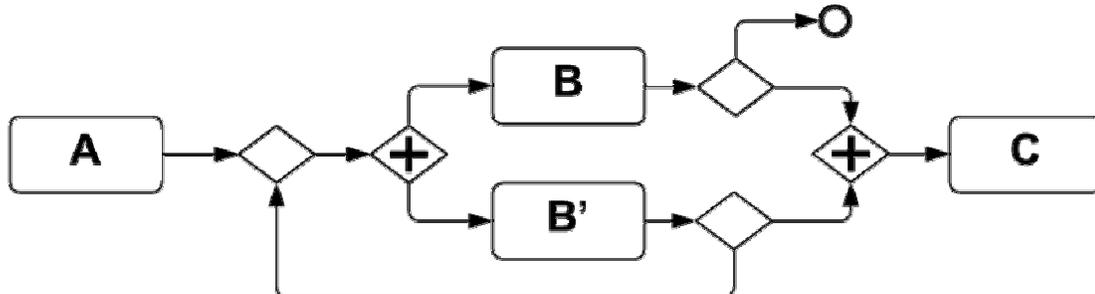


Figura 3.29: MI Sem Conhecimento Prévio (BPMN) (Fonte [15])

3.2.5 PADRÕES BASEADOS EM ESTADOS

Estendem a expressividade de linguagem de *workflow* por meio da representação de estados.

16) Escolha Postergada

Definição: É um ponto no processo de *workflow* no qual um entre vários ramos de atividades é escolhido. Diferentemente da Escolha Exclusiva, na Escolha Postergada, a escolha não é feita de modo explícito, ou seja, não se baseia em dados ou em uma decisão. A escolha se baseia em informações que podem não estar disponíveis no momento que este ponto é atingido. Assim, a escolha é postergada até a ocorrência de algum evento.

Exemplo: Após a execução da atividade A, as atividades B e C são habilitadas, mas somente uma delas será escolhida explicitamente para execução.

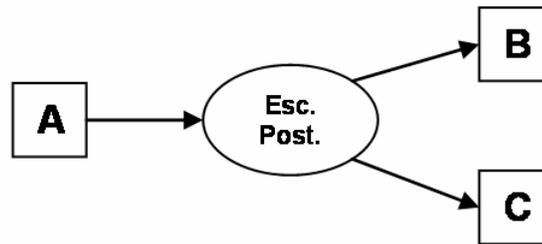


Figura 3.30: Padrão Escolha Postergada (Fonte [1])

Em NPDL:

SET P = A . (B + C)
Escolha Postergada (Fonte [4])

Em BPMN: Para representar a escolha postergada, o *gateway* exclusivo baseado em dados é utilizado. Eventos intermediários são adicionados para controlar a escolha que será feita.

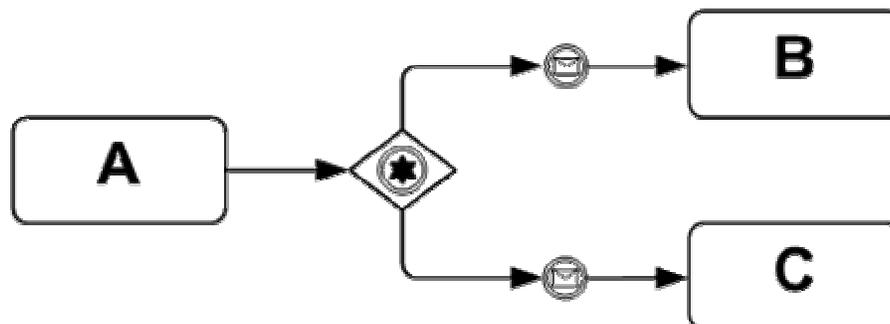


Figura 3.31: Escolha Postergada (BPMN) (Fonte [15])

17) Roteamento Paralelo Entrelaçado

Definição: Ocorre quando um conjunto de atividades é executado em uma ordem arbitrária, que é decidida em tempo de execução, e duas atividades nunca são executadas ao mesmo tempo, mas todas serão executadas.

Exemplo: A atividade A é executada, habilitando a execução de B, seguida pela execução de C, ou a execução de C seguida pela execução de B. Após o término da execução de B e C (ou C e B), a atividade D é habilitada para execução;

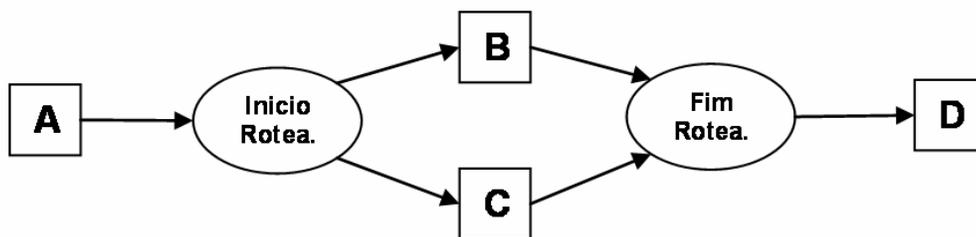


Figura 3.32: Padrão Roteamento Paralelo Entrelaçado (Fonte [1])

Em NPDL:

SET P = A . (B | * C) . D
 Roteamento Paralelo Entrelaçado (Fonte [4])

Em BPMN: Processos *Ad-Hoc* são utilizados para representar este padrão. As atividades contidas no subprocesso *Ad-Hoc* serão todas executadas, mas uma por vez.

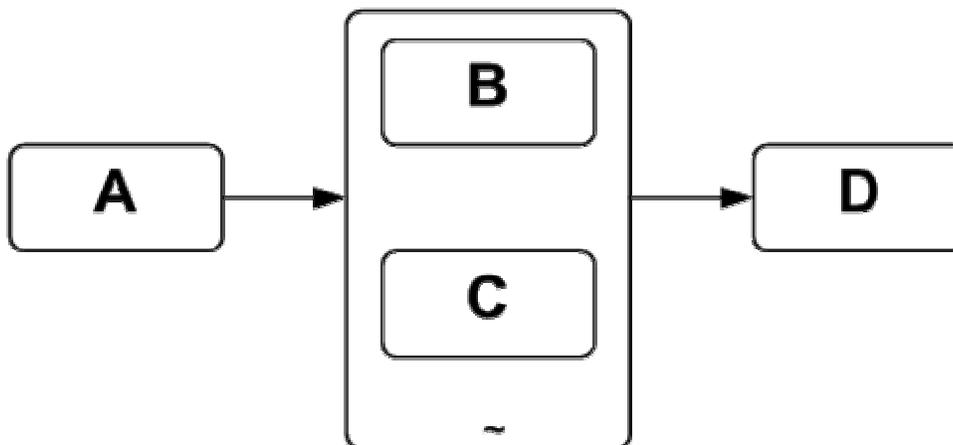


Figura 3.33: Roteamento Paralelo Entrelaçado (BPMN)

18) Marco

Definição: Ocorre quando a habilitação de uma atividade é dependente de um estado específico da instância de um processo. Ou seja, a atividade só é habilitada se um certo marco tiver sido atingido e ainda não estiver expirado.

Exemplo: Considere três atividades A, B e C. A atividade A somente será habilitada depois que a atividade B tiver sido executada, mas se C não tiver sido executada ainda, ou seja, A não estará habilitada antes da execução de B e nem

depois da execução de C. O estado entre B e C é modelado pelo lugar M. Este lugar é o marco para A.

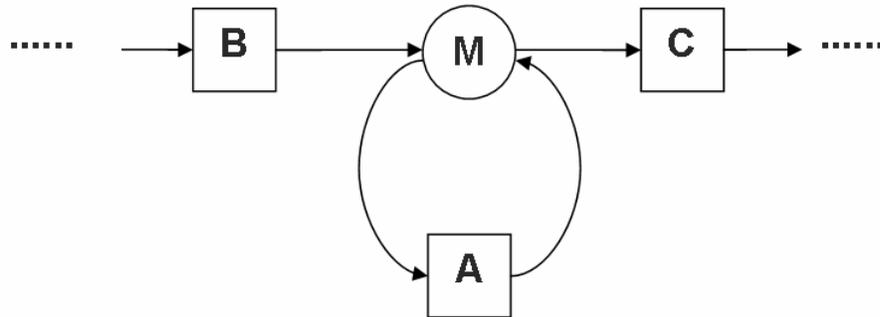


Figura 3.34: Padrão Marco (Fonte [1])

Em NPDL:

$$\begin{aligned} \text{SET } P1 &= A \cdot P1 + A \cdot C \\ \text{SET } P &= B \cdot P1 + B \cdot C \\ \text{Marco (Fonte [4])} \end{aligned}$$

Em BPMN: A combinação dos padrões de junção e escolha postergada pode ser utilizada para representar o padrão Marco.

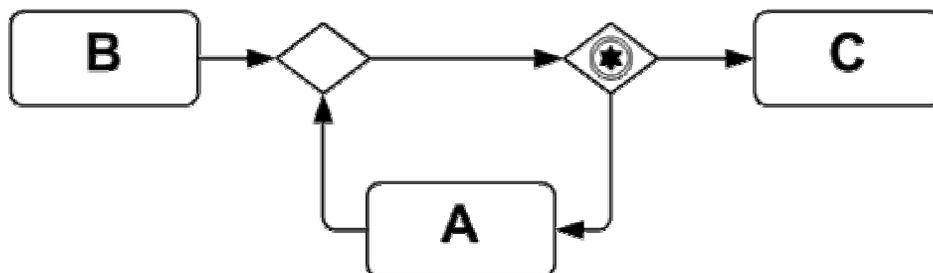


Figura 3.35: Marco (BPMN)

3.2.6 PADRÕES DE CANCELAMENTO

Modelam o cancelamento de atividades condicionado à ocorrência de um determinado evento.

19) Atividade Cancelável

Definição: Ocorre quando uma atividade já habilitada pode ser desabilitada durante a execução da instância do processo. Quando a atividade é desabilitada, é necessário remover a linha que controla a execução da atividade em questão.

Exemplo: Esta é uma maneira de tratar exceções durante a execução do processo. Uma atividade A é executada, habilitando a execução das atividades B e C. Quando a atividade B terminar, ela lançará uma exceção. Se a atividade C estiver sendo executada quando isto acontecer, ela será interrompida e o fluxo de execução seguirá através de uma nova linha de execução.

Em NPDL: O símbolo “#” representa o símbolo de *deadlock* da Álgebra de processos.

$$\text{SET } P = A . (B \parallel ((C . D) + \#))$$

Atividade Cancelável (Fonte: [4])

Em BPMN: Quando a atividade B terminar, o evento intermediário disparará um gatilho, avisando o evento intermediário acoplado a atividade C que interromperá a execução desta, caso esteja sendo executada, e continuará a execução do fluxo através do fluxo de seqüência ligado ao evento associado a atividade C.

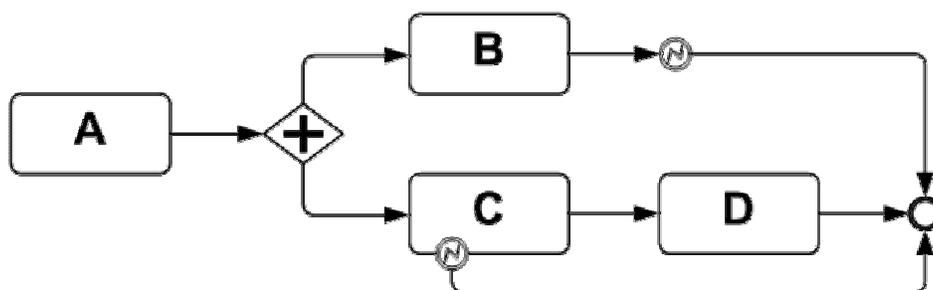


Figura 3.36: Atividade Cancelável (BPMN) (Fonte: [15])

20) Caso Cancelável

Definição: Ocorre quando a instância do *workflow* pode ser removida completamente.

Exemplo: Após o término da atividade B ou do término da atividade D, todo o processo é cancelado.

Em NPD: Toda instância de processo definido em NPD pode ser cancelado em qualquer instante de sua execução.

Caso Cancelável (Fonte: [4])

Em BPMN: O evento de término é usado para modelar este padrão.

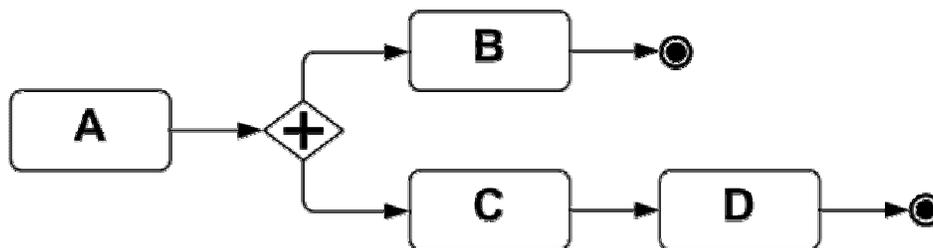


Figura 3.37: Caso Cancelável (BPMN)

Capítulo 4

CONCLUSÃO

A modelagem de negócio é a ferramenta utilizada para fornecer uma visão ampla e possibilitar a otimização de processos mas, para que isto seja possível, é necessário que esta modelagem seja feita de forma compreensível a todos. Para tal, utiliza-se a BPMN.

Para uma maior compreensão, além da simples descrição do negócio, é necessário que seja feita a simulação deste e, para sua análise, de ser possível gerar estatísticas sobre a execução do fluxo, evidenciando pontos que possam estar comprometendo o desempenho do processo.

Para tanto, é necessário descrever o negócio utilizando uma linguagem que tenha fundamentos garantindo-se, assim, a corretude dos resultados. Uma linguagem que pode ser utilizada é a NPD, que tem seus fundamentos apoiados na Álgebra de Processos.

Como a BPMN já é conhecida e utilizada entre os profissionais da área e a NPD é fundamentada em Álgebra de Processos e já possui ferramentas para a execução de processos de negócio, fora apresentado, neste trabalho, a maneira de representar alguns padrões utilizando a BPMN. Mostrou-se, também, como representar estes mesmos padrões utilizando a NPD. Desta maneira, demonstramos que é possível representar um mesmo processo através de diagramas em BPMN e de expressões em NPD.

4.1 Trabalhos Futuros

Apesar de descrevermos na seção 3.2 como representar os diversos padrões na notação BPMN e na linguagem NPD, a identificação destes padrões nos processos e a sua tradução para expressões NPD nem sempre são tarefas triviais. Desta forma, como um possível trabalho futuro tem se o desenvolvimento de

uma ferramenta gráfica que possibilite esta tradução de uma maneira simples e que atenda às necessidades descritas na seção 3.1.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] AALST, W.M.P. van der, HOFSTEDE, A.H.M. ter, KIEPUSZEWSKI, B., BARROS, A.P. **Workflow Patterns. In Distributed and Parallel Databases**, Vol. 14, No. 1, Jul/2003, p. 5-51.

[2] BERGSTRA, J. A. **Handbook of Process Algebra**. Elsevier Science, Amsterdam, 2001.

[3] BRAGHETTO, K. R.; TAKAI, O. K.; FERREIRA, J. E.; PU, C. **Controlling processes in relational data model with NPDL and NavigationPlanTool**. Technical report, Departamento de Ciência da Computação, Universidade de São Paulo, 2005. Based on accepted work for DEMO section in SBBD2005. Disponível em <http://www.ime.usp.br/~jef/kelly/NPDL.pdf>.

[4] BRAGHETTO, K. R. **Padrões de Fluxo de Processo em Banco de Dados Relacionais**, Dissertação de Mestrado, Instituto de Matemática e Estatística da Universidade de São Paulo.

[5] DAVENPORT, T. **Reengenharia de Processo**. 5 ed. Rio de Janeiro, Campus.1994.

[6] DELPHI GROUP. BPM 2005: market milestone report, 2005. Disponível em www.delphigroup.com.

[7] FERREIRA, J. E.; TAKAI, O. K.; PU, C. **Integration of Business Processes with Autonomous Information Systems: a case study in Government Services**. In: INTERNATIONAL IEEE CONFERENCE ON E-COMMERCE TECHNOLOGY 2005 (CEC-05), 7., 2005, Munich, Germany.

[8] FERREIRA, J. E.; TAKAI, O. K.; PU, C. **Integration of Collaborative Information System in Internet Applications using RiverFish Architecture**. In: INTERNATIONAL CONFERENCE ON COLLABORATIVE COMPUTING: NETWORKING APPLICATIONS AND WORK SHARING, 2005, San Jose, USA

[9] MILLA. J. M. **Classificação de Regras de Negócio através de ontologias: Um Estudo de Caso**. Dissertação de Mestrado apresentado ao Instituto de Pesquisas Tecnologias do Estado de São Paulo – IPT, 2005.

[10] SMITH, H.; FINGAR, P. **BUSINESS PROCESS MANAGEMENT: the third wave**. Tampa: Meghan-Kffer Press, 2003.

[11] VON HALLE, B. **Business Rules Applied: Building Better Systems Using the Business Rule Approach**, Wiley Computer Publishing, 2002.

[12] WHITE, S. A. **Introduction To BPMN**.

[13] ZULIANI, D. **Projeto Conceitual de Workflow**, Dissertação de Mestrado, Instituto de Pesquisas Tecnológicas do Estado de São Paulo.

[14] **Workflow Patterns**. Disponível em: <http://www.workflowpatterns.com>

[15] **Business Process Modeling Notation Specification**. Disponível em <http://www.bpmn.org>