

## 7

### Considerações Finais

#### 7.1.

##### Conclusão

Devido ao crescimento de SMA, se torna necessário criar linguagens, metodologias e processos de desenvolvimento que facilitem ao desenvolvedor na construção deste tipo de sistemas. Após analisar na literatura um conjunto variado de processos e metodologias, foi possível perceber que muitas delas não contemplavam todo o processo de desenvolvimento ou não apresentavam de forma muito clara como a partir de uma dada modelagem se obtém o seu código correspondente.

O processo de desenvolvimento apresentado neste trabalho compreende todo o ciclo de desenvolvimento de uma aplicação, ou seja, desde a sua modelagem até sua implementação, utilizando a abordagem MDA. Este processo pode ser dividido em três principais etapas: (i) modelagem do SMA utilizando a linguagem MAS-ML, que é uma linguagem específica para SMA e independente de plataforma; (ii) geração de modelos UML no formato XMI, dependentes de plataforma e baseados no framework ASF, a partir dos modelos MAS-ML; (iii) geração do código a partir dos modelos UML gerados na etapa anterior.

Cada etapa existente no processo proposto está relacionada a uma etapa de MDA. Os modelos MAS-ML criados por serem independente de plataforma estão relacionados aos modelos PIM de MDA. Os modelos UML gerados na segunda etapa do processo estão relacionados aos modelos PSM. A transformação entre os modelos MAS-ML e UML e dos modelos UML em código é realizada de forma clara e objetiva utilizando-se a tecnologia XMI.

A criação da ferramenta permite o desenvolvimento de sistemas multi-agentes seguindo o processo proposto neste trabalho. O UML XMI gerado através desta ferramenta pode ser usado em outras ferramentas CASE já existentes, que utilizam a linguagem de modelagem UML e tem como funcionalidade a importação de documentos XMI. Durante este trabalho, o UML XMI criado pela ferramenta foi testado em outras ferramentas como por

exemplo, Poseidon e Together. No teste realizado com o Poseidon, a geração do código com base no XMI foi realizada sem qualquer perda de informação, ou seja, todas as informações presentes do documento foram traduzidas corretamente para o código. No caso do Together, pelo fato desta ferramenta não ser capaz de interpretar a implementação de métodos presente no XMI, o código gerado não apresentava a implementação dos construtores.

## **7.2. Principais Contribuições**

Neste trabalho foram apresentadas contribuições para a área de engenharia de software baseada em agentes, conforme já mencionado na Seção 1, são elas:

1. Criação de um processo de desenvolvimento de sistemas multi-agentes, que contém as seguintes características:

- Utilização da abordagem MDA no processo de desenvolvimento permitindo a geração de diferentes modelos para cada etapa de transformação, apresentando os diferentes pontos de vista do sistema.

Além de se beneficiar das vantagens decorrentes do uso de MDA, são elas:

- portabilidade: os modelos MAS-ML são portáveis para diferentes plataforma de implementação. Tais modelos descrevem o SMA sem incluir detalhes de implementação dos mesmos;
- reusabilidade: um modelo MAS-ML pode ser reutilizado por diferentes desenvolvedores que desejem implementar o sistema utilizando diferentes plataformas;
- interoperabilidade: se dá pelo fato dos modelos serem descritos em XMI. Os modelos MAS-ML e UML são interoperáveis por devido ao uso do MAS-ML XMI e UML XMI.
- baixo acoplamento: existente entre o conceitual da aplicação e o modelo da solução (modelo computacional). Na abordagem proposta os modelos MAS-ML são modelos PIM que não incluem detalhes relacionados a plataforma escolhida. Tais detalhes estão presentes apenas nos modelos UML, modelos PSM.

- Utilização da tecnologia XMI, favorecendo o processo de desenvolvimento proposto por se beneficiar das vantagens de XMI. Durante o processo, a transformação dos modelos é realizada através do formato XMI (MAS-ML XMI e UML XMI). O XMI é um padrão para a especificação e

transferência de modelos que estejam de acordo com MOF. Dentre as vantagens que o processo adquire por utilizar este padrão, podemos citar:

- estabelecimento de um acordo explícito para armazenamento e compartilhamento de objetos: os modelos MAS-ML e UML são armazenados no formato XMI para posterior utilização;
- busca de padronização que vem a permitir a interoperabilidade entre diversas ferramentas, devido à consistência e compatibilidade entre aplicações desenvolvidas com ferramentas, ambientes e fornecedores diversos: o UML XMI gerado durante o processo pode ser importado por outras ferramentas que tenham esta funcionalidade, com o objetivo de visualizar o modelo e gerar o código a partir do mesmo.

- O emprego da linguagem MAS-ML específica para SMA foi utilizada como linguagem de modelagem no processo de desenvolvimento proposto por atender as principais características desses sistemas e ser independente de plataforma;

- O framework ASF foi utilizado na transformação dos modelos MAS-ML em modelos UML específicos de plataforma, sendo baseado no mesmo framework conceitual de MAS-ML.

- Transformação de modelos em código → viabilização da transformação de modelos independentes de plataforma em código, ou seja, cobre todo o ciclo de desenvolvimento, desde a modelagem até a implementação.

2. Ferramenta que apóia o processo de desenvolvimento proposto, permitindo a utilização do processo de forma simples, através da modelagem gráfica em MAS-ML (primeira etapa do processo) e geração automática das demais etapas, aplicando fielmente as transformações existentes no processo. Além disso, é possível obter tanto o código gerado a partir da modelagem como também os artefatos gerados no decorrer do processo, como o MAS-ML XMI e UML XMI.

### **7.3. Trabalhos Futuros**

Alguns pontos importantes no processo de desenvolvimento proposto ficaram fora do escopo deste trabalho. Portanto, os seguintes trabalhos futuros da presente dissertação, incluem:

1. A utilização do diagrama dinâmico de MAS-ML no processo de desenvolvimento → Apenas os diagramas estruturais de MAS-ML foram utilizados durante o processo de desenvolvimento. Para que a geração de código considere também o comportamento das entidades existentes na aplicação é necessário considerar também o diagrama dinâmico presente em MAS-ML (diagrama de seqüência estendido) durante o processo de desenvolvimento.

2. Visualização gráfica na ferramenta dos modelos UML gerados a partir dos modelos MAS-ML → Incluir na ferramenta que dá suporte ao processo de desenvolvimento a visualização gráfica dos modelos UML gerados na segunda etapa do processo. Atualmente, estes modelos são gerados apenas no formato XMI.

3. Consistência entre modelos de diferentes níveis de abstração → Incluir na ferramenta que dá suporte ao processo de desenvolvimento a consistência entre modelos. Isto é, mudanças ocorridas nos modelos MAS-ML devem ser propagadas para os modelos UML e para o código. Contudo, nem todas as atualizações ocorridas nos modelos UML devem ser repassadas para os modelos MAS-ML, caso estejam relacionadas a detalhes de implementação, mas devem ser repassadas ao código. Além disto, nem todas as alterações no código da aplicação devem ser propagadas para os modelos UML e MAS-ML.

4. Flexibilidade para utilização de diferentes plataformas → Flexibilizar a ferramenta de forma a permitir a utilização de diferentes plataformas de acordo com o processo de desenvolvimento proposto.