

UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Diogo Otto Kunde

**INTEROPERABILIDADE ENTRE SIMULADORES CONSTRUTIVOS E  
VIRTUAIS TÁTICOS: UMA ABORDAGEM PARA EXTENSÃO DO  
SIMULATION OBJECT MODEL DO SIMULADOR MASA SWORD**

Santa Maria, RS  
2018

**Diogo Otto Kunde**

**INTEROPERABILIDADE ENTRE SIMULADORES CONSTRUTIVOS E VIRTUAIS  
TÁTICOS: UMA ABORDAGEM PARA EXTENSÃO DO SIMULATION OBJECT MODEL  
DO SIMULADOR MASA SWORD**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação, Área de Concentração em Ciência da Computação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Mestre em Ciência da Computação**.

ORIENTADOR: Dr. Prof. Raul Ceretta Nunes

Santa Maria, RS  
2018

Kunde, Diogo

Interoperabilidade entre simuladores construtivos e virtuais táticos: uma abordagem para extensão do Simulation Object Model do simulador MASA SWORD / Diogo Kunde.- 2018.

89 p.; 30 cm

Orientador: Raul Ceretta Nunes

Dissertação (mestrado) - Universidade Federal de Santa Maria, Centro de Tecnologia, Programa de Pós-Graduação em Ciência da Computação, RS, 2018

1. Simulação Distribuída 2. Simulação Virtual Tática 3. Simulação Construtiva 4. Modelagem de Multiresolução 5. High Level Architecture I. Ceretta Nunes, Raul II. Título.

**Diogo Otto Kunde**

**INTEROPERABILIDADE ENTRE SIMULADORES CONSTRUTIVOS E VIRTUAIS  
TÁTICOS: UMA ABORDAGEM PARA EXTENSÃO DO SIMULATION OBJECT MODEL  
DO SIMULADOR MASA SWORD**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação, Área de Concentração em Ciência da Computação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Mestre em Ciência da Computação**.

**Aprovado em 27 de fevereiro de 2018:**

---

**Raul Ceretta Nunes, Dr. (UFSM)**  
(Presidente/Orientador)

---

**Luís Alvaro de Lima Silva, Dr. (UFSM)**

---

**Edison Pignaton de Freitas, Dr. (UFRGS)**

Santa Maria, RS  
2018

## DEDICATÓRIA

*Dedico este trabalho primeiramente à minha família, em especial aos meus pais, Ângela Otto Kunde e Curt Kunde, que sempre me apoiaram e incentivaram durante todos esses anos de graduação e pós-graduação. Dedico este trabalho à minha filha, Manuela Devavane Abelin Kunde, que desde seu nascimento despertou-me uma motivação muito grande. Dedico este trabalho também a todos colegas que tive durante esses anos de muito estudo, especialmente aqueles com que pude dividir laboratório e trocar experiências diárias. Dedico este trabalho a todos amigos e amigas que estiveram presentes em minha vida para alegrá-la ou torná-la melhor. Por fim, dedico este trabalho ao professor Raul Ceretta Nunes pela orientação e pela atenção prestada durante todo o desenvolvimento do mestrado.*

## AGRADECIMENTOS

*Este trabalho é materialização de muito esforço, dedicação e é fruto da colaboração, apoio e incentivo de diversas pessoas e entidades. Em especial, agradeço:*

- Ao professor Dr. Raul Ceretta Nunes por ter me aceito como seu aluno de mestrado e ter prestado orientação e incentivo durante todo o curso;*
- À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo apoio financeiro durante a realização do Mestrado em Ciência da Computação;*
- À MASA Group por fornecer a licença do MASA SWORD e auxiliar na pesquisa desenvolvida através do seu simulador;*
- Ao projeto SISTEMA DE SIMULAÇÃO ASTROS 2020 - desenvolvido na Universidade Federal de Santa Maria em parceria com o Exército Brasileiro - pela oportunidade de desenvolvimento de pesquisa dentro do seu contexto;*
- Aos professores que tive durante toda a jornada acadêmica;*
- Aos colegas que tive no decorrer de minha formação acadêmica. Em especial aos colegas de laboratório com quem tive a oportunidade de trabalhar diariamente nesses dois anos de mestrado: Adriel Secco, Guilherme Miollo, Peter Prevedello, Ricardo Luiz Paul e Vanderlan Dupont de Oliveira;*
- Aos meus pais por prestar todo o apoio na minha formação. São meus maiores apoiadores e o meu alicerce;*
- Aos meus amigos que sempre estiveram por perto, tanto nos momentos bons quanto nos ruins. Em especial, agradeço a Raissa Devavane de Souza Abelin Amaral, mãe da minha filha, que cuidou da Manuela diariamente para que eu pudesse me dedicar e executar as atividades do mestrado.*

## RESUMO

### INTEROPERABILIDADE ENTRE SIMULADORES CONSTRUTIVOS E VIRTUAIS TÁTICOS: UMA ABORDAGEM PARA EXTENSÃO DO SIMULATION OBJECT MODEL DO SIMULADOR MASA SWORD

AUTOR: Diogo Otto Kunde  
ORIENTADOR: Dr. Raul Ceretta Nunes

Diferenças na abstração e fidelidade desejadas para o desenvolvimento de um sistema de simulação promovem modelos com propósitos e modelagens distintas, ou seja, diferenças de resolução. Dentro do campo de simulação distribuída, a interoperabilidade é característica que promove integração de sistemas diferentes em um único exercício. Tendo isso em vista, fazer com que dois simuladores de diferentes resoluções trabalhem juntos é uma tarefa não trivial e que apresenta uma série de desafios. Neste contexto, o MASA SWORD, que é um simulador construtivo agregado com abstrações de alto nível, utilizado principalmente para treinamento na tomada de decisões de alto nível, possui um *plug-in* HLA (*High Level Architecture*) afim de torná-lo um produto compatível para atuar como federado em diferentes federações HLA. Entretanto, este *plug-in* HLA foi projetado tendo em vista sua modelagem construtiva que abstrai diversos detalhes, ou seja, trata-se de um *plug-in* HLA com baixo nível de resolução. Desta maneira sua capacidade de interoperar limita-se a simuladores semelhantes, tornando a interoperabilidade com modelos de simulação de nível de resolução maior deficiente. Este trabalho apresenta uma solução, denominada SWAF (*SWORD Auxiliary Federate*), para contornar essa barreira e estender o SOM (*Simulation Object Model*) - conjunto de informações com que o simulador trabalha através de sua interface HLA - do simulador construtivo, permitindo que o mesmo interopere com um simulador virtual tático em uma federação HLA. Trata-se de um federado complementar que atua paralelamente ao *plug-in* HLA do MASA SWORD e fornece informações complementares sobre ele. Neste sentido, também é apresentada uma discussão para levantamento de requisitos de integração entre simulador construtivo e virtual tático, e um acordo de federação é modelado baseado nisso. O trabalho utiliza a metodologia de extensão indireta do SOM para contornar a barreira de software comercial do MASA SWORD e garantir a manutenção da generalidade do seu *plug-in* HLA original. Para validação do trabalho, um estudo de caso é desenvolvido com o Simulador Virtual Tático de Reconhecimento, Escolha e Ocupação de Posição (SVTat REOP).

**Palavras-chave:** Simulação Distribuída. High Level Architecture. MASA SWORD. Simulação Virtual Tática. Simulação Construtiva. Modelagem de Multiresolução.

## ABSTRACT

### INTEROPERABILITY BETWEEN CONSTRUCTIVE AND VIRTUAL TACTICAL SIMULATORS: AN APPROACH TO EXTENSION OF THE SIMULATION OBJECT MODEL OF MASA SWORD SIMULATOR

AUTHOR: Diogo Otto Kunde  
ADVISOR: Raul Ceretta Nunes

Differences in the desired fidelity and abstraction for the development of simulation systems provide models with distinct purposes and modelling, i.e. resolution discrepancies. In the distributed simulation field, interoperability is a feature that enables the linkage of different systems in an integrated exercise. The development of this linkage is a non-trivial task and presents many challenges. In this context, the MASA SWORD, which is an aggregated constructive simulator with high-level abstraction and mainly used to strategic training, has a HLA plug-in to enable the simulator to participate in a HLA federation as a federate. However, this HLA plug-in was developed based on your constructive model that abstracts many details and works with a small set of information. This characteristic makes it unable for integration with simulator that require a higher information level. This work presents a solution for this problem, called SWAF (SWORD Auxiliary Federate), which extends the SOM (Simulation Object Model) - information set with which the simulator works through its HLA interface - of the constructive simulator to interoperate with a virtual tactical simulator in a HLA federation. The SWAF allows the publication of additional information about MASA SWORD through an auxiliary federate that works in parallel with the original HLA plug-in of SWORD. A discussion about interoperability is also presented and used to define requirements specification that are modeled in a new federation agreement. The work uses an indirect SOM extension methodology to overcome COTS (Commercial off-the-shelf) barrier and kept intact the original HLA interface of SWORD simulator. To validate the work, a study case was developed with Simulador Virtual Tático de Reconhecimento, Escolha e Ocupação de Posição (SVTat REOP).

**Keywords:** Distributed Simulation. High Level Architecture. MASA SWORD. Virtual Tactical Simulation. Constructive Simulation. Multiresolution Modeling.

## LISTA DE FIGURAS

Figura 2.1 – Comunicação entre federado e RTI .....	19
Figura 2.2 – Estrutura do módulo de classes de entidades físicas do RPR FOM .....	23
Figura 2.3 – Estrutura dos módulos NETN FOM .....	24
Figura 2.4 – Interface do cliente de <i>gaming</i> do MASA SWORD .....	25
Figura 2.5 – Exemplo da constituição hierárquica dos exercícios do MASA SWORD ..	26
Figura 2.6 – Arquitetura de rede do MASA SWORD .....	27
Figura 2.7 – Composição da mensagem enviada pelo MASA SWORD .....	28
Figura 2.8 – Tipos de <i>tags</i> de mensagens utilizadas pelo MASA SWORD .....	28
Figura 2.9 – Interface de configuração do plug-in HLA do MASA SWORD .....	29

## LISTA DE QUADROS

Quadro 2.1 – Classes de objetos do SOM do MASA SWORD .....	30
Quadro 2.2 – Classes de interações do SOM do MASA SWORD .....	31
Quadro 2.3 – Classes de objetos e interações do SOM do MASA SWORD ao utilizar NETN .....	31
Quadro 4.1 – Classes de objetos do SOM atualizado do MASA SWORD .....	39
Quadro 4.2 – Classes de interações do SOM atualizado do MASA SWORD .....	39

## LISTA DE ABREVIATURAS E SIGLAS

<i>API</i>	Application Programming Interface
<i>COTS</i>	Commercial Of-The-Shelf
<i>CRC</i>	Central RTI Component
<i>DIS</i>	Distributed Interactive Simulation
<i>DLL</i>	Dynamic-Link Library
<i>DoD</i>	Department of Defense
<i>FOM</i>	Federation Object Model
<i>HLA</i>	High Level Architecture
<i>IEEE</i>	Instituto dos Engenheiros Eletricistas e Eletrônicos
<i>LRC</i>	Local RTI Component
<i>LVC</i>	Live, Virtual and Constructive
<i>M&amp;S</i>	Modeling and Simulation
<i>MRM</i>	Multi-Resolution Modelling
<i>NETN</i>	North Atlantic Treaty Organization
<i>OMT</i>	Object Model Template
<i>PDU</i>	Protocol Data Unit
<i>RPR</i>	Real-Time Platform Reference
<i>RTI</i>	Run-Time Infrastructure
<i>SISO</i>	Simulation Interoperation Standards Organization
<i>SOM</i>	Simulation Object Model
<i>SVTatREOP</i>	Simulador Virtual Tático de Reconhecimento, Escolha e Ocupação de Posição
<i>XML</i>	eXtensible Markup Language

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
1.1	OBJETIVOS	12
1.2	CONTRIBUIÇÃO E JUSTIFICATIVA	13
1.3	ORGANIZAÇÃO DO TRABALHO	15
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>16</b>
2.1	SIMULAÇÃO E LVC	16
2.2	SIMULAÇÃO DISTRIBUÍDA E INTEROPERABILIDADE	18
2.3	<i>HIGH LEVEL ARCHITECTURE</i>	18
<b>2.3.1</b>	<b>Grupos de serviços</b>	<b>19</b>
<b>2.3.2</b>	<b>Regras</b>	<b>21</b>
<b>2.3.3</b>	<b>FOM</b>	<b>21</b>
<b>2.3.4</b>	<b>FOMs de Referência</b>	<b>22</b>
2.3.4.1	<i>RPR FOM</i>	22
2.3.4.2	<i>NETN FOM</i>	23
2.4	MODELAGEM DE MULTIRESOLUÇÃO	24
2.5	MASA SWORD	25
<b>2.5.1</b>	<b>Arquitetura de rede do MASA SWORD</b>	<b>27</b>
<b>2.5.2</b>	<b>Plug-in HLA</b>	<b>29</b>
2.6	TRABALHOS RELACIONADOS	32
<b>2.6.1</b>	<b>Multiresolução em simulação</b>	<b>32</b>
<b>2.6.2</b>	<b>Federação SIS-ASTROS</b>	<b>35</b>
<b>3</b>	<b>REQUISITOS PARA INTEROPERABILIDADE ENTRE SIMULADOR CONSTRUTIVO E VIRTUAL TÁTICO</b>	<b>36</b>
3.1	REQUISITOS PARA COMPLEMENTAÇÃO DO EXERCÍCIO	36
3.2	NOVO MÓDULO FOM	36
<b>4</b>	<b>PROJETO DE EXTENSÃO INDIRETA DO SOM DO SIMULADOR MASA SWORD</b>	<b>37</b>
4.1	OBJETIVOS TÉCNICOS DA SOLUÇÃO	37
4.2	ARQUITETURA E FUNCIONAMENTO	38
<b>4.2.1</b>	<b>SOM atualizado</b>	<b>38</b>
<b>5</b>	<b>VALIDAÇÃO E RESULTADOS</b>	<b>41</b>
5.1	FEDERAÇÃO SIS-ASTROS	41
5.2	ESTUDO DE CASO	41
5.3	AVALIAÇÃO DA SOLUÇÃO DESENVOLVIDA	42
<b>5.3.1</b>	<b>Prós da Metodologia de extensão indireta do SOM</b>	<b>43</b>
<b>5.3.2</b>	<b>Contras da Metodologia de extensão indireta do SOM</b>	<b>44</b>
<b>6</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>46</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>48</b>

## 1 INTRODUÇÃO

Segundo DoD (1998), os sistemas de simulação podem ser classificados em três categorias distintas de acordo com sua modelagem e nível de interação humana. Essa classificação é conhecida como LVC, ou *Live, Virtual e Constructive* (vivo, virtual e construtivo): a viva envolve pessoas reais operando sistemas reais; na virtual, pessoas reais operam sistemas simulados que são projetados para inseri-las no *human-in-the-loop* (modelo que requer interação humana) para exercitar diversas habilidades específicas; o modelo construtivo envolve um ambiente totalmente sintético onde pessoas simuladas operam sistemas simulados e pessoas reais produzem estímulos (entradas) para o sistema, que determina os resultados sem envolvimento humano.

A simulação distribuída se tornou uma área de estudo muito importante, principalmente no contexto militar, onde é muito utilizada para treinamento de oficiais, diminuindo custos e riscos (NICOL, 2011). A simulação distribuída evoluiu conforme a tecnologia em redes de computadores avançou, permitindo a utilização de redes locais ou de WAN (*Wide Area Network*) para execução de um treinamento utilizando computadores distribuídos geograficamente (FUJIMOTO, 2000). Dentro desta área encontra-se outro tópico de grande relevância, a interoperabilidade entre simuladores, que é uma característica que permite integrar diferentes sistemas através de um meio de comunicação. Atualmente a interoperabilidade é promovida através de arquiteturas, como é o caso da arquitetura de alto nível ou *High Level Architecture* (HLA) (IEEE, 2010b). Nesta arquitetura os simuladores correspondem a federados que fazem parte de uma federação. A federação possui um acordo de federação (*Federation Object Model* - FOM) que regula a troca de informações entre os federados, e cada um deles define através de seu *Simulation Object Model* (SOM) as classes de objetos e interações com que irá trabalhar (IEEE, 2010c).

Em um exercício integrado podem existir diferentes categorias de simuladores com níveis distintos de granularidade de informação. Um exemplo clássico é a integração entre simuladores virtuais e construtivos (STOBER et al., 1995). Exercícios envolvendo multi-resolução, como estes são conhecidos, exigem uma sincronização especial, visto que as informações enviadas pelo remetente devem chegar ao destinatário no formato em que este possa refleti-la conforme sua modelagem. Em geral, a arquitetura HLA não resolve tais problemas, cabendo aos desenvolvedores elaborarem suas próprias soluções (TOLK, 2012).

Simuladores tem níveis e granularidade de informações diferentes por causa da abstração e fidelidade desejada (REYNOLDS; NATRAJAN; SRINIVASAN, 1997). A resolução pode variar em diferentes dimensões: objetos, escala temporal, terreno, sistema e processo (RABELO et al., 2015). A dimensão relacionada a objetos está ligada diretamente com os níveis de unidade (agregado) e entidade (desagregado). Embora existam

exceções, construtivos são geralmente de nível de unidade e modelos virtuais são de nível de entidade. A maioria dos trabalhos no campo da multiresolução em simulação trata aspectos relacionados a objetos, terrenos e escala de tempo (TOLK, 2012). Embora LVC refira-se aos três tipos, este trabalho é focado em modelos construtivos e virtuais táticos no domínio militar de simulação.

O principal motivo para integrar modelos com diferentes granularidades é a utilização dos pontos fortes de cada simulador em um exercício único para alcançar certos objetivos (FRANCESCHINI; PETTY, 1995). Para Stober et al. (1995), essa integração entre modelos deve ser desenvolvida de modo que um simulador possa influenciar o outro ou afetá-lo. Existem vários benefícios na promoção dessa interoperabilidade, como um melhor ambiente de treinamento e capacidades de visualização, além do reuso e treino multinível para operações combinadas (CAYIRCI, 2009). Franceschini e Petty (1995) atribuem os ganhos que ocorrem tanto para quem se utiliza do ambiente construtivo quanto para a comunidade de treinamento virtual como os principais motivos para integração. Nesse caso, o construtivo pode obter um nível mais detalhado de informações vinda do cenário virtual, e o modelo virtual pode conduzir pequenos treinamentos dentro de um cenário construtivo muito maior, adicionando realismo e motivação ao treino. Além disso, Franceschini e Petty (1995) destacam que o treinamento pode ser enriquecido ao se fornecer informações adicionais entre os diferentes níveis de resolução.

Nesse contexto, há uma série de dificuldades em realizar a integração entre simuladores de níveis de resolução diferentes. O simulador MASA SWORD é um simulador construtivo, agregado e automatizado para treinar as habilidades de tomada de decisão de equipes de comando de brigadas e divisões, permitindo a imersão em simulações de conflitos, operações militares e controle e prevenção de desastres (MASA GROUP, 2017). Ele é um simulador comercial amplamente utilizado no treinamento militar, sendo inclusive adotado pelo Exército Brasileiro. O simulador possui um *plug-in* HLA afim de realizar integração com outros simuladores. Embora esse *plug-in* cubra o essencial para realização do exercício simulado entre simuladores construtivos, ele é limitado a resolução em que atua e informações relevantes para simuladores de nível de granularidade mais fina não são publicadas. Na tentativa de integração com um simulador virtual tático pôde-se notar a existência de uma carência de informações, onde há uma necessidade maior de informações do que ele poderia fornecer através da sua interface HLA. Portanto o problema tratado neste trabalho é a necessidade destas informações não exploradas pelo simulador MASA SWORD para interoperabilidade com simuladores que exigem um nível de detalhamento maior em um ambiente multiresolução interoperável, onde existe a integração construtiva e virtual tática.

O MASA SWORD disponibiliza (MASA, 2015a) uma interface de comunicação onde os dados são serializados a partir do uso do mecanismo de serialização estruturada de dados conhecido como Protobuf (*Protocol Buffers*) (GOOGLE, 2017), e também armazena

diversas configurações de exercício em arquivos XML. Através desses meios é possível obter uma série de informações que podem ser utilizadas para diversos fins, inclusive estender suas funcionalidades em relação a sua interface HLA. Sendo assim, este trabalho buscou solucionar a necessidade de informações por parte de um simulador de nível de resolução mais alto em contraste com as que são fornecidas pelo MASA SWORD através da sua interface HLA, a partir do desenvolvimento de um *plug-in* adicional que cubra essa necessidade.

Para avaliar as necessidades de interoperabilidade na construção da solução foi realizada uma discussão para levantamento de requisitos em uma integração envolvendo o simulador construtivo MASA SWORD e um simulador virtual tático, que é um simulador que mescla características virtuais e construtivas em seu modelo. Essa discussão, assim como os requisitos extraídos, levou em consideração elementos do modelo de simulação construtivo que poderiam ser representados no modelo virtual, ou seja, o foco do trabalho foi no sentido de transferir informações do ambiente construtivo para o virtual, identificando elementos com semântica para que o modelo virtual tático pudesse utilizar em seu ambiente de treinamento. Nesse sentido, para complementação do exercício foi preciso identificar tipos de informações para complementação do exercício com objetivo de vencer essa barreira de multiresolução.

A solução desenvolvida foi nomeada como SWAF (*SWORD Auxiliary Federate*) e foi construída através de uma metodologia diferente, a metodologia de extensão indireta do SOM. Essa metodologia faz com que seja introduzida uma interface adicional para o simulador, onde o federado complementar passa a atuar como um segundo federado transmitindo informações sobre este simulador na federação HLA. Não se trata de uma interface que modifica ou substitui o *plug-in* HLA original do simulador construtivo, mas sim um complemento para atuar paralelamente com esta, que tem a função de obter informações internas do simulador construtivo e criar semântica para que elas possam ser utilizadas na integração com outros simuladores. A utilização da metodologia de extensão indireta possibilita estender o SOM do simulador conforme os requisitos desejados e no caso deste trabalho foi utilizada para quebrar a barreira de diferenças de resolução entre um modelo de simulação construtivo e um virtual tático.

A versão do MASA SWORD utilizada neste trabalho é a 6.2, e para fins de validação da solução proposta foi elaborado um estudo de caso envolvendo o Simulador Virtual Tático de Reconhecimento, Escolha e Ocupação de Posição (SVTat REOP), integrante da Federação SIS-ASTROS (Sistema de Simulação Integrada ASTROS).

## 1.1 OBJETIVOS

O objetivo geral do trabalho foi estender as funcionalidades em relação à interface HLA do simulador MASA SWORD afim de tornar o seu SOM mais abrangente para integra-

ção com simuladores que necessitam de um nível de informações mais apurado, focando especialmente em simuladores virtuais com características táticas. Os objetivos específicos são listados abaixo:

- Avaliação e definição dos requisitos para interoperabilidade entre simulador construtivo e virtual tático. Nesse contexto é realizada uma discussão que enumera diversos requisitos para desenvolvimento dessa interoperabilidade e modela um acordo de federação baseado em tais requisitos;
- Projeção da solução que habilite o simulador construtivo MASA SWORD atuar em conjunto com simuladores que necessitem de um nível de informações mais elevado, levando-se em consideração os requisitos obtidos para interoperar com simuladores virtuais táticos;
- Desenvolvimento e validação da solução com um estudo de caso entre o simulador construtivo MASA SWORD e um simulador virtual tático. Para isso, a solução desenvolvida foi integrada a Federação SIS-ASTROS, que conta o Simulador Virtual Tático de REOP.

## 1.2 CONTRIBUIÇÃO E JUSTIFICATIVA

O trabalho apresenta uma solução para complementação de informações na integração entre diferentes níveis de simulação envolvendo o simulador construtivo MASA SWORD. A contribuição principal é o novo *plug-in* HLA desenvolvido para extensão do SOM do SWORD. Ele estende as capacidades de interoperabilidade do simulador e abrange as necessidades de integração com simuladores de maior nível de resolução. Os requisitos foram levantados através da discussão de interoperabilidade entre o MASA SWORD e um simulador virtual tático, e o estudo de caso foi desenvolvido no contexto da Federação SIS-ASTROS. A solução pode ser utilizada por qualquer federação que necessite um nível de informações mais elevado, mas é voltada para o contexto de interoperabilidade com modelos virtuais táticos.

Se tratando de um contexto militar, a solução possibilita um treinamento multinível através da interação entre o modelo construtivo e o virtual tático, e habilita um treinamento multi-hierárquico, que faz com que oficiais de alto nível possam treinar suas habilidades estratégicas no simulador construtivo e oficiais de um nível não tão alto possam se aperfeiçoar no cenário virtual tático, de forma integrada. Ao realizar a implementação do novo *plug-in*, buscou-se identificar conceitos importantes para simuladores virtuais táticos em contraste com as informações que estão disponíveis no modelo construtivo. A solução resolve problemas na demanda de informações, fornecendo meios para complementação do exercício multiresolução através das seguintes contribuições:

- Representação situacional apurada entre os simuladores: modelagem de abstrações internas do simulador construtivo que podem ser transmitidas para outros simuladores, para que estes possam refletir tais abstrações a partir de sua modelagem, criando a representatividade de elementos importantes no contexto virtual tático e, dessa forma, promovendo um ambiente integrado com maior nível de detalhes. É alcançada com a publicação de objetos não adotados no *plug-in* HLA original do MASA SWORD, como os desenhos de linhas táticas e a relação entre agregados, por exemplo.
- Transmissão de ordens entre os modelos: proporcionada através da passagem de interações que dizem respeito a ordens dadas no construtivo e que podem ser utilizadas por outros simuladores para os mais diversos fins. Essa contribuição fornece informações sobre atividades realizadas no simulador construtivo e, além disso, proporciona uma melhoria nas capacidades de visualização do simulador integrado ao SWORD, visto que esse pode criar representação para ações remotas vindas do construtivo. Em um simulador virtual, por exemplo, eventos podem ser criados a partir de ordens, gerando ações para visualização de efeitos em três dimensões das mesmas;
- Sincronia entre os modelos: são criadas interações para manter os modelos integrados consistentemente em relação a dimensão temporal. Essas interações fornecem informações temporais precisas e em tempo de execução;

Além de produzir contribuições para o exercício nos itens destacados acima, o trabalho se diferencia pelo método utilizado, adotando uma extensão indireta do SOM do simulador construtivo. O uso dessa metodologia habilita adicionar quaisquer novas funcionalidades ao SOM do federado de maneira mais simples, sem necessidade de alterar o código-fonte interno do simulador. Essa abordagem mantém a generalidade da interface HLA original do SWORD, e quando um maior nível de informações sobre o modelo construtivo se torna necessário para atuar na federação, a solução pode ser utilizada para suprimir essa demanda.

O desenvolvimento da federação SIS-ASTROS é um dos esforços pioneiros no Exército Brasileiro pela realização de exercícios integrados que envolvem diferentes níveis LVC. A construção de novos modelos de simulação, como é o caso do Simulador Virtual Tático de REOP, e de ferramentas para interoperabilidade, como a solução desenvolvida neste trabalho, representam esforços para inverter essa dependência e gerar tecnologia nacional para defesa na área de simulação de multiresolução. Portanto, este trabalho justifica-se tanto pelo estudo realizado no campo de multiresolução e integração entre modelos diferentes granularidades, quanto no sentido de desenvolvimento de tecnologia para a área de defesa.

### 1.3 ORGANIZAÇÃO DO TRABALHO

O restante do trabalho está organizado da seguinte forma: no Capítulo 2 é apresentado um breve referencial teórico sobre os principais tópicos abordados na dissertação, como LVC, simulação distribuída, HLA, modelagem de multiresolução, aspectos importantes de funcionamento do MASA SWORD, e, além disso, o capítulo traz uma seção de trabalhos relacionados dentro do tópico de multiresolução e do ambiente da Federação SIS-ASTROS; o Capítulo 3 é utilizado para especificação de requisitos no desenvolvimento do SWAF, trazendo inicialmente uma discussão sobre levantamento das necessidades em relação a interoperabilidade entre simuladores construtivos e virtuais táticos, e, após isso, apresenta aspectos de modelagem do novo acordo de federação baseado nesse requisitos; o Capítulo 4 é responsável por descrever o projeto do *plug-in* HLA complementar desenvolvido com base no novo módulo FOM, detalhando cada módulo da solução; no Capítulo 5 é apresentada a validação do trabalho e os resultados alcançados a partir da utilização do novo *plug-in* complementar atuando sobre Federação SIS-ASTROS; por fim, o Capítulo 6 traz as considerações finais e apresenta trabalhos futuros que podem ser desenvolvidos dentro deste contexto.

## 2 REFERENCIAL TEÓRICO

Este capítulo é destinado a apresentar uma fundamentação teórica necessária para embasamento científico do trabalho. A Seção 2.1 traz um breve resumo da categorização LVC de modelos de simulação. Um breve contexto sobre simulação distribuída e interoperabilidade é apresentado na Seção 2.2. Na Seção 2.3 são detalhadas questões normativas e técnicas da HLA, fornecendo também uma visão geral sobre acordos de federação. A Seção 2.4 traz uma breve explicação sobre modelagem de multiresolução dentro do campo de simulação. A Seção 2.5 aborda, principalmente, questões da arquitetura de rede e interface HLA do simulador MASA SWORD, além de outros aspectos significantes para o desenvolvimento trabalho. Por fim, a Seção 2.6 discute sobre os principais trabalhos relacionados à construção de federação multiresolução e desenvolvimento da Federação SIS-ASTROS.

### 2.1 SIMULAÇÃO E LVC

Simulações são modelos de representação da realidade, a partir de certas características-chave ou estado de um sistema abstrato (NICOL, 2011). A simulação é utilizada principalmente para treinamento e os principais motivos para sua adoção são a redução dos custos, aumento da segurança, melhor visibilidade dos eventos e reprodutibilidade das ações (FLETCHER, 2009).

O DoD (1998) classifica os sistemas de simulação em três níveis de acordo com sua modelagem e nível de interação humana e essa classificação é conhecida como LVC (*Live, Virtual and Constructive*). A primeira categoria é a viva (*Live*), que tratam-se de simulações onde pessoas reais operam sistemas reais. Este tipo de simulação é muito utilizado para o fornecimento de treinamento realístico que permite ao indivíduo treinado interagir com o ambiente e equipamentos reais ou muito próximos da realidade. Uma partida de *paintball* é um exemplo de simulação viva, onde pessoas utilizam um marcador de ar comprimido - ou equipamento semelhante - para atirar bolas de tinta em seus inimigos e derrotá-los, ou seja, trata-se de uma simulação viva de batalha.

O segundo nível é o virtual, que segundo o DoD (1998) é representado por simuladores em que pessoas reais operam sistemas simulados e este sistema é projetado para inserir quem está sendo treinado no *human-in-the-loop* (homem no *loop*). As simulações virtuais são utilizadas para fornecer treinamento de habilidades específicas de maneira com que o sistema seja virtualizado e o humano possa interagir com ele na execução de ações dentro do mundo virtual (FLETCHER, 2009). Exemplos de simulação virtual são os softwares de simuladores de voo, que emulam o ambiente de uma cabine de aeronave e inserem o humano nele para que sua habilidade de controlar e utilizar os equipamentos

disponíveis sejam treinadas. Isso permite ao indivíduo treinado praticar sem correr riscos ou desperdiçar recursos.

Por fim, o terceiro nível trata-se do construtivo. Esse tipo de simulação engloba um ambiente sintético em que pessoas simuladas operam sistemas simulados e pessoas reais produzem entradas para o sistema (DOD, 1998). A determinação dos resultados é realizada pela inteligência artificial e algoritmos do simulador construtivo sem o envolvimento humano. A simulação construtiva é utilizada para treinar aspectos de decisão estratégica, ou seja, preocupa-se com as características de alto nível, abstraindo pequenos detalhes do seu modelo (TOLK, 2012). Exemplos de simuladores construtivos são os *wargames*, que são utilizados para preparação de estratégias em batalhas, guerras, etc.

Como mencionado, cada categoria de simulação possui seus objetivos próprios e os requisitos de desenvolvimento determinam a sua categorização. Seguindo isso, existem simuladores que mesclam características de diferentes categorias. Um caso disso é o simulador virtual tático, que se trata da união de elementos construtivos com elementos virtuais para construção de um modelo único. Um simulador deste tipo é habilitado a fornecer treinamento de nível virtual em conjunto com a parte decisional tática. O fato de conseguir unir duas categorias em um modelo deve-se a flexibilidade da simulação virtual, que pode modelar tanto aspectos de categoria viva quanto da construtiva. Adoção de modelos virtuais táticos deve-se ao fato de que o custo computacional para construção de um cenário virtual com vasto número de entidades é muito elevado para a execução de um treinamento tático. Logo uma simulação virtual de alto nível torna-se útil para o treinamento de pequenas guarnições com um número não tão elevado de entidades (FERREIRA et al., 2017).

Um exemplo desse tipo de simulador é o SVTat REOP (utilizado posteriormente no estudo de caso deste trabalho), que emprega a parte tática para treinamento de oficiais de alto nível no campo de artilharia baseando-se em doutrinas militares para efetivar ações com viaturas ASTROS (*Artillery SaTuration Rocket System*, ou Sistema de Foguetes de Artilharia para Saturação de Área), e, além disso, dispõe de alto detalhamento de terreno, de maneira que se deva escolher a melhor localização para disparo de munição, além de outras características que remetam a modelos de simulação virtual. Sendo assim, a união de elementos virtuais com elementos construtivos promove o desenvolvimento de um simulador virtual com características táticas.

A principal diferença entre construtivo e virtual tático é que o primeiro preocupa-se com questões estratégicas, enquanto o segundo conta com treinamento voltado para parte tática. Estratégia é um conceito bastante amplo que preocupa-se com a organização como um todo e envolve planejamento de alto nível voltado para um prazo mais longo. Por outro lado, questões táticas se preocupam com a execução das ações em um prazo mais curto, ou seja, trata-se da implementação da estratégia na execução das ações para alcançar seu planejamento.

## 2.2 SIMULAÇÃO DISTRIBUÍDA E INTEROPERABILIDADE

Fujimoto (2000) define simulação distribuída como sendo uma tecnologia que habilita um programa de simulação ser executado em um sistema contendo múltiplos processadores através de uma rede de comunicação. As máquinas desse ambiente podem estar fisicamente divididas em ambientes próximos, como uma sala ou um prédio, ou espalhadas geograficamente ao redor do mundo. Fujimoto (2000) ainda destaca que os quatro principais benefícios da simulação distribuída são a redução de tempo de execução, a distribuição geográfica, a integração de máquinas de diferentes fabricantes e a tolerância a falhas.

O desenvolvimento da simulação distribuída começou a ocorrer por volta de 1983, nos Estados Unidos, com o desenvolvimento do SIMNET (*SIMulator NETworking*) pela DARPA (*Defense Advanced Research Projects Agency*) (NICOL, 2011). Com o avanço no campo, surgiram necessidades de integração entre diferentes produtos. A interoperabilidade entre simuladores busca fazer com que dois ou mais produtos (simuladores) diferentes trabalhem juntos de forma integrada, trocando mensagens e interagindo entre si. Para isso, existem algumas arquiteturas que padronizam essa comunicação entre os diferentes sistemas.

A primeira arquitetura a ter destaque neste campo foi a DIS (*Distributed Interactive Simulation*), que é uma arquitetura *peer-to-peer* que liga simulações de nível de plataforma em tempo real, sendo basicamente um protocolo que permite que os simuladores se comuniquem uns com os outros (NICOL, 2011). Conforme a IEEE (2012), os conceitos de arquitetura básica são uma extensão do SIMNET. O padrão especifica o layout das mensagens de dados, chamados PDU (*Protocol Data Units*), que são organizados dentro de domínios conhecidos como família de protocolos, que incluem *Entity Information/Interaction, Warfare, Logistics, Simulation Management*, etc. Essas mensagens são trocadas entre os *hosts* da simulação através da rede.

## 2.3 HIGH LEVEL ARCHITECTURE

Segundo IEEE (2010b), HLA é uma arquitetura técnica desenvolvida para interoperar entre sistemas de simulação distribuídos, conhecidos como federados. Essa arquitetura fornece um *framework* para que os desenvolvedores possam estruturar suas aplicações e descrever seus sistemas de simulação. A base para o funcionamento dela é o RTI (*Runtime Infrastructure*), que é um *middleware* que fornece um conjunto de serviços, definidos pela especificação da interface em IEEE (2010b), utilizados pelos federados para coordenar as operações e troca de dados durante o tempo de execução da federação (conjunto de dois ou mais federados atuando em um exercício). Sua estrutura é formada por 3 componentes principais:

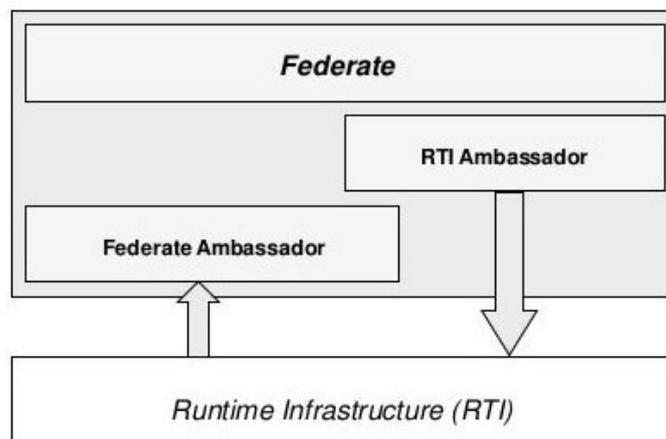
- *HLA Framework and Rules Especification*: fornece uma visão geral da HLA, define a família de documentos da arquitetura e apresenta as dez regras da mesma. Cinco destas regras são para os federados, e outras cinco para a federação (IEEE, 2010b).
- *HLA Federate Interface Especification*: especifica os tipos de serviços que permitem aos simuladores se conectar, trocar dados e coordenar atividades durante uma simulação distribuída. Além disso, descreve as capacidades e princípios do funcionamento do RTI (IEEE, 2010a).
- *HLA Object Model Template*: a HLA OMT fornece uma especificação para descrever os modelos de objetos, interações, atributos e parâmetros que definem a informação produzida ou requerida pelas aplicações de simulação (IEEE, 2010c).

As próximas subseções se referem a conceitos do HLA que são importantes no desenvolvimento do trabalho. A Subseção 2.3.1 resume os grupos de serviços existentes na HLA, dando uma breve descrição de cada um. A Subseção 2.3.2 enumera as regras da arquitetura. Por fim, a Subseção 2.3.3 traz uma breve explicação sobre acordos de federação (FOM) e sua formação, além de apresentar dois FOMs de referência utilizados durante estudo e desenvolvimento da dissertação.

### 2.3.1 Grupos de serviços

Como mencionado, a comunicação entre um federado e os demais membros da federação ocorre através do RTI. Para isso, a HLA fornece uma padronização para API de programação, onde temos as duas principais classes: *RTI Ambassador* e *Federate Ambassador* (Figura 2.1). É importante lembrar que a normatização apenas estabelece padrões a serem seguidos, a implementação dessas APIs de programação é de responsabilidade de seus desenvolvedores, assim como as implementações de RTIs.

Figura 2.1 – Comunicação entre federado e RTI



Fonte: (D'AMBROGIO; GIANNI; IAZEOLLA, 2007) (slides presentation).

Nesse paradigma de embaixadores, o RTI *Ambassador* agrupa os serviços prestados pelo RTI, onde todas as solicitações feitas do federado para o RTI assumem a função *calls* (chamadas). Por outro lado, a classe abstrata *Federate Ambassador* identifica as funções de retorno, *callbacks*. Neste contexto, cada embaixador implementa seus métodos agrupados em grupos de serviços (IEEE, 2010a). Existem sete grupos de serviços e cada um deles é resumido a seguir:

- *Federation Management*: permite coordenar as atividades a nível de federação;
- *Declaration Management*: permite aos federados especificar os tipos de dados que eles fornecerão ou receberão da federação em execução;
- *Object Management*: apoia as atividades do ciclo de vida dos objetos e interações usadas pelos federados durante a execução da federação;
- *Ownership Management*: é utilizado para estabelecer propriedades específicas para os federados membros de uma federação, para o fornecimento de valores para um atributo de instância de objeto, bem como facilitar a transferência dinâmica dessa propriedade.
- *Time Management*: permite aos federados trabalhar com o conceito de tempo para operar com um conceito lógico de relógio virtual distribuído. Suporta simulações de eventos discretos e garantia de ordenação causal entre os eventos. As políticas de gerenciamento de tempo são *regulating* e *constrained*. Um federado *regulating* pode gerar mensagens ordenadas através do *Timestamp* (*Timestamp order* ou TSO), enquanto que um federado *constrained* pode receber mensagens TSO. Um federado pode ser tanto *constrained* ou *regulating*, quanto *constrained* e *regulating*, ou nenhum dos dois.
- *Data Distribution Management* (DDM): permite aos federados especificar condições de distribuição, além dos fornecidos pelo *Declaration Management*, para especificar os dados enviados e recebidos. Os serviços prestados pelo DDM podem reduzir a transmissão e recepção de dados irrelevantes, reduzindo a largura de banda necessária para troca de dados, por exemplo.
- *Support Services*: serviços variados utilizados pelos federados. Um exemplo de método disponível através do serviço de suporte é a obtenção do *handle* (espécie de identificador utilizado na HLA para manusear determinado atributo, por exemplo) de algum atributo de um objeto, por exemplo.

### 2.3.2 Regras

As regras presentes em IEEE (2010b) garantem a interação adequada entre federados em uma federação e definem responsabilidades dos componentes para que os modelos possam interoperar de forma adequada. As cinco primeiras regras referem-se à federação e as cinco posteriores aos federados. Observe as regras a seguir:

1. Federações devem ter um FOM HLA documentado de acordo com a HLA OMT;
2. Em uma federação, todas instâncias de objetos associados a simulação devem estar nos federados e não no RTI;
3. Durante a execução de uma federação, toda troca de dados FOM entre federados deve ocorrer via RTI;
4. Durante a execução de uma federação, os federados devem interagir com o RTI de acordo com as especificações de interface HLA;
5. Durante a execução de uma federação, um atributo de uma instância deve ser propriedade de no máximo um federado em qualquer momento;
6. Federados devem ter um SOM HLA documentado de acordo com a HLA OMT;
7. Federados devem ser capazes de atualizar e/ou refletir qualquer atributo de instância e enviar e/ou receber interações conforme especificado em seu SOM;
8. Federados devem estar aptos a transferir e/ou aceitar a posse de atributos dinamicamente durante a execução da federação conforme especificado em seu SOM;
9. Federados devem estar habilitados a variar as condições sobre as quais fornecem informações de atributos, conforme especificado em seu SOM;
10. Federados devem ser capazes de administrar os tempos locais de uma forma que lhes permita coordenar a troca de dados com os outros membros da federação.

### 2.3.3 FOM

Toda troca de dados relacionada a um exercício deve ser especificada no FOM (*Federation Object Model*) do exercício, ou seja, o FOM determina todas as informações que poderão ser utilizadas durante a execução da federação, e este acordo deve ser especificado antes da simulação ser desenvolvida. Modelos de objetos HLA são compostos de grupos de componentes inter-relacionados especificando informações sobre as classes de objetos e seus atributos, e as interações e seus parâmetros. O OMT define os vários componentes de um FOM, e os principais são (IEEE, 2010c):

- Tabela de Classes de objetos: coleção de objetos com certas características e atributos em comum. São definidas de maneira hierárquica, podendo ter vários níveis de especialização através de subclasses;
- Tabela de Classes de interação: uma interação é definida como uma ação tomada por um federado que pode afetar ou impactar outro federado. Classes de interação também podem conter especializações;
- Tabela de atributos: especifica as características dos objetos;
- Tabela de parâmetros: especifica as características das interações;
- Tabela de tipos de dados: especifica detalhes da representação de tipos de dados.

O FOM é especificado em um arquivo XML seguindo o esquema definido através da OMT (IEEE, 2010c). Ciente do FOM, para que um federado passe a publicar informações sobre determinado objeto ou interação, ele precisa demonstrar seu interesse através da operação de *publish* (publicar) sobre mesmo. Da mesma forma, para que ele receba informações sobre estes, ele precisa demonstrar interesse através do *subscribe* (inscrever/assinar). Com base nesse interesse (*publish/subscribe*) que o SOM é definido. O *Simulation Object Model* trata das informações de cada federado em relação a federação, isto é, o FOM especifica o conjunto geral das informações que podem ser trocadas dentro da execução da federação, e o SOM especifica as informações referentes a cada federado.

### 2.3.4 FOMs de Referência

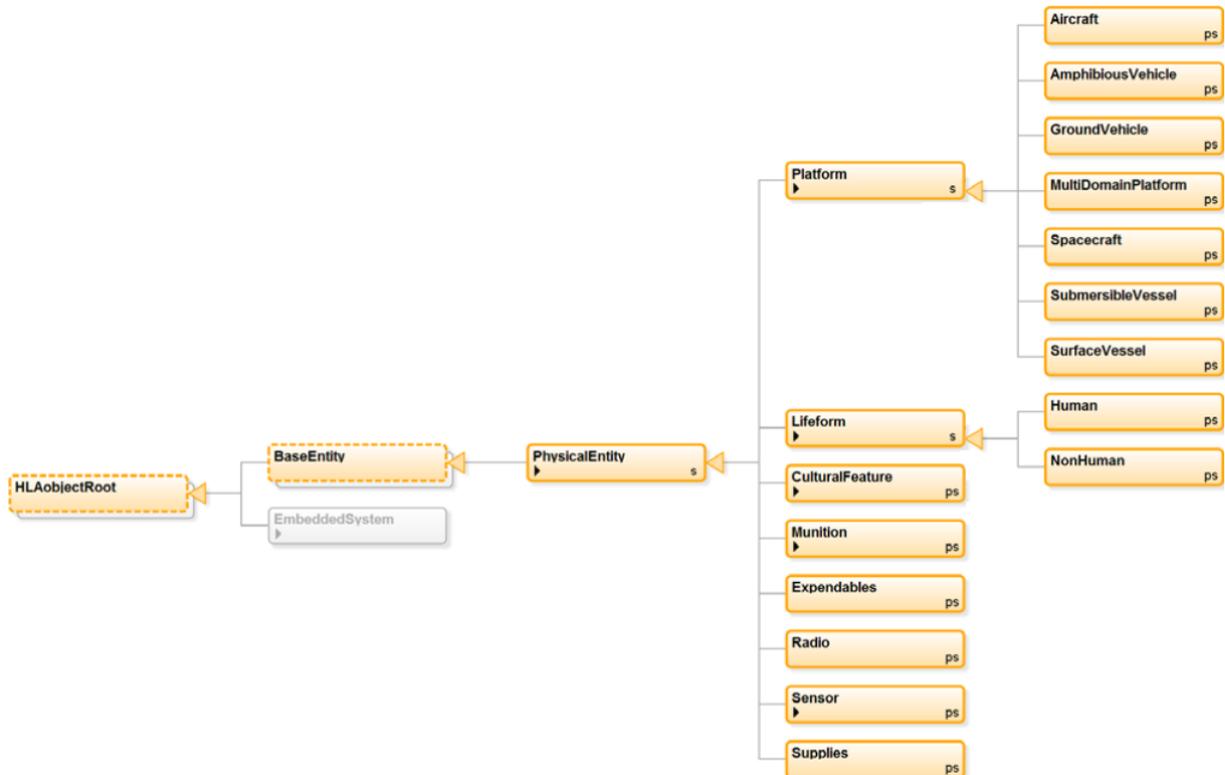
HLA foi criada para desenvolvimento de interoperabilidade entre simuladores. Essa interoperabilidade pode ser garantida através de aplicações compatíveis que possuem acordos de federação entre elas. Para fortalecer isso e promover reusabilidade, foram criados alguns FOMs de referência. Estes são acordos padronizados que se tornaram referência devido a sua generalização e adoção por diversas organizações. Por serem referência, usuários são livres para adicionar conteúdo conforme necessidade. Neste trabalho, são utilizados como base o RPR FOM (*Real-time Platform Reference Federation Object Model*) e o NETN FOM (*NATO Education And Training Network*), ambos descritos brevemente a seguir.

#### 2.3.4.1 RPR FOM

Segundo SISO (2015), o *Real-time Platform Reference Federation Object Model* define uma hierarquia de classes de objeto e interação para a HLA. O RPR FOM foi projetado pela SISO (*Simulation Interoperability Standards Organization*) com base no padrão DIS para suportar simulações em tempo real de entidades físicas discretas, como aviões,

navios, soldados, etc. Essas simulações são consideradas de “tempo real” porque cada segundo do tempo de execução decorrido é equivalente a um segundo de tempo no mundo virtual (SISO, 2015). A Figura 2.2 traz como exemplo da modelagem do RPR a estrutura do módulo de classes de objetos de entidades físicas presentes no RPR.

Figura 2.2 – Estrutura do módulo de classes de entidades físicas do RPR FOM



Fonte: (SISO, 2015).

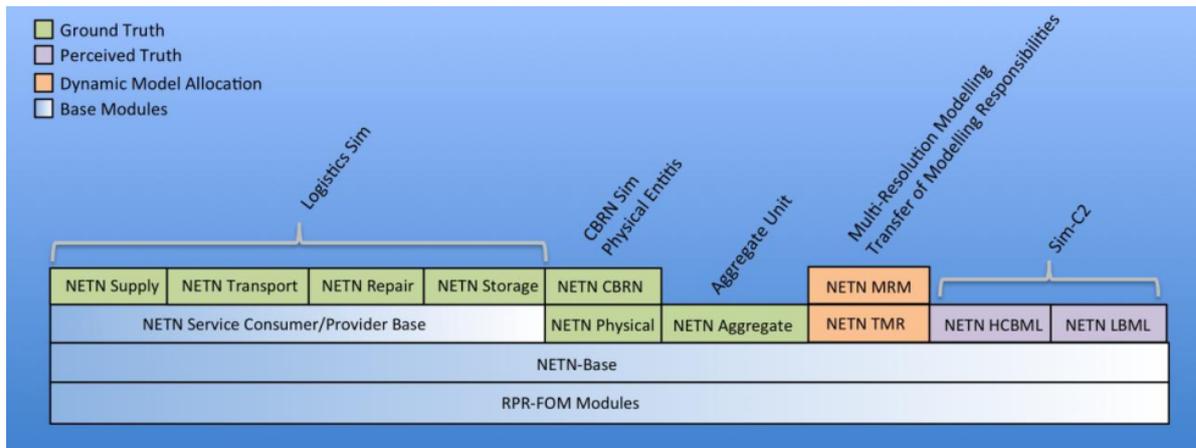
Como pode ser observado na Figura 2.2, a Entidade Física (*PhysicalEntity*) é especificada de forma hierárquica. Ela pertence a um módulo raiz (*HLAObjectRoot*) e a uma Entidade Base (*BaseEntity*), e é especializada em várias subclasses, como a *Munition*, *Radio* e a *Platform*, que por sua vez é especializada em diversas outras subclasses posteriores. Dessa forma, todas as outras classes de objetos e interações são especificadas dentro do FOM. O RPR é o FOM mais difundido, tornando-se base para diversos outros acordos, como o NETN, por exemplo.

#### 2.3.4.2 NETN FOM

O NETN foi desenvolvido pela NATO (*North Atlantic Treaty Organization*) com base no RPR FOM. Diversos módulos foram adicionados para cobrir requisitos dos projetos da organização (NATO, 2012). A Figura 2.3 apresenta a estrutura de módulos do NETN.

Como pode ser observado na Figura 2.3, o FOM da NATO é construído sobre o RPR. Diversos outros módulos são adicionados, como é o caso do NETN *Aggregate* e

Figura 2.3 – Estrutura dos módulos NETN FOM



Fonte: (RUIZ et al., 2016).

NETN MRM, por exemplo, que são módulos específicos deste FOM. Todos os módulos desenvolvidos especificamente para o NETN recebem o prefixo “NETN\_”. Esse acordo de federação é voltado à um alto nível de reusabilidade e complementação de informações que não eram tratadas no RPR, como é o caso da parte logística e de modelagem de multiresolução, por exemplo.

## 2.4 MODELAGEM DE MULTIRESOLUÇÃO

A modelagem de Multiresolução (*Multi-Resolution Modeling* ou MRM) aborda problemas acerca da diferença dos modelos de simulação e sua modelagem. Resolução é a quantidade de detalhes representada em um modelo. Ela pode ser classificada em dois níveis, o nível de unidade e de entidade (TOLK, 2012).

Segundo (STOBER et al., 1995) modelos de nível de unidade tipicamente utilizam/controlam grupos de entidades como agregados e avançam o tempo de forma discreta. Nesse modelo, posição, movimento, composição são mantidos por uma unidade como um todo, ou seja, a unidade é um agregado de entidades. Simulações construtivas são geralmente de nível de unidade e são utilizadas para treinamento de oficiais de mais alto nível (KARR; ROOT, 1994). Por outro lado, nos modelos de nível de entidade, as entidades são representadas individualmente (desagregadas), operando geralmente em tempo real e possuem cenários bem menores (KARR; ROOT, 1994). Sendo assim, modelos de nível de entidade são considerados de alta resolução e granularidade fina, enquanto que modelos de nível de unidade são considerados de baixa resolução ou alta granularidade.

Segundo Tolk (2012), as pesquisas e desenvolvimento de modelos de combate de multiresolução são motivados por fatores como a flexibilidade de modelagem, escalabilidade de cenário, fidelidade nos resultados, realismo contextual, aperfeiçoamento do treinamento e estímulo de sensores. Em muitos modelos de combate de multiresolução, o nível

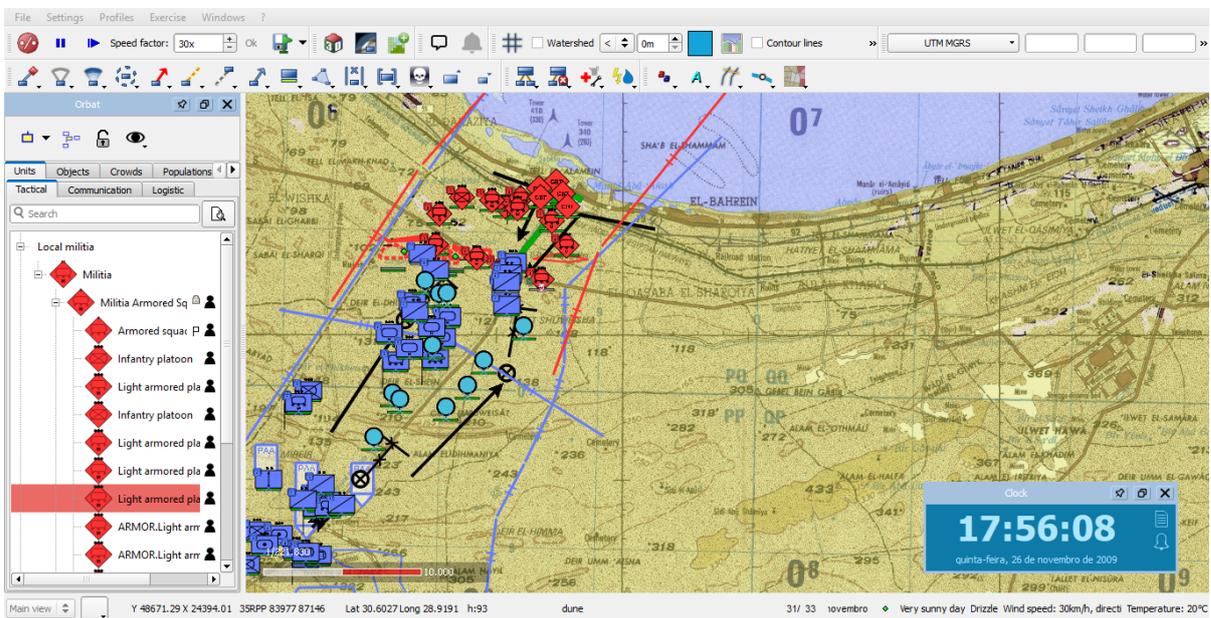
de entidade e de unidade são conectados via protocolos de interoperabilidade de simulação. O DIS contém mensagens padronizadas predefinidas dentro do seu protocolo que suportam a execução de multiresolução. A HLA, no entanto, não inclui questões de multiresolução dentro do próprio protocolo. Sendo assim, é necessário que o desenvolvedor projete soluções para contornar esse problema.

Uma questão bastante discutida no campo de MRM são as interações entre diferentes níveis. Casos tradicionais envolvem o tiro direto e indireto e como questões de cálculo de dano é tratado entre os diferentes modelos que possuem algoritmos diferentes para a mesma tarefa (FRANCESCHINI; PETTY, 1995; KARR; ROOT, 1994).

## 2.5 MASA SWORD

O simulador MASA SWORD é um simulador construtivo, agregado e automatizado, desenvolvido para treinar as habilidades de tomada de decisão de equipes de comando de brigadas e divisões, permitindo a imersão em simulações de conflitos, operações militares e controle e prevenção de desastres (MASA, 2015c). A Figura 2.4 apresenta a interface do simulador.

Figura 2.4 – Interface do cliente de *gaming* do MASA SWORD

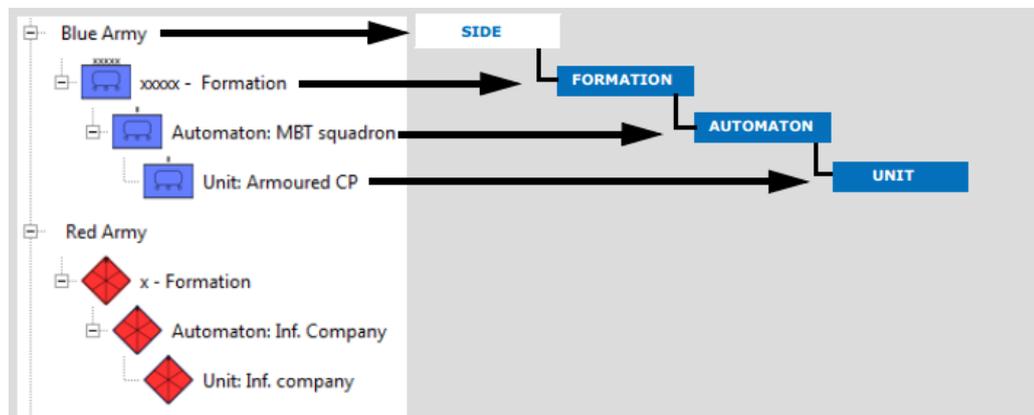


Fonte: MASA.

A Figura 2.4 traz a interface do MASA SWORD. Nela podemos notar a presença da hierarquia do exercício no canto esquerdo, o relógio no canto inferior direito marcando o horário e data do ambiente simulado. Também existe o mapa com as unidades representadas no cenário como o destaque ao centro da interface, e na parte superior diversos menus. Esta é a interface do *Gaming*, principal componente do SWORD. Contudo, existem outros componentes fundamentais, como o *Adapt* que possui a ferramenta de geração de

terreno para o simulador e o *Authoring*, que habilita o usuário a editar a base de dados do modelo de simulação. A hierarquia do exercício no SWORD é formada por quatro níveis, conforme exemplo da Figura 2.5.

Figura 2.5 – Exemplo da constituição hierárquica dos exercícios do MASA SWORD



Fonte: (MASA, 2015b).

Conforme Figura 2.5, os quatro níveis de hierarquia de exercício do SWORD são os seguintes:

- *Party (side)*: representam os times, geralmente compostos por forças amigas e inimigas. Pode conter formações e são o mais alto nível da hierarquia do exercício;
- *Formação (formation)*: é o segundo nível da hierarquia e pode ser um batalhão, uma brigada, uma companhia, etc. Pode conter outras formações ou autômatos. Não são representadas fisicamente no mapa, apenas representam uma entidade organizacional.
- *Autômato (automat)*: sempre está ligado a uma formação e pode conter apenas unidades. Quando engajadas, comandam as unidades.
- *Unidade (unit)*: sempre possui um autômato como superior, mas quando desengajadas podem atuar individualmente sem controle do seu superior. Na HLA, as unidades são conhecidas como entidades agregadas.

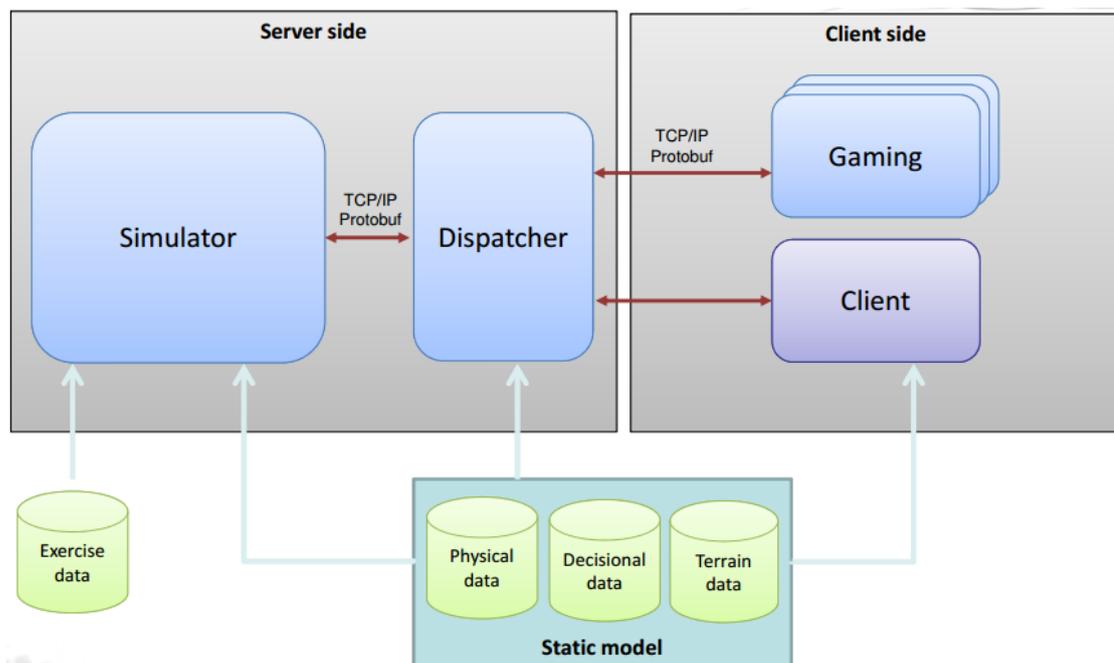
Outra ferramenta importante é a *Prepare*. Essa ferramenta possibilita ao usuário criar todo o exercício, desde a escolha do terreno, até a quantidade de unidades que cada time possui e a quantidade de times. No *Prepare* são estabelecidos todos os componentes que compõem o exercício, incluindo os perfis de acesso ao mesmo, ou seja, toda a situação inicial é estabelecida por meio desta ferramenta. Grande parte dessas informações de configuração são salvas em arquivos XML e podem ser acessadas pelo usuário através do diretório do exercício.

O simulador dispõe de duas formas de comunicação para implementação distribuída da simulação. A primeira é através da sua interface de rede, que implementa a comunicação entre cliente e servidor utilizando o protocolo TCP/IP e Protobuf. A segunda trata-se de um *plug-in* HLA que habilita o SWORD a se conectar com outros simuladores e realizar exercícios distribuídos integrados.

### 2.5.1 Arquitetura de rede do MASA SWORD

A arquitetura do simulador MASA SWORD é baseada no modelo cliente servidor, onde, do lado do servidor, temos o simulador constituído por toda sua base de dados e elementos de simulação, e um *dispatcher*, que faz a comunicação simulador-cliente (e vice-versa). A arquitetura pode ser observada na Figura 2.6.

Figura 2.6 – Arquitetura de rede do MASA SWORD



Fonte: (MASA, 2015a).

Conforme Figura 2.6, que apresenta a arquitetura de rede do MASA SWORD, do lado do servidor há o simulador que armazena todos os elementos do exercício e processa todas as ações do modelo e o *dispatcher*, que faz o papel de intermediário na comunicação do simulador com os seus clientes. Do outro lado, no papel do cliente temos, inicialmente, o *Gaming*, que é a aplicação acessada pelo usuário para controle do exercício simulado (MASA, 2015a), e um *client*, que pode ser desenvolvido para receber e enviar mensagens para o servidor. É importante notar a presença do modelo estático de dados e os dados do exercício. Eles guardam informações sobre o simulador e exercícios configurados para ele. Cada exercício possui seu diretório onde dados particulares do mesmo são armazenados.

Para estabelecer a comunicação entre cliente e simulador é necessário que o *Gaming* inicie um exercício. De forma simplificada, para desenvolver um cliente é necessário estabelecer um canal entre ambos. Após estabelecer o canal, é preciso enviar uma mensagem de autenticação contendo usuário e senha. Essa mensagem será autenticada pelo simulador, e caso a autenticidade do usuário seja comprovada, ele passará a receber todos os dados enviados pelo *dispatcher* (MASA, 2015a). Cliente e Servidor se comunicam utilizando Protocolo TCP/IP. O número de porta é definido no arquivo “*session.xml*”, usualmente o SWORD utiliza-se da porta 10001. As mensagens enviadas pela rede contém três campos (tamanho, *tag* e conteúdo), conforme Figura 2.7.

Figura 2.7 – Composição da mensagem enviada pelo MASA SWORD



Fonte: (MASA, 2015a).

Observando a Figura 2.7, o campo *size* é o tamanho do corpo da mensagem especificado em bytes. A *tag* contém o tipo do conteúdo que está dentro da mensagem (Figura 2.8) e o *content* é o conteúdo dela.

Figura 2.8 – Tipos de *tags* de mensagens utilizadas pelo MASA SWORD

Message	Tag
AarToClient	3335378374
AdminToLauncher	487143762
AuthenticationToClient	468259588
ClientToAar	496354747
ClientToAuthentication	3393095543
ClientToMessenger	2651046712
ClientToReplay	745507513
ClientToSim	496449857
DispatcherToClient	3012412245
DispatcherToSim	4036107609
LauncherToAdmin	1796776068
MessengerToClient	4281023046
ReplayToClient	2824568610
SimToClient	3540368649

Fonte: (MASA, 2015a).

Como citado, a *tag* especifica o tipo de mensagem. Os tipos são apresentados na Figura 2.8. Os principais tipos de mensagens para o contexto deste trabalho são: *ClientToAuthentication*, que é o tipo de mensagem que o cliente utiliza para se autenticar ao servidor; *SimToClient* que são mensagens enviadas do simulador para o cliente contendo

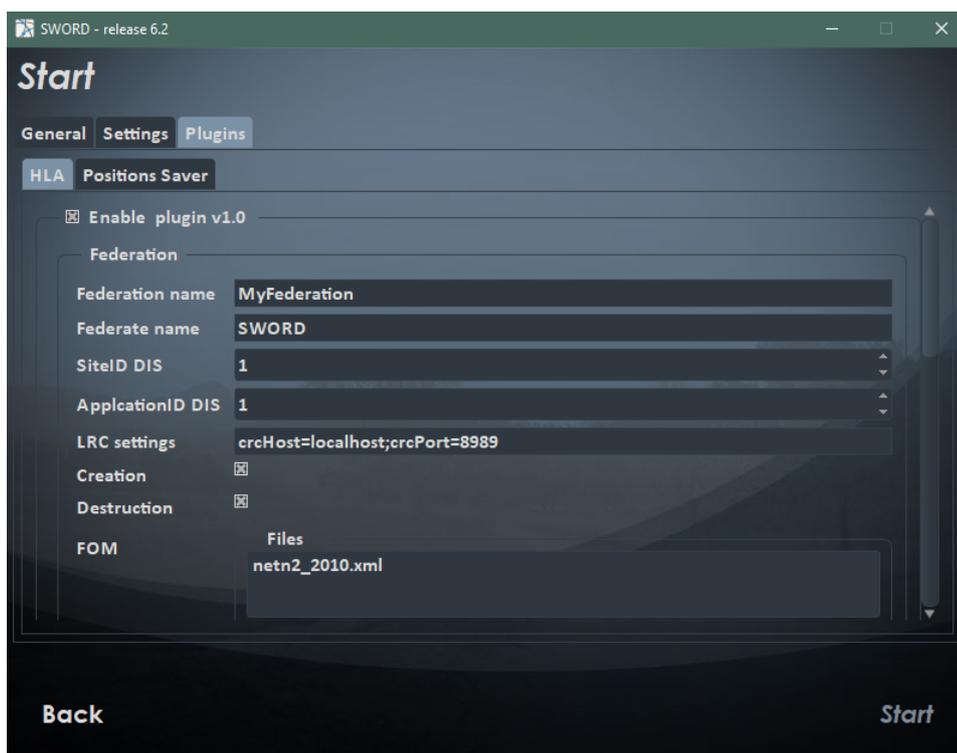
as mais diversas informações, como posição das unidades, ordens, *tick*, etc; *ClientToSim* são as mensagens que o cliente envia ao simulador, que pode conter diversas ações como controle de pausa e andamento da simulação, ações mágicas e ordens, por exemplo.

Apenas o conteúdo é serializado utilizando Protobuf. *Protocol Buffers* (Protobuf) é um mecanismo flexível, eficiente e automatizado para serialização estruturada de dados, desenvolvido pela empresa Google com o objetivo de tornar a comunicação mais eficiente entre seus milhares de servidores (GOOGLE, 2017). Comparado ao XML, o Protobuf é menor (de 3 a 10 vezes menor), mais simples e mais rápido (de 20 a 100 vezes mais rápido). Atualmente o Protobuf se encontra na sua terceira versão (proto3) e dispõe de suporte a cinco linguagens de programação: C++, C#, Go, Java e Python. O desenvolvedor pode definir como os dados serão estruturados, e gerar código fonte para gerenciá-los. A estrutura dos dados é definida em arquivos “\*.proto” que são compilados através do compilador *Protocol Buffer* para geração das classes para acesso dos dados na linguagem escolhida.

### 2.5.2 *Plug-in* HLA

Para trabalhar de forma interoperável, existe um *plug-in* HLA desenvolvido para o simulador da MASA. Ele possui compatibilidade com três FOMs: RPR2, NETN e Alliance 2 (MASA, 2011). A Figura 2.9 traz a interface do menu de configuração do *plug-in* HLA do simulador.

Figura 2.9 – Interface de configuração do *plug-in* HLA do MASA SWORD



Fonte: MASA.

Na Figura 2.9 é possível notar informações referentes a conexão HLA/RTI, como o nome da federação (*MyFederation*), o nome do federado (*SWORD*), configurações de LRC, opções de criação e destruição da federação e o FOM utilizado, neste caso o NETN. Além dessas informações, há também opções de tempo e opções avançadas (escolha do protocolo HLA utilizado, versão do FOM, opções de publicação e propriedade de objetos, etc) (MASA, 2011).

O MASA SWORD possui um SOM focado na simulação construtiva e trabalha com uma quantidade pequena de objetos e interações. As classes de objetos com os quais o *plug-in* trabalha estão no Quadro 2.1.

Quadro 2.1 – Classes de objetos do SOM do MASA SWORD

Classes de objetos	P/S
HLAobjectRoot.BaseEntity.PhysicalEntity.Platform.Aircraft	PS
HLAobjectRoot.BaseEntity.PhysicalEntity.Platform.SurfaceVessel	PS
HLAobjectRoot.BaseEntity.PhysicalEntity.CulturalFeature	PS
HLAobjectRoot.BaseEntity.AggregateEntity	PS
HLAobjectRoot.EnvironmentObject.ArealObject.OtherArealObject	PS
HLAobjectRoot.EnvironmentObject.LinearObject.BreachableLinearObject	PS
HLAobjectRoot.EnvironmentObject.PointObject.BreachablePointObject	PS
HLAobjectRoot.EnvironmentObject.PointObject.OtherPointObject	PS
HLAobjectRoot.Minefield	PS

Fonte: Adaptado de (MASA, 2011).

O Quadro 2.1 apresenta as classes de objetos que o SWORD publica/assina. A coluna da esquerda traz o nome do objeto e toda hierarquia ao qual este está ligado e na coluna da direita está a relação de interesse do SWORD em cada uma das classes, sendo que a letra P se refere ao interesse de publicação e a letra S ao interesse de receber informações sobre da classe. Como pode-se observar, o SWORD demonstra interesse de publicação e recebimento de todas as classes de objetos do quadro. Um detalhe importante, e que não foi colocado no quadro, é que quando se utiliza o FOM NETN, existem objetos próprios do NETN que podem ser utilizados no lugar dos objetos base do RPR. Esses objetos são semelhantes aos originais, contudo trazem algumas características próprias planejadas pela NATO (MASA, 2011). Exemplo disso é o *NETN\_Aircraft*.

É importante destacar que o *plug-in* HLA do MASA SWORD trabalha com objetos agregados e publica as unidades pertencentes ao exercício em forma de entidades agregadas (classe de objeto *AggregateEntity*). Essas entidades correspondem a um nível da hierarquia. Quando se utiliza o FOM NETN através do *plug-in* HLA do SWORD, as entidades agregadas correspondem a unidades. Ainda em relação as capacidade do seu *plug-in* HLA, as interações presentes no SOM do SWORD podem ser observadas no Quadro 2.2.

Da mesma forma que no Quadro 2.1, o Quadro 2.2 traz o nome das interações do lado esquerdo e seu *publish/subscribe* na coluna do lado direito. Apenas essas quatro interações estão presentes em ambos FOMs compatíveis. Importante notar a presença de

Quadro 2.2 – Classes de interações do SOM do MASA SWORD

Classes de interações	P/S
HLAinteractionRoot.WeaponFire	PS
HLAinteractionRoot.MunitionDetonation	PS
HLAinteractionRoot.Acknowledge	PS
HLAinteractionRoot.TransferControl	PS

Fonte: Adaptado de (MASA, 2011).

uma interação de transferência de controle (*transferControl*), que é utilizada para passagem de propriedade de objetos do simulador.

Assim como nas classes de objetos, o SWORD possui compatibilidade com algumas interações que estão presentes apenas no NETN. No Quadro 2.3 estão presentes as funcionalidades adicionais do SWORD em relação a seu *plug-in* HLA quando utiliza o FOM NETN e opta por trabalhar com seus módulos (opção que deve ser marcada na interface de configuração do *plug-in*).

Quadro 2.3 – Classes de objetos e interações do SOM do MASA SWORD ao utilizar NETN

Classe de objeto/interação	P/S
HLAobjectRoot.BaseEntity.PhysicalEntity.Platform.Aircraft.NETN_Aircraft	PS
HLAobjectRoot.BaseEntity.PhysicalEntity.Platform.SurfaceVessel.NETN_SurfaceVessel	PS
HLAobjectRoot.BaseEntity.AggregateEntity.NETN_Aggregate	PS
HLAobjectRoot.ATP45HazardAreal	PS
HLAobjectRoot.RawDataHazardContourGroup	PS
HLAinteractionRoot.SCP_Service.SCP_RequestService.RequestTransport	PS
HLAinteractionRoot.SCP_Service.SCP_OfferService.OfferTransport	PS
HLAinteractionRoot.SCP_Service.SCP_ServiceComplete	PS
HLAinteractionRoot.SCP_Service.SCP_ReadyToReceiveService	PS
HLAinteractionRoot.SCP_Service.SCP_AcceptOffer	PS
HLAinteractionRoot.SCP_Service.SCP_ServiceStarted	PS
HLAinteractionRoot.SCP_Service.SCP_ServiceReceived	PS
HLAinteractionRoot.SCP_Service.SCP_RejectOffer	PS
HLAinteractionRoot.SCP_Service.SCP_CancelService	PS
HLAinteractionRoot.SCP_Service.TransportEmbarkmentStatus	PS
HLAinteractionRoot.SCP_Service.TransportDisembarkmentStatus	PS
HLAinteractionRoot.SCP_Service.TransportDestroyedEntities	PS
HLAinteractionRoot.TMR.TMR_OfferTransferModellingResponsibility	PS
HLAinteractionRoot.TMR.TMR_RequestTransferModellingResponsibilitys	PS
HLAinteractionRoot.MRM_Object.MRM_DisaggregationRequest	PS
HLAinteractionRoot.MRM_Object.MRM_DisaggregationResponse	PS
HLAinteractionRoot.MRM_Object.MRM_AggregationRequest	PS
HLAinteractionRoot.MRM_Object.MRM_AggregationResponse	PS
HLAinteractionRoot.MRM_Object.MRM_ActionComplete	PS

Fonte: MASA.

Como pode-se notar, existem diversas classes de objetos e interações disponíveis para o SWORD ao optar pelo NETN. São cinco classes de objetos adicionais, sendo que três são especializações dos objetos *Aircraft*, *SurfaceVessel* e *Aggregate*, todas disponíveis já no RPR. O número de interações é bem maior, porém está ligado com funcionalidades de serviços específicos, como é o caso das interações pertencentes ao módulo *SCP\_Service*, que trata-se da classe base para todas as interações do NETN relacionadas ao padrão consumidor-fornecedor (*Consumer-Provider Pattern*). Além deste, outros dois módulos base estão presentes, o *TMR*, que trata-se da classe base para interações de Transferência de Modelagem de Responsabilidade, e o *MRM\_Object*, que é classe base para interações de Modelagem de Multiresolução.

## 2.6 TRABALHOS RELACIONADOS

Esta seção está destinada a apresentação de trabalhos que tem relação com esta dissertação em algum segmento de sua execução. Para melhor entendimento, ela foi dividida em duas subseções: atento ao fato de que esse trabalho busca resolver problemas acerca da diferença de resolução em uma federação HLA envolvendo o SWORD, a Subseção 2.6.1 apresenta brevemente os principais trabalhos na construção de soluções ou federações de multiresolução no campo de simulação; já a Subseção 2.6.2 aborda trabalhos dentro do contexto da Federação SIS-ASTROS, que é uma federação que representa um dos primeiros esforços no sentido de desenvolvimento de simulação de multiresolução, e é utilizada para validação da solução desenvolvida neste trabalho.

### 2.6.1 Multiresolução em simulação

Segundo Tolk (2012), existiam pelo menos quinze casos de integração de simuladores de diferentes resoluções até a publicação de seu livro em 2012. Os primeiros trabalhos nesse sentido foram desenvolvidos no início da década de 1990, com a evolução da simulação distribuída e início das tentativas de integração de diferentes sistemas de simulação.

Um dos primeiros trabalhos nesse sentido é o de Karr e Root (1994), que integra o simulador construtivo *Eagle* com um simulador virtual que opera através do SIMNET. Neste trabalho vários temas são abordados, como, por exemplo, operações de agregação e desagregação, pseudo-desagregação, interações de fogo direto e indireto. Karr e Root (1994) também citam uma solução para a passagem de ordens entre diferentes níveis de resolução, que seria a transmissão na forma escrita entre os simuladores.

Franceschini e Petty (1995) desenvolvem um tutorial para desenvolvimento de integração utilizando o DIS e explica cada módulo do seu sistema proposto, que, além do simulador virtual e do construtivo, possui um módulo acoplado ao construtivo que desenvolve operações de agregação/desagregação, e um módulo de CGF (*Computer Generated*

*Forces*), que são forças semi-automatizadas que controlam unidades remotas em ambos modelos. Além disso, neste trabalho são abordadas questões de quando o controle entre construtivo e virtual deve ser transferido. Para isso, existem gatilhos, que podem ser disparados nas seguintes ocasiões:

- Terreno ou área de alta resolução: quando se determina que quando uma unidade entra em determinada área, o controle da simulação seja passado para o simulador virtual para obter um maior nível de detalhe;
- Intenção do operador: quando o operador julgar relevante a mudança de paradigma de simulação;
- Proximidade do inimigo: a distância para o inimigo pode ser importante para se trocar de resolução, dependendo do nível de fidelidade ou tipo de algoritmo que se deseja para execução de tarefas de combate, por exemplo;
- Intenção de interação: quando um simulador demonstrar intenção de interagir com outro de resolução diferente, operações de agregação/desagregação podem ser utilizadas para controlar desafios acerca das diferenças;
- Visão do comandante: o gatilho pode ser disparado pela opção do comandante em obter mais ou menos detalhes no campo de batalha, por exemplo.

O trabalho de Franceschini e Petty (1995) ainda aborda de maneira rasa questões de sincronização temporal e espacial, além da interação entre os níveis. Os autores trabalham com interação de fogo direto/indireto entre os simuladores. Além disso, para as ordens, é dito que a transmissão delas através dos modelos pode ser importante no sentido construtivo-virtual, onde o simulador virtual pode contar com um planejador para receber e processar as ordens ou podem apenas mostrá-las para o operador do simulador virtual, para que este possa seguir as instruções nela expressas enquanto segue seu treinamento.

Em Schricker, Franceschini e Adkins (1998) e Schricker, Barbosa e Smith (1999) são abordadas questões de pseudo-desagregação e estas são aplicadas no contexto dos simuladores Eagle e ModSaf. Em especial, Schricker, Barbosa e Smith (1999) abordam questões sobre o posicionamento de unidades desagregadas no simulador virtual, resolvendo problemas de correlação da localização destes. Neste trabalho, é desenvolvida uma técnica utilizada pelo simulador virtual para suprimir a taxa de atualizações lenta do modelo construtivo, resolvendo assim problemas de sincronização dos modelos. Em Schricker, Franceschini e Adkins (1998) ainda é adicionada uma capacidade de interação nas entidades pseudo-desagregadas, e tais adições variam de acordo com requisitos.

No contexto de federações HLA, Rieger e Beeker (2000) apresentam um dos primeiros trabalhos que falam sobre interoperabilidade entre modelos de nível de unidade e entidade através da arquitetura de alto nível. Uma questão importante tratada no artigo

é a gerência de propriedade dos objetos como uma forma de contornar obstáculos na diferença de algoritmos. Segundo o autor, a passagem de propriedade no processo de agregação/desagregação possibilita a execução de determinadas atividades em um único modelo para evitar anomalias que poderiam surgir nas diferenças dos modelos.

Em Bowers e Prochnow (2003) são descritos aspectos de integração na federação JTLS-JCATS, principalmente no gerenciamento de propriedade dos objetos através do serviço fornecido pelo RTI. Nesta federação onde JTLS trabalha em nível unidade e JCATS em nível de entidade, JTLS simula um cenário muito grande e com poucos detalhes, enquanto JCATS simula uma pequena área dentro deste cenário maior. As informações são enviadas de forma agregada pelo JTLS e quando o operador requer maior nível de detalhes, a simulação passa a ocorrer no JCATS, passando a propriedade de certos atributos do modelo de nível de unidade para o modelo de entidade. Assim, diferenças nos algoritmos são vencidas pela utilização de um único modelo para resolver certa tarefa.

Em Zabek et al. (2004), o JCATS é integrado com outro simulador construtivo, o CBS. Neste trabalho também é utilizada a gerência de propriedade do HLA para habilitar operações de agregação/desagregação. Para desencadear essas operações, ao invés da seleção do operador, são desenhados polígonos no mapa que criam zonas de desagregação, onde a unidade que entra neste polígono passa a ser posse do simulador de nível de entidade, o JCATS. Da mesma forma, quando sai desta área, o CBS retoma o controle. Quando desagregada, o operador pode ordenar as entidades no JCATS. Fogo indireto é permitido nos dois sentidos e os resultados de danos sobre entidades desagregadas são enviados via interações HLA ao CBS para que ele reflita os danos em seu cenário.

Kwong et al. (2009) trata questões de capacidade do sistema, diferenças na taxa de atualização e no terreno como formas de enriquecer a visualização da simulação em integração do construtivo 2D e o virtual com visualização 3D, ou seja, esse trabalho tem como foco a melhoria nas capacidades de visualização em ambientes integrados de diferentes níveis. No trabalho de Raue e Gallois (2011) são utilizados COTS para treinamento envolvendo virtual e construtivo e ordens são transferidas do simulador de alto nível para o baixo, no entanto nenhum comportamento é disparado pela falta de inteligência artificial para a executar a tarefa.

Os trabalhos discutidos nesta subseção trouxeram contribuições para o campo de interoperabilidade em modelos de simulação com diferentes granularidades de detalhes de abstração. Cada um aborda algum ou vários desafios nessa integração. O trabalho aqui apresentado se diferencia dos demais por tentar superar a carência de informações no caso de federações multiresolução onde há interoperabilidade entre simulador construtivo e simulador virtual tático, fornecendo meios para complementação do exercício, através de uma melhor sincronia entre modelos, representação situacional mais apurada do ambiente construtivo sendo refletido no ambiente virtual tático e transferência de ordens entre os modelos de simulação.

### 2.6.2 Federação SIS-ASTROS

Durante o desenvolvimento da Federação SIS-ASTROS, alguns trabalhos foram desenvolvidos e diversos problemas tratados. Neste contexto, dois tópicos foram discutidos de forma mais destacada, a conversão de resolução e o *plug-in* HLA do SVTat REOP, e ambos se transformaram em componentes da federação.

Seção restrita por conter dados sensíveis. Contatar [ceretta@inf.ufsm.br](mailto:ceretta@inf.ufsm.br) para mais detalhes.

### 3 REQUISITOS PARA INTEROPERABILIDADE ENTRE SIMULADOR CONSTRUTIVO E VIRTUAL TÁTICO

Esse capítulo tem por objetivo apresentar uma discussão sobre requisitos para interoperabilidade entre um simulador construtivo e um virtual tático, além de especificar a modelagem de um módulo FOM com base nestes requisitos. O capítulo traz primeiramente na Seção 3.1 uma discussão sobre requisitos no desenvolvimento de interoperabilidade em um ambiente multiresolução, em especial a partir da modelagem construtiva do SWORD e integração deste com um simulador virtual tático; logo após, na Seção 3.2 o novo módulo FOM é especificado conforme os requisitos levantados na seção anterior.

#### 3.1 REQUISITOS PARA COMPLEMENTAÇÃO DO EXERCÍCIO

Desenvolvimento interoperável entre simuladores de resoluções diferentes é uma tarefa não trivial que exige tratativas diferenciadas, pois geralmente são desenvolvidas interfaces de interoperação baseadas no nível de resolução do próprio modelo. O SWORD não é diferente, e por se tratar de um modelo construtivo, abstrai grande parte dos detalhes e trabalha com objetos de alta resolução através do seu *plug-in* HLA. Nessa seção, serão apresentados requisitos provenientes da tentativa de tornar o MASA SWORD compatível para interoperar com um simulador virtual com características táticas, ou seja, um simulador virtual tático.

Seção restrita por conter dados sensíveis. Contatar [ceretta@inf.ufsm.br](mailto:ceretta@inf.ufsm.br) para mais detalhes.

#### 3.2 NOVO MÓDULO FOM

Uma das grandes evoluções provenientes do HLA *Evolved* (última versão da norma) foi a possibilidade de modularização do FOM. Essa característica importante permite, além de dividir o acordo da federação em vários arquivos diferentes, a adição de novas capacidades após o exercício (federação) iniciado, ou seja, um federado pode entrar posteriormente ao início da federação e adicionar novas funcionalidades através de módulos (MÖLLER; LÖFSTRAND, 2007).

Seção restrita por conter dados sensíveis. Contatar [ceretta@inf.ufsm.br](mailto:ceretta@inf.ufsm.br) para mais detalhes.

## 4 PROJETO DE EXTENSÃO INDIRETA DO SOM DO SIMULADOR MASA SWORD

Esse capítulo tem por objetivo apresentar o projeto da solução do *plug-in* HLA complementar do SWORD baseado nas necessidades de interoperabilidade entre simulador construtivo e simulador virtual tático identificadas no Capítulo 3. O simulador MASA SWORD possui código-fonte fechado, assim não é possível alterá-lo ou estendê-lo de forma direta. No entanto, apesar de fechado, o sistema disponibiliza uma interface de comunicação através da sua interface de rede e do uso do mecanismo de serialização estruturada de dados conhecido como Protobuf. Através dessa interface é possível obter uma série de informações que podem ser utilizadas para estender indiretamente o seu SOM, ou seja, suas capacidades referentes à interface HLA. Além da interface de rede do simulador, muitos dos elementos de sua base de dados são acessíveis e podem ser usados no suprimento de informações. O capítulo é inteiramente dedicado ao projeto e desenvolvimento do *plug-in* HLA complementar, iniciando pela Seção 4.1, que apresenta os objetivos técnicos da solução. A arquitetura do *plug-in* HLA e cada um de seus componentes são detalhados na Seção 4.2.

### 4.1 OBJETIVOS TÉCNICOS DA SOLUÇÃO

Tendo como base o acordo de federação especificado no capítulo anterior, foi projetado e desenvolvido um novo *plug-in* HLA que pudesse agregar funcionalidade complementares ao SOM do MASA SWORD de forma indireta. O *plug-in* HLA complementar desenvolvido neste trabalho tem papel de fazer com que os requisitos planejados na Seção 3.1 pudessem ser atendidos através da implementação de uma interface para publicação dos objetos, interações e tipos de dados modelados no novo módulo FOM, conforme Seção 3.2. Para alcançar esse objetivo, a solução apresenta os seguintes objetivos técnicos:

- Conectar-se à federação como um federado;
- Conectar-se ao MASA SWORD através da sua interface de rede e conseguir processar as mensagens recebidas através desta, filtrando informações relevantes que possam ser do interesse da federação;
- Ler arquivos de configuração do exercício do MASA SWORD afim de capturar informações relevantes para o exercício;
- Publicar os objetos modelados conforme o acordo de federação estabelecido. Para isso também deverá utilizar os tipos de dados descritos nesse acordo. Os objetos publicados pelo SWAF deverão possuir o prefixo "SWAF\_" para identificá-los;
- Publicar interações modeladas conforme o acordo de federação estabelecido;

- Receber informações sobre unidades publicadas no SWORD através da interface HLA e relacioná-las com unidades locais descobertas através da interface de rede.

Cada um dos itens acima representa um objetivo técnico para construção do SWAF. O SWAF se utiliza da metodologia de extensão indireta do SOM, que proporciona a presença de um federado adicional para fornecimento de informações complementares do simulador construtivo. O MASA SWORD possui seu próprio *plug-in* HLA atuando como federado primário na federação (detalhes sobre ele estão na seção 2.5). Todas informações provenientes do MASA SWORD que são publicadas na federação e que vão além do escopo do seu *plug-in* HLA original, são realizadas pelo SWAF.

## 4.2 ARQUITETURA E FUNCIONAMENTO

O SWAF é o *plug-in* complementar desenvolvido neste trabalho. Sua função é fornecer informações adicionais do SWORD para a federação. O SWAF é constituído por dois elementos, uma DLL contendo as funções que fazem a interface RTI/HLA, e uma aplicação que atua como um cliente do SWORD, que se conecta ao simulador, recebe mensagens e as processa conforme desejado. Além disso, o cliente SWORD obtém informações dos arquivos de configuração do exercício. Esse cliente utiliza as funções importadas da DLL para interagir e trocar dados com a federação.

Seção restrita por conter dados sensíveis. Contatar [ceretta@inf.ufsm.br](mailto:ceretta@inf.ufsm.br) para mais detalhes.

### 4.2.1 SOM atualizado

Com a implementação do SWAF, o SOM do MASA SWORD pôde ser estendido de forma indireta. Essa extensão pode fornecer meios para suplementação de informação com base nos requisitos levantados. O SOM atualizado do SWORD referente as classes de objetos pode ser observado no Quadro 4.1.

O Quadro 4.1 apresenta o SOM atualizado do MASA SWORD a partir da adição do novo *plug-in* HLA desenvolvido neste trabalho. Neste quadro podemos notar a coluna da esquerda contendo nome da classe de objeto, assim como toda hierarquia a qual a mesma pertence. Na coluna do meio está seu nível de interesse através do *publish/subscribe*, e na coluna da direita está a origem da classe, ou seja, se ela é proveniente do *plug-in* HLA original do SWORD ou se é originária do SWAF. Sendo assim, pode-se notar a presença de quatro novas classes de objetos, além do SOM original do simulador. Essas quatro novas classes de objetos estão ligadas à classe raiz *SWAF\_Objects* e são provenientes da modelagem do novo módulo FOM. É importante notar que as quatro novas classes possuem apenas interesse de publicação segundo o SOM. Isso quer dizer que informa-

Quadro 4.1 – Classes de objetos do SOM atualizado do MASA SWORD

<b>Classes de objetos</b>	<b>P/S</b>	<b>Origem</b>
HLAobjectRoot.BaseEntity.PhysicalEntity.Platform.Aircraft	PS	Original
HLAobjectRoot.BaseEntity.PhysicalEntity.Platform.SurfaceVessel	PS	Original
HLAobjectRoot.BaseEntity.PhysicalEntity.CulturalFeature	PS	Original
HLAobjectRoot.BaseEntity.AggregateEntity	PS	Original
HLAobjectRoot.EnvironmentObject.ArealObject.OtherArealObject	PS	Original
HLAobjectRoot.EnvironmentObject.LinearObject.BreachableLinearObject	PS	Original
HLAobjectRoot.EnvironmentObject.PointObject.BreachablePointObject	PS	Original
HLAobjectRoot.EnvironmentObject.PointObject.OtherPointObject	PS	Original
HLAobjectRoot.Minefield	PS	Original
HLAobjectRoot.SWAF_Objects.Drawing	P	SWAF
HLAobjectRoot.SWAF_Objects.DrawingLayer	P	SWAF
HLAobjectRoot.SWAF_Objects.ExerciseProfile	P	SWAF
HLAobjectRoot.SWAF_Objects.Automat	P	SWAF

Fonte: Autor.

ções desses objetos serão apenas publicadas e não há interesse em receber informações desses objetos por outros federados.

Além das classes de objetos, foram adicionadas algumas classes de interações ao SOM do MASA SWORD através do novo *plug-in* HLA. As classes de interação do SOM atualizado podem ser observadas no Quadro 4.2.

Quadro 4.2 – Classes de interações do SOM atualizado do MASA SWORD

<b>Classes de interação</b>	<b>P/S</b>	<b>Origem</b>
HLAinteractionRoot.WeaponFire	PS	Original
HLAinteractionRoot.MunitionDetonation	PS	Original
HLAinteractionRoot.Acknowledge	PS	Original
HLAinteractionRoot.TransferControl	PS	Original
HLAinteractionRoot.SWAF_Interactions.SWAF_MoveToLocation	P	SWAF
HLAinteractionRoot.SWAF_Interactions.SWAF_FireAtLocation	P	SWAF
HLAinteractionRoot.SWAF_Interactions.Deploy	P	SWAF
HLAinteractionRoot.SWAF_Interactions.Time	P	SWAF
HLAinteractionRoot.SWAF_Interactions.TimeScale	P	SWAF
HLAinteractionRoot.SWAF_Interactions.UnitTeleport	P	SWAF

Fonte: Autor.

O Quadro 4.2 aponta o SOM atualizado contendo as quatro interações originais do *plug-in* HLA do SWORD, e, além delas, seis novas interações inseridas através do SWAF. Assim como no Quadro 4.1, no Quadro 4.2 a coluna da esquerda também apresenta o

nome e posição da classe na hierarquia do acordo de federação, a coluna da direita o interesse para publicar e assinar a classe de interação, e a coluna da direita apresenta a procedência da classe (*plug-in* HLA original do SWORD ou SWAF). Todas as interações novas possuem apenas interesse de publicação pois não houveram requisitos para volta de interações do SWAF para o SWORD.

## 5 VALIDAÇÃO E RESULTADOS

Este capítulo tem por objetivo apresentar um estudo de caso de utilização do SWAF em conjunto com o SWORD em uma federação HLA que conta com um simulador virtual tático. Este último é um simulador com nível de detalhes mais elevado e exige um maior nível de informações para realização de um exercício conjunto e retratação de um ambiente fiel. Portanto, o objetivo da realização deste experimento é demonstrar a importância de se fornecer informações adicionais do SWORD para o simulador virtual tático através do SWAF. O estudo de caso foi desenvolvido sobre a Federação SIS-ASTROS e os resultados serão apresentados em forma dos ganhos obtidos a partir da utilização do *plug-in* HLA complementar do SWORD dentro dessa federação. O capítulo se divide em três seções: a Seção 5.1 apresenta a Federação SIS-ASTROS e fornece uma visão geral sobre sua constituição, focando principalmente no SVTat REOP; o estudo de caso é desenvolvido na Seção 5.2; a avaliação dos resultados alcançados é discutida na Seção 5.3, que também apresenta os prós e contras da extensão indireta do SOM para introdução de novas funcionalidades.

### 5.1 FEDERAÇÃO SIS-ASTROS

A Federação SIS-ASTROS é uma federação HLA em desenvolvimento. Atualmente ela conta com dois simuladores, sendo o Simulador Virtual Tático de REOP e o MASA SWORD. Além dos dois simuladores, a federação também conta com um conversor de multiresolução baseado em doutrinas, desenvolvido por Paul (2017), que converte objetos de baixa resolução para representação no modelo de alta resolução e vice-versa. Esta seção apresenta uma visão geral sobre a Federação SIS-ASTROS.

Seção restrita por conter dados sensíveis. Contatar [ceretta@inf.ufsm.br](mailto:ceretta@inf.ufsm.br) para mais detalhes.

### 5.2 ESTUDO DE CASO

O objetivo desse estudo de caso é demonstrar a utilidade do *plug-in* HLA complementar do MASA SWORD atuando em conjunto com os demais federados da Federação SIS-ASTROS. Neste estudo, o SVTat REOP será utilizado como uma extensão do MASA SWORD, refletindo suas ações e abstrações com auxílio do SWAF, além de alcançar sincronia a partir da adoção do *plug-in* HLA complementar.

Seção restrita por conter dados sensíveis. Contatar [ceretta@inf.ufsm.br](mailto:ceretta@inf.ufsm.br) para mais detalhes.

### 5.3 AVALIAÇÃO DA SOLUÇÃO DESENVOLVIDA

O SWAF atingiu seu objetivo que era estender o SOM do MASA SWORD para fins de interoperabilidade com simulador virtual tático a partir da modelagem de um novo módulo FOM. A metodologia de extensão indireta do SOM do simulador foi utilizada e sua capacidade foi validada com o estudo de caso, demonstrando ser uma opção válida para casos semelhantes que envolvam simuladores comerciais.

O estudo de caso buscou ressaltar os ganhos a partir da utilização do SWAF como um federado da Federação SIS-ASTROS. Os resultados foram obtidos através da utilização do novo *plug-in* HLA complementar que foi projetado através do Módulo SWAF do FOM especializado para realizar a extensão indireta do SOM do SWORD. É importante destacar que essas extensões não ferem a definição dos FOMs de referência, elas apenas acrescentam um conjunto de classes de informações adicionais que podem ser utilizadas pelos federados para complementação do exercício. Seguindo os requisitos de interoperabilidade entre simulador construtivo e virtual tático, a extensão do SOM ocorreu em três campos: transferência situacional, transmissão de ordens e sincronização temporal. Embora essas tenham sido as contribuições no quesito de informações adicionais para o exercício, o *plug-in* HLA auxiliar pode ser programado para fornecer qualquer tipo de informação desde que o SWORD possua a representação em seu modelo de simulação.

A transferência situacional proporciona a representação de abstrações internas do SWORD e que não são publicadas através de seu *plug-in* HLA. Essas abstrações podem ser muito interessantes para diversos outros simuladores, incluindo o simulador Virtual Tático de REOP, e o estudo de caso provou isso através da representação de desenhos e da capacidade de filtragem para diferentes visualizações de objetos. A introdução dessa funcionalidade permite inserir o indivíduo treinado no simulador virtual tático em diferentes lados de uma batalha e permite também a construção de um cenário virtual e tático baseado no ambiente construtivo. Todas essas abstrações fazem com que o SVTat REOP possa ser utilizado como uma extensão do SWORD e que em algum momento da execução do exercício, ele possa ser desconectado da federação e do modelo construtivo, para que então possa exercer suas próprias atividades e proporcionar um treinamento especializado a partir de uma situação construtiva.

Outro ganho muito claro para federação SIS-ASTROS fica por conta da questão de melhoria no processo de visualização através da transmissão de ordens. Um exemplo disso é o processo de disparo de munição, que engloba ações desde o desdobramento até a detonação da munição. Sem o SWAF, o SWORD publicaria apenas as interações de *WeaponFire* (na hora do tiro) e *MunitionDetonation* (efeitos do tiro). Para integração com outro simulador construtivo isso seria o suficiente. Porém, o SVTat REOP dispõe do *Wall 3D* e a realização de um disparo sem os ajustes necessário não seria algo desejável e acarretaria na perda de causalidade dos fatos. Dessa forma, as interações de *Deploy*,

que fornece ordem para desdobramento, e *SWAF\_FireAtLocation*, que fornece a ordem de tiro para os membros da federação (incluindo seu alvo e tipo de munição), torna possível a representação de todas as ações que antecedem a detonação, ou seja, garante a causalidade entre os eventos no ambiente 3D e proporciona visualização dos fatos também no ambiente de duas dimensões. Além da ordem de fogo, a ordem de movimentação, anteriormente mencionada, também representa uma forma de conseguir transmitir ações realizadas no construtivo e que precisam de uma representação no plano virtual. As ordens transmitidas podem ter diversas consequências para o plano virtual tático e a modelagem destas depende do que se espera da integração dos modelos.

Por fim, as interações criadas para garantir sincronização dos modelos permitem que os relógios dos simuladores trabalhem ao mesmo passo e que ajustes do próprio exercício no simulador virtual tático possam ser realizados através das informações transmitidas pelo construtivo. Um claro exemplo disso se reflete no horário de exercício que ao ser recebido pelo SVTat REOP, pode, em seu mundo virtual, proporcionar diferentes condições de iluminação com base na posição do sol para aquele horário, por exemplo. Mais detalhes relacionados a sincronização utilizando o SVTat REOP estão no trabalho de Oliveira (2017), que também utiliza o SWAF em seu estudo para suprimento de informações e detalha a reação no recebimento das interações implementadas aqui, além de outras técnicas aplicadas para garantia de consistência entre os modelos.

As próximas duas subseções fazem uma avaliação do método de extensão indireta do SOM. A subseção 5.3.1 discute os pontos positivos na utilização deste método e a subseção 5.3.2 apresenta os pontos desfavoráveis na adoção da metodologia de extensão indireta do SOM.

### **5.3.1 Prós da Metodologia de extensão indireta do SOM**

Em primeiro lugar, a adoção desta metodologia visou quebrar a barreira de software comercial que mantém o código-fonte fechado, evitando que desenvolvedores externos estendam as funcionalidades do sistema conforme o desejado. Por razões óbvias, as empresas que desenvolvem modelos de simulação comercial não expõem a modelagem e código-fonte do seu produto e quando mudanças ou adições se fazem necessárias, elas cobram pelo desenvolvimento de novas funcionalidades. Sendo assim, o primeiro ponto positivo da adoção desta metodologia é conseguir vencer a barreira comercial imposta pelas empresas de simulação, já que é possível estender as capacidades do simulador sem precisar alterar sua modelagem interna. Além de vencer essa barreira, é possível desenvolver as funcionalidades desejadas da forma que se espera, desde que o sistema de simulação possibilite isso. Isso quer dizer que qualquer entidade que queira interoperar com simuladores diversos que possibilitem a extensão indireta do seu SOM possam estender o SOM da maneira que entender, proporcionando uma independência neste de-

envolvimento.

Em relação a construção do SWAF e sua função de proporcionar informações adicionais para um ambiente multiresolução, há um ganho evidente em manter a generalidade do *plug-in* HLA do MASA SWORD. A adição de modelagens de classes de objetos e interações específicas para integração com um simulador virtual tático à interface HLA original do simulador construtivo estaria tirando a generalidade do seu *plug-in* HLA. O MASA SWORD utiliza FOMs de referência que proporcionam um alto nível de reusabilidade e interoperabilidade, logo não seria interessante adicionar módulos a esses acordos de federação para fins específicos, pois corresponderiam a modelagens não-generalistas. Isso estaria alterando as próprias características de modelagem construtiva da sua interface HLA.

Pode-se dizer que este trabalho não esperou a incorporação das suas funcionalidades no *plug-in* HLA original do simulador construtivo, pois não pretendeu alterar as características do simulador da MASA. A intenção, ao desenvolver este estudo, foi propor meios para que quando atuando de forma integrada com simuladores variados e que desejam informações que vão além de seu SOM, a metodologia de extensão indireta possa ser utilizada para obter essas informações e publicá-las. A característica de poder dividir o FOM em módulos foi um peça-chave que possibilitou a introdução dessa metodologia através da modelagem de classes de informações específicas em um módulo separado para ser utilizado pelo SWAF e demais federados interessados.

### 5.3.2 Contras da Metodologia de extensão indireta do SOM

O uso da metodologia de extensão indireta para extensão do SOM de um simulador, de certa forma, quebra o paradigma da HLA, que, embora não proíba sua adoção, por não especificar que um simulador só pode ser representado por um federado na federação, deixa a entender em sua normatização que cada federado trata-se de um simulador diferente ou interface para um simulador. Desta forma, a adoção de um federado auxiliar traz uma pequena mudança para o paradigma HLA.

É importante destacar que o uso desta metodologia não se aplica a todos os simuladores. Ela somente pode ser utilizada em simuladores que dispõem de modelagem semelhante ao MASA SWORD. No caso deste trabalho, explorou-se características de modelagem do SWORD para obter informações adicionais para extensão do SOM do simulador. No entanto, nem todos os simuladores dispõem de uma interface de rede para desenvolvimento de um cliente e nem todos armazenam informações de forma aberta em arquivos XML para que se possa criar um *parser* para filtragem de informações. Sendo assim, o uso da metodologia de extensão indireta é dependente das características do modelo de simulação, sua interface de comunicação (caso exista) e a forma com que se armazena informações de exercício ou de qualquer outra característica desejável. Ao desenvolver um *plug-in* HLA adicional para um simulador através desta metodologia, existe

uma clara limitação no que diz respeito as funcionalidades novas que podem ser desenvolvidas. Isso quer dizer que as funcionalidades podem ser estendidas de acordo com o que o simulador disponibiliza por qualquer meio. Já que não se tem acesso ao código-fonte do sistema, a extensão pode ser desenvolvida com base no tipo de informações que pode ser obter do simulador. Esta é a maior limitação no uso da metodologia de extensão indireta do SOM.

Ao fazer uso dessa metodologia, um ponto sobre o qual se deve ter atenção é o formato com que se recebe dados do modelo de simulação a ser estendido e envia informações para o RTI. Ao desenvolver um *plug-in* HLA voltado para interoperabilidade, é necessário respeitar as características para promover interoperabilidade, que exige uma padronização para transmissão de informações, como, por exemplo, o uso de um sistema de coordenadas padrão. Além disso, não é desejável que o *plug-in* HLA adicional possa interferir no funcionamento do *plug-in* HLA primário, e isso envolve diversos fatores, como a não publicação de informações duplicadas, de informações contraditórias ou que já são fornecidas através da interface HLA. O SWAF trabalha neste sentido em conjunto com o MASA SWORD recebendo suas entidades agregadas e relacionando estas com as unidades recebidas através da interface de rede. Tendo esta relação, é possível promover interações para tais unidades/entidades agregadas de maneira transparente sem que ocorram inconsistências. Além disso, o SWAF atua no sentido de fornecer interações para ações que o SWORD executa mas não possui representação para publicação através de seu *plug-in* HLA.

## 6 CONSIDERAÇÕES FINAIS

Esta dissertação trouxe como principal contribuição o *plug-in* HLA complementar para o SWORD. Essa solução denominada SWAF foi desenvolvida com base nos requisitos de interoperabilidade entre simulador construtivo e simulador virtual tático. Embora a extensão indireta do SOM do MASA SWORD tivesse sido construída com base nos requisitos identificados, o SWAF pode ser estendido para compreender requisitos de qualquer federação que utilize o SWORD como um dos seus federados e necessite de informações adicionais. Além disso, o *plug-in* HLA complementar se desenvolveu como uma forma de quebrar as dificuldades de integração com simuladores que possuem abstrações diferentes, como é o caso do modelo construtivo e do virtual tático utilizado neste trabalho. Neste sentido de complementação de informações para geração de um exercício integrado mais adequado, o SWAF proporcionou contribuições com o exercício em três sentidos: transferência situacional, transmissão de ordens e sincronização.

Durante a fase de pesquisa e revisão bibliográfica, foram encontrados alguns trabalhos acerca de desenvolvimento de federações que tratavam multiresolução, porém nenhum utiliza metodologia de extensão indireta do SOM do federado de forma a contornar tal problema. Desta forma, a solução desta dissertação se diferencia dos trabalhos relacionados envolvendo simulação distribuída em ambientes de multiresolução pela forma com que o SOM do federado foi estendido indiretamente. Além disso, o trabalho faz uma discussão sobre interoperabilidade em um ambiente envolvendo um modelo construtivo e um modelo virtual com características táticas, e especifica os requisitos de desenvolvimento do *plug-in* HLA complementar com base nessa discussão. Nesse sentido, a principal contribuição é dada para a identificação de elementos construtivos que pudessem ser importantes para o modelo virtual tático de simulação e que pudessem ser representados neste.

A utilização da metodologia de extensão indireta do SOM é avaliada na Seção 5.3 e seu prós e contras são discutidos em subseções dentro desta. Pode-se dizer que esta metodologia atende a demanda para o MASA SWORD, porém não é replicável para qualquer simulador. O mais importante a ser destacado é que a extensão indireta pode ser utilizada como forma de vencer a barreira de simuladores comerciais fechados que não implementam todas as funcionalidades desejadas através de sua interface HLA. Quando possível, sua adoção pode fornecer um ganho importante para a federação através da complementação de informações sobre determinado federado.

Quanto aos trabalhos relacionados que estão dentro do contexto da Federação SIS-ASTROS, juntamente com o trabalho apresentado nesta dissertação, eles demonstram grande importância para o desenvolvimento de tecnologias nacionais na área de defesa e de simulação. O desenvolvimento do *plug-in* HLA por Oliveira (2017), e o conversor de multiresolução de Paul (2017) são peças chave para a Federação SIS-ASTROS funcionar.

No entanto, sem o *plug-in* HLA complementar apresentado aqui, o problema identificado, ou seja, a falta de informações do modelo construtivo em contraste com os requisitos para integração com um simulador virtual tático, não seria superado e diversas ações e abstrações não poderiam ser representadas de forma adequada no simulador virtual. Portanto, pode-se dizer que a solução desenvolvida neste trabalho é fundamental para a federação, e serve como um complemento dos trabalhos citados acima. Todos os componentes da federação são importantes e ela não atingiria seu objetivo sem a presença de cada um deles.

O trabalho deixa claro que o SWAF é uma ferramenta que complementa o *plug-in* HLA original do MASA SWORD e é utilizado com o objetivo de fornecer informações que vão além da modelagem de sua interface HLA original. O estudo de caso serviu justamente para comprovar isso. No estudo de caso, o *plug-in* HLA complementar foi integrado à Federação SIS-ASTROS e, dessa forma, informações requisitadas puderam ser fornecidas para que o SVTat REOP pudesse trabalhar da maneira desejada a partir dessa extensão. As ações e representações desencadeadas no SVTat REOP a partir da publicação de informações adicionais com o SWAF na federação são desenvolvidas com base nos requisitos do seu projeto. Essas ações e abstrações são utilizadas como uma forma de provar que as funcionalidades previstas pelo SWAF estão sendo implementadas e que sua utilização compreende os requisitos de funcionamento esperados. Além disso, o desenvolvimento do *plug-in* HLA complementar não previu a volta de informações para o SWORD através do SWAF, visto que abordou a complementação de informações no sentido construtivo-virtual. Entretanto, isso apenas não foi implementado pela falta de requisitos, e pode ser desenvolvido através do cliente SWORD caso exista essa necessidade. Um exemplo de funcionalidade que poderia ser implementada é o envio de ordens recebidas via HLA para o SWORD através do SWAF para a realização das mais diversas tarefas, como o controle de pausa e aceleração da simulação, por exemplo.

Em relação aos trabalhos futuros que podem ser realizados no contexto do SWAF, além da implementação da volta de informação no sentido SWAF-SWORD, conforme citado no parágrafo anterior, espera-se que se possa desenvolver uma GUI para tornar o software mais amigável com menus de configuração para ferramenta. Entre as configurações poderiam ser definidos parâmetros em relação a conexão com o RTI, como a definição de endereço IP do *middleware*, a porta utilizada para comunicação, o nome da federação, o nome do federado, etc. Também existem diversas possibilidades para desenvolvimento de extensões que vão além das apresentadas neste trabalho, porém a adição de novas classes de objetos e de interações depende do contexto de utilização do SWAF, e deve ser realizada de acordo com requisitos de interoperabilidade da federação com que ele irá atuar.

## REFERÊNCIAS BIBLIOGRÁFICAS

BOWERS, A.; PROCHNOW, D. L. Multi-resolution modeling in the jtls-jcats federation. **Proceedings of the IEEE Fall Simulation Interoperability Workshop**, 2003.

CAYIRCI, E. Multi-resolution federations in support of operational and higher level combined/joint computer assisted exercises. In: WINTER SIMULATION CONFERENCE. **Winter Simulation Conference**. Austin, TX, USA, 2009. p. 1787–1797.

DOD, D. o. D. **DoD modeling and simulation (M&S) glossary**. Modeling and Simulation Coordinations Office, 1998. Acesso em outubro de 2017. Disponível em: <<https://biotech.law.lsu.edu/blaw/dodd/corres/pdf/500059m\0198/p500059m.pdf>>.

D'AMBROGIO, A.; GIANNI, D.; IAZEOLLA, G. Software technologies for the interoperability, reusability and adaptability of distributed simulators. In: SISO. **Proceedings of the 2007 European Simulation Interoperability Workshop**. Genoa, Italy, 2007.

FERREIRA, V. E. N. et al. **A simulação virtual tática no ensino e no treinamento militar**. DefesaNet, 2017. Acesso em novembro de 2017. Disponível em: <<http://www.defesanet.com.br/doutrina/noticia/26410/A-simulacao-virtual-tatica-no-ensino-e-no-treinamento-militar/>>.

FLETCHER, J. Education and training technology in the military. **science**, American Association for the Advancement of Science, v. 323, n. 5910, p. 72–75, 2009.

FRANCESCHINI, R. W.; PETTY, M. D. Linking constructive and virtual simulation in dis. **Distributed interactive simulation systems for simulation and training in the aerospace environment**, p. 281–298, 1995.

FUJIMOTO, R. M. Parallel and Distributed Simulation Systems. **John Wiley and Sons, Inc.**, 2000.

GOOGLE. **Protocol Buffers**. 2017. "<[www.developers.google.com/protocol-buffers/docs/overview](http://www.developers.google.com/protocol-buffers/docs/overview)>". Acesso em junho de 2017.

IEEE. Ieee standard for modeling and simulation (m&s) high level architecture (hla)– federate interface specification. **IEEE Std 1516.1-2010 (Revision of IEEE Std 1516.1-2000)**, p. 1–378, Aug 2010.

\_\_\_\_\_. Ieee standard for modeling and simulation (m&s) high level architecture (hla)– framework and rules. **IEEE Std 1516-2010 (Revision of IEEE Std 1516-2000)**, p. 1–38, Aug 2010.

\_\_\_\_\_. Ieee standard for modeling and simulation (m&s) high level architecture (hla)– object model template (omt) specification. **IEEE Std 1516.2-2010 (Revision of IEEE Std 1516.2-2000)**, p. 1–112, Aug 2010.

\_\_\_\_\_. Ieee standard for distributed interactive simulation–application protocols. **IEEE Std 1278.1-2012 (Revision of IEEE Std 1278.1-1995)**, p. 1–747, Dec 2012.

KARR, C.; ROOT, E. Integrating constructive and virtual simulations. In: **Proceedings of the 16th Interservice/Industry Training Systems and Education Conference**. Orlando, FL, USA: Defense Technical Information Center, 1994.

KWONG, K. et al. Disaggregation for realistic 3d visualisation. In: **I/ITSEC. Proceedings of the 2009 Interservice/Industry Training, Simulation and Education Conference**. Orlando, FL, USA, 2009.

MASA GROUP. **MASA Sword Simulator**: Home. MASA, 2017. Acesso em setembro de 2017. Disponível em: <<http://www.masagroup.net/>>.

MASA, M. G. Sword hla module - user manual. 2011.

\_\_\_\_\_. Masa sword network api documentation. Acessado em setembro de 2017. 2015.

\_\_\_\_\_. Sword gaming player guide. Acessado em outubro de 2017. 2015.

\_\_\_\_\_. Sword general user guide. Acessado em setembro de 2017. 2015.

MÖLLER, B.; LÖFSTRAND, B. Use cases for the hla evolved modular forms. In: **SIMULATION INTEROPERABILITY STANDARDS ORGANIZATION. Proceedings of 2007 Euro Simulation Interoperability Workshop**. Istanbul, Turkey, 2007.

NATO. **NATO Education and Training Network - Final Report of Task Group MSG-068**. 2012. [www.dtic.mil/get-tr-doc/pdf?AD=ADA560402](http://www.dtic.mil/get-tr-doc/pdf?AD=ADA560402). Acessado em setembro de 2017.

NICOL, J. **Fundamentals of Real-Time Distributed Simulation**. USA: Primal Soup Media, 2011. ISBN 9780986841408.

OLIVEIRA, V. D. de. **SVTat REOP HLA Plug-in: Estratégia de Sincronização de Dados, Eventos e Tempo para Simulação Distribuída**. 2017. Dissertação (Mestrado) — Programa de Pós-Graduação em Ciência da Computação - Universidade Federal de Santa Maria, Santa Maria/RS, Brasil, 2017.

PAUL, R. L. **Simulação Distribuída: Flexibilidade e Dinamismo no Tratamento Multi-resolução**. 2017. Dissertação (Mestrado) — Programa de Pós-Graduação em Ciência da Computação - Universidade Federal de Santa Maria, Santa Maria/RS, Brasil, 2017.

PAUL, R. L. et al. Doctrine based multi-resolution hla distributed simulation. In: **ACM. Proceedings of the Symposium on Applied Computing**. Marrakesh, Morocco, 2017. p. 59–64.

RABELO, L. et al. Multi Resolution Modeling. In: **Proceedings of the 2015 Winter Simulation Conference**. Piscataway, NJ, USA: IEEE Press, 2015. (WSC '15), p. 2523–2534. ISBN 978-1-4673-9741-4. Disponível em: <<http://dl.acm.org/citation.cfm?id=2888619.2888909>>.

RAUE, E.; GALLOIS, V. A dynamic multi-resolution model based on hla interconnection of commercial-off-the-shelf (cots) simulation tools. In: **NATO. NATO Modeling and Simulation Group (NMSG) Symposium**. Bern, Switzerland, 2011.

REYNOLDS, J. P. F.; NATRAJAN, A.; SRINIVASAN, S. Consistency maintenance in multi-resolution simulation. **ACM Transactions on Modeling and Computer Simulation (TOMACS)**, ACM, v. 7, n. 3, p. 368–392, 1997.

RIEGER, L.; BEEKER, E. Aggregation of entities for entity-aggregate simulation interoperability. In: **I/ITSEC. Proceedings of the Interservice/Industry Training Systems and Education Conference 2000**. Orlando, FL, USA: Defense Technical Information Center, 2000.

RUIZ, J. et al. Towards a new nato certification capability for hla interoperability. In: **SISO. Proceedings of the 2016 Simulation Innovation Workshop**. Orlando, FL, USA, 2016.

SCHRICKER, S. A.; BARBOSA, E. J.; SMITH, R. Enhancing the realism of pseudo-disaggregated entities. In: SISO. **Proceedings to the 8th Conference on Computer Generated Forces and Behavioral Representation**. Orlando, FL, USA, 1999.

SCHRICKER, S. A.; FRANCESCHINI, R.; ADKINS, M. Using pseudo-disaggregation to populate a large battlefield for sensor systems. In: SISO. **Paper of the 7th Conference on Computer Generated Forces and Behavioral Representation**. Orlando, FL, USA, 1998.

SISO. **Standard for Guidance, Rationale, and Interoperability Modalities for the Real-time Platform Reference Federation Object Model**. 2015. "[www.sisostds.org/productspublications/standards/sisostandards.aspx](http://www.sisostds.org/productspublications/standards/sisostandards.aspx)". Acessado em setembro de 2017.

STOBER, D. et al. Survey of constructive+virtual models. In: INSTITUTE FOR SIMULATION AND TRAINING. **Proceedings of the 5th Conference on Computer Generated Forces and Behavioral Representation**. Orlando, FL, USA, 1995. p. 93–102.

TOLK, A. **Engineering Principles of Combat Modeling and Distributed Simulation**. 1. ed. Hoboken, New Jersey, USA: John Wiley & Sons, Inc., 2012. ISBN 9781118180303.

ZABEK, A. et al. The army constructive training federation - multi-resolution modeling: The next generation of land component commander training at the unit of employment echelon of command. In: SISO. **Proceedings of the Fall 2004 Simulation Interoperability Workshop**. Orlando, FL, USA, 2004. v. 24.