

1 Introdução

Com o crescimento da internet e a demanda por sistemas complexos e distribuídos, surgiu o paradigma de agentes, visando atender à necessidade de outras abstrações para o desenvolvimento de sistemas. As pesquisas em engenharia de software nessa área cresceram, produzindo linguagens de modelagem, modelos conceituais, métodos de desenvolvimento, arquiteturas, frameworks de implementação, linguagens de desenvolvimento, frameworks de teste e outras tecnologias para ajudar no desenvolvimento de sistemas multi-agentes (SMA).

Em sistemas multi-agentes, diversos agentes interagem visando atingir objetivos individuais ou coletivos. Muitas vezes os agentes participantes das interações são desenvolvidos separadamente, inclusive por diferentes desenvolvedores, o que dificulta o conhecimento antecipado da estrutura interna de cada agente e do comportamento do conjunto de agentes.

Para lidar com a imprevisibilidade do comportamento do SMA's, são usados mecanismos de governança, que fornecem estruturas como convenções, restrições e normas para regular as interações. Esses mecanismos ajudam a controlar o comportamento em tempo de execução dos agentes e podem guiar o desenvolvimento dos mesmos.

Um desses mecanismos, a abordagem de leis de regulação das interações entre agentes em sistemas multi-agentes abertos, visto em (Paes, 2005a; Paes et al., 2005b), propõe um modelo conceitual baseado em normas, protocolos de interação e relógios com controle de tempo, interligados por eventos, para a especificação das regras de regulação separada da especificação do SMA em si. Essa abordagem provê um controle centralizado do comportamento do sistema multi-agente através de um sistema mediador que monitora as interações e aplica a lei. A especificação da lei é feita em código XML, chamado XMLaw, que é interpretada pelo sistema mediador.

O objetivo deste trabalho é o de propor uma linguagem de modelagem baseada em UML (OMG, URL, 2007) e mapeada para código XMLaw para facilitar a especificação das leis da abordagem de regulação de interações entre agentes.

Além da linguagem, o trabalho propõe também uma ferramenta de transformação automática dos modelos dos diagramas em código completo XMLaw.

1.1. Motivação

O desenvolvimento de sistemas multi-agentes se dá muitas vezes de forma descentralizada e distribuída. Frequentemente esses sistemas são abertos, ou seja, novos agentes podem entrar no sistema sem que se tenha conhecimento prévio das regras de interação do sistema (LESKI, URL, 2007a). Esses tipos de sistemas são chamados Sistemas Multi-Agentes abertos e representam, possivelmente, a mais importante área de aplicação de Sistemas Multi-Agentes (Wooldridge et al., 1999).

A abordagem de leis de regulação das interações entre agentes em sistemas multi-agentes abertos (Paes, 2005a; Paes et al., 2005b) permite a separação entre o desenvolvimento de agentes e o desenvolvimento dos mecanismos de controle das interações entre os agentes. Esse mecanismo é feito a partir da especificação das leis em código XML (chamada XMLaw), que são interpretadas por um sistema mediador, responsável por monitorar as interações e garantir o cumprimento das leis.

A especificação das leis não é feita a partir de um processo bem definido, nem é adotada uma notação de alto nível para tal. O desenvolvedor das leis deve conhecer bem o domínio do sistema, os conceitos da abordagem de regulação e a tecnologia XML para criá-las. A partir de rascunhos feitos sem um conjunto de regras, símbolos e padrões, o desenvolvedor das leis deve criar (ou editar) um arquivo texto com as leis (código XMLaw). A tarefa de modelar a lei pode ser grande e custosa, devido à complexidade natural dos SMA's (veja um exemplo de especificação de lei no Anexo B).

Se o desenvolvimento das leis for feito por uma equipe, a comunicação e compreensão das leis ficam limitadas à tarefa mental de entendimento de arquivos textos ou de modelos gerados sem uma notação de alto nível. Além disso, conforme a lei do sistema evolui, surgem também os problemas relativos à alteração das especificações, ou seja, é preciso interpretar o arquivo texto e possivelmente ter o retrabalho de recriar manualmente a modelagem das leis, novamente sem uma padronização formal.

Algumas abordagens de regulação das interações entre os agentes encontradas na literatura propõem o uso de linguagens declarativas baseadas em modelos abstratos para a especificação das regras de regulação. Outras propõem que, além da especificação textual, parte da especificação seja gráfica, facilitando a compreensão e a comunicação dos modelos. Contudo, nenhuma dessas abordagens propõe que essa especificação seja feita com o uso de uma linguagem gráfica baseada em um padrão mundial, que é a UML (OMG, URL, 2007).

Dado que a modelagem gráfica permite uma comunicação eficaz dos modelos conceituais, além de facilitar a compreensão do sistema (Booch et al., 2006), este trabalho pretende melhorar a compreensão e a comunicação dos modelos de leis. Isso será possível através da proposição de uma padronização da modelagem visual das leis de regulação de interações entre agentes que aproveita a familiaridade da comunidade de desenvolvimento de software com os diagramas e elementos da UML.

1.2. Solução Proposta

Esta dissertação propõe a padronização da modelagem das leis de interação entre os agentes através de uma *linguagem visual* baseada na UML, chamada LawML, que permita especificar as diferentes visões da lei – estrutural e dinâmica – em diferentes diagramas. Essas diferentes visões permitem captar diferentes aspectos do modelo em diferentes diagramas, com cada diagrama abstraindo um conjunto de aspectos comuns.

Com a linguagem LawML, o desenvolvedor das leis pode realizar a tarefa de especificação com maior foco no domínio do sistema, sem se ater ao conhecimento específico de tecnologia XML. A tarefa de especificação pode ser facilitada e a comunicação dos modelos mais eficaz. Além disso, a padronização da modelagem pode permitir que a manutenção das leis seja feita de maneira mais eficaz e eficiente, permitindo evitar o retrabalho de especificação das leis.

Para que a linguagem seja funcional, será proposta juntamente com um conjunto de regras de transformação dos modelos de LawML para código XMLLaw. Será apresentada também a *LawGenerator*, uma ferramenta de transformação automática dos modelos gráficos dos diagramas de LawML em modelos descritos em XMLLaw. Os modelos gráficos devem ser gerados através de uma ferramenta CASE baseada na UML (OMG, URL, 2007) que permita a exportação dos dados para um arquivo no formato XML.

Dessa maneira, o desenvolvedor das leis de regulação poderá utilizar a modelagem gráfica com base nas regras da linguagem LawML em uma ferramenta CASE UML e transformar os modelos gráficos em código textual XML com a ferramenta *LawGenerator*. A verificação de consistência entre os elementos dos modelos gerados fica a cargo do desenvolvedor de leis, que pode seguir as orientações do *Laudo de Consistência*, apresentado junto com a linguagem de modelagem.

1.3. Contribuições

São esperadas as seguintes contribuições ao desenvolvimento e à manutenção de leis de regulação de interações entre agentes em sistemas multi-agentes abertos:

1. LawML

Uma linguagem de notação para a especificação das leis de regulação de interações entre agentes, relacionada à abordagem de leis (Paes, 2005a) e baseada na UML (Booch et al., 2006). A UML é uma linguagem mundialmente conhecida e já é adotada como padrão para especificação de software.

LawML divide a modelagem das leis em uma visão estrutural e uma visão dinâmica, cada uma com seus diagramas. O grande diferencial em relação às outras abordagens de regras de regulação está no uso de uma linguagem gráfica para a especificação de leis.

2. Regras de transformação de modelos LawML para código XMLaw

O trabalho propõe um conjunto de regras para a transformação de modelos especificados em LawML para o formato XMLaw. Todos os elementos, propriedades e relacionamentos são mapeados para código XMLaw. Para conectar os elementos e relacionamentos dos diferentes diagramas, o trabalho propõe um breve laudo de consistência – um conjunto de regras que facilitam a validação das leis através da rastreabilidade entre elementos do modelo.

As regras de transformação e o laudo de consistência permitem a criação de ferramentas de geração de código XMLaw com base na modelagem de LawML.

3. LawGenerator – Geradora de Leis

Uma ferramenta de transformação automática de especificações de leis em LawML para XMLaw, com base nas regras de transformação. A modelagem gráfica deve ser feita com o uso de uma ferramenta CASE

baseada na UML (OMG, URL, 2007). Posteriormente os dados são exportados para um arquivo no formato XML.

A ferramenta *LawGenerator* lê esse arquivo XML, realiza a transformação do modelo e gera o arquivo XML com a lei completa especificada em XMLaw, que é usada pelo mecanismo de software de mediação das interações e aplicação da lei M-Law.

1.4. Organização da Dissertação

O restante deste trabalho está estruturado conforme listado a seguir.

– **Capítulo 2** – são apresentados os conceitos básicos de Governança de SMA's necessários para o entendimento da dissertação.

– **Capítulo 3** – mostra uma visão geral da área de pesquisa no qual esta dissertação se situa. O propósito desse capítulo é facilitar o entendimento das contribuições desta dissertação.

– **Capítulo 4** – apresenta a linguagem LawML. A notação, o mapeamento dos elementos, das propriedades e dos relacionamentos para código são apresentados juntamente com um problema-exemplo, para ilustrar como os conceitos são especificados na linguagem. Além disso, esse capítulo apresenta um breve laudo de consistência entre elementos do modelo.

– **Capítulo 5** – apresenta a ferramenta de geração automática de código. Esse capítulo explica o que ela é, como ela foi feita e como usá-la.

– **Capítulo 6** – um exemplo de modelagem de lei de sistema multi-agente é utilizado para ilustrar o desenvolvimento das leis com o uso da abordagem. Além da especificação desse exemplo, é apresentada a transformação do modelo em código no formato XMLaw. O código completo gerado pela transformação encontra-se no Anexo B.

– **Capítulo 7** – apresenta os resultados, conclusões e trabalhos futuros.