



Francisco Vitor Gomes Salvador

Licenciado em Engenharia Informática

**Avaliação comparativa de formatos de dados
georreferenciados para troca de informação
entre sistemas de Comando e Controlo no
contexto militar**

Dissertação para obtenção do Grau de Mestre em
Engenharia Informática

Orientador : Doutor Miguel Carlos Pacheco Afonso Goulão, Prof.
Auxiliar, Universidade Nova de Lisboa

Co-orientador : Mestre Fernando Sérgio Bryton Dias Marques, Prof.
Assistente, Universidade Nova de Lisboa

Júri:

Presidente: Prof. Doutor Henrique João Lopes Domingues

Arguente: Prof. Doutor Bruno Emanuel da Graça Martins

Vogal: Prof. Doutor Miguel Carlos Pacheco Afonso Goulão



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Novembro, 2011

Avaliação comparativa de formatos de dados georreferenciados para troca de informação entre sistemas de Comando e Controlo no contexto militar

Copyright © Francisco Vitor Gomes Salvador, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

*À minha esposa, à minha filha, à minha mãe e ao meu falecido
pai.*

Agradecimentos

Uma dissertação de mestrado é um processo longo e cheio de desafios, por vezes difíceis, que obriga a muito esforço e dedicação. Durante este percurso, muitas pessoas colaboraram comigo, deram-me o seu apoio, incentivaram-me e acreditaram em mim e nas minhas capacidades.

Em primeiro lugar, agradeço ao meu orientador Professor Doutor Miguel Goulão, pela sua sábia orientação, grande perseverança, elevados conhecimentos, continuada dedicação, disponibilidade sem reservas e apoio incondicional durante toda a realização desta dissertação, essenciais para o seu sucesso.

Em segundo lugar, agradeço ao meu coorientador Mestre Fernando Sérgio Bryton Dias Marques como perito do domínio o saber que emprestou, como elemento de ligação à equipa da NATO os esforços que desenvolveu para estreitar relações e possibilitar contactos com o principal interessado nos resultados desta dissertação. Como Engenheiro de Software a sua experiência de desenvolvimento sistemas próximos daqueles estudados constitui uma enorme mais valia no desenvolvimento de todo o trabalho. Estas três valências foram realmente fundamentais para que este trabalho tenha sido concluído com sucesso. Quero agradecer-lhe ainda, todos os esforços e sacrifícios pessoais que fez para estar presente nos muito momentos que foi necessário dar o seu valioso contributo.

Agradecer ao Mark Postal, responsável pelo desenvolvimento do NVG na *NATO C3 Agency*, pela sua colaboração na avaliação e respetivo *feedback* à lista de características usada na avaliação qualitativa.

Agradecer ao Prof. Doutor João Moura Pires, pela disponibilidade manifestada na discussão acerca das métricas de XML a aplicar na avaliação quantitativa.

Agradeço ao Maj Inf Eng. Inf. Agostinho Valente e a todos os militares que desempenham funções no Instituto Geográfico do Exército, por todo o apoio e disponibilidade que manifestaram aquando da execução da experiência realizada na avaliação quantitativa e na impressão desta dissertação.

Agradeço ao Segundo Tenente Eng. Cabrita Branco da Marinha de Guerra Portuguesa pela disponibilidade manifestada e pelo apoio que prestou na fase inicial desta dissertação na instalação e experimentação de ferramentas SIG.

Deixo também uma palavra de agradecimento a todos os meus colegas e amigos com quem debati várias vezes os desafios com que me deparei e as soluções que foram adotadas, ao longo da realização deste trabalho.

Por último quero agradecer a toda a minha família, em especial à minha mulher e à minha sogra, pela sua ajuda e apoio inequívoco durante toda a minha fase académica e por sempre acreditarem em mim e nas minhas capacidades.

Muito obrigado a todos!

Resumo

A NATO (*North Atlantic Treaty Organization*) desenvolveu recentemente um novo formato de dados georreferenciados *NATO Vector Graphics* (NVG), para codificar informação georreferenciada enviada entre sistemas de comando e controlo. Existem vários formatos de dados georreferenciados que poderiam ser, do ponto de vista técnico, uma opção credível para a NATO, nomeadamente, formatos abertos da W3C (*World Wide Web Consortium*) e da OGC (*Open Geospatial Consortium*). Contudo, por razões de complexidade ou aparente inadaptação ao contexto militar, a NATO decidiu pelo desenvolvimento de um novo formato de dados georreferenciados.

Não se conhecem quaisquer estudos que envolvam uma avaliação comparativa do NVG com outros formatos alternativos que permitam fazer uma análise fundamentada acerca das vantagens e desvantagens do NVG em relação aos restantes.

Para se perceber se existem alternativas melhores ao formato NVG no contexto militar, propõe-se nesta dissertação realizar uma avaliação comparativa do formato NVG com os formatos de dados alternativos. Esta avaliação será alicerçada num estudo qualitativo das características dos formatos de dados NVG, SVG, KML e NVG, e numa análise quantitativa, realizada a partir de um simulador de troca de informação georreferenciada entre sistema de comando e controlo, desenvolvido no âmbito deste trabalho, que permita comparar os formatos de dados NVG, SVG e KML com recurso a métricas apropriadas.

Os principais contributos desta dissertação são: i) Comparação qualitativa dos formatos em avaliação, resultante de um estudo sistemático da especificação de cada um deles, segundo os mesmos critérios; ii) Desenho e implementação de um simulador de cenários militares; iii) Seleção de métricas para a avaliação dos vários formatos e implementação de um sistema para a recolha automática das referidas métricas; iv) Comparação quantitativa entre os diversos formatos de dados georreferenciados, num cenário militar.

Palavras-chave: Formato de dados Georreferenciados, NVG, KML, SVG, GML, Análise

x

Comparativa

Abstract

NATO (*North Atlantic Treaty Organization*) has recently developed a new georeferenced data format, NATO Vector Graphics (NVG), to encode geo-referenced information sent between command and control systems. There are several geo-referenced data formats currently available in the market, which could have, from a technical point of view, been a credible option for NATO, specifically, open formats such W3C (*World Wide Web Consortium*) or the OGC (*Open Geospatial Consortium*). Nevertheless, whether it was due to the unavailability of some of the other formats during the initial development of NVG, or because of their complexity or inappropriateness in a military context, NATO decided to invest in the development of a new geo-referenced data format.

There are no known comparative NVG studies against other format alternatives which would permit a substantiated analysis with respect to the advantages and disadvantages of using NVG in comparison with the others.

In order to understand if there are better alternatives to NVG in military context, I propose in this dissertation to carry out a comparative evaluation of the NVG format against the following formats: *Scalable Vector Graphics* (SVG), *Keyhole Markup Language* (KML) and *Geography Markup Language* (GML). This evaluation will be based on a qualitative study of the different format characteristics, and a quantitative analysis, performed from a simulator which will exchange georeferenced information between command control systems, developed in the scope of this work, which will allow to compare the various formats with appropriate metrics.

The principals contribute of this dissertation are: i) qualitative comparison of the formats in assessment resulting from a systematic study of each specification, using the same criteria, ii) Design and implementation of a military scenarios simulator iii) Selection of metrics for the evaluation of various data formats and implementation of a system for automatic collection of such metric; iv) quantitative comparison between the various georeferenced data formats in a military scenario.

Keywords: Georeferenced Data Format, NVG, KML, SVG, GML, Comparativa Analysis

Conteúdo

1	Introdução	1
1.1	Motivação	1
1.2	Descrição do problema	2
1.2.1	NATO e novas tecnologias	2
1.2.2	Sistemas de Comando e Controlo	3
1.2.3	Questões de investigação	5
1.3	Metodologia	6
1.4	Principais contribuições	11
2	Enquadramento	13
2.1	Cenário	13
2.1.1	Medidas de Coordenação	14
2.1.2	Força	14
2.1.3	Evolução do Cenário	16
2.1.4	Batalha de al-Khafji	17
2.2	Estratégias de Comunicação	17
2.2.1	<i>Request-Response</i>	18
2.2.2	<i>Publish-Subscribe</i>	19
2.2.3	Mapeamento de cenários em ficheiros	19
2.3	Cenários com e sem Medidas de Coordenação	20
2.4	Formatos de Dados e Critérios de Elegibilidade	21
2.5	Formatos de dados selecionados	24
2.5.1	<i>NATO Vector Graphics</i> [NC310b] (NVG)	24
2.5.2	<i>Scalable Vector Graphics</i> [Gro09] (SVG)	25
2.5.3	<i>Keyhole Markup Language</i> [Wil08] (KML)	26
2.5.4	<i>Geography Markup Language</i> [Gro07] (GML)	28
3	Trabalho Relacionado	31
3.1	Avaliação qualitativa de formatos de dados georreferenciados	31

3.2	Avaliação quantitativa de formatos de dados	33
3.3	Métricas de XML	34
4	Avaliação Qualitativa	37
4.1	Introdução	37
4.2	Exemplo prático	37
4.2.1	NATO Vector Graphics[NC310b]	39
4.2.2	Scalable Vector Graphics	42
4.2.3	KML	46
4.2.4	GML[Gro07]	51
4.3	Comparação qualitativa dos formatos	57
4.3.1	Introdução	57
4.3.2	Análise das características	58
4.4	Desafios em aberto	65
5	Avaliação Quantitativa	67
5.1	Introdução	67
5.2	Planeamento da experiência	68
5.2.1	Objetivos	68
5.2.2	Unidades Experimentais	68
5.2.3	Ferramentas de Apoio	68
5.2.4	Tarefas a realizar no processo de partilha ficheiros	74
5.2.5	Hipóteses, parâmetros e Variáveis	75
5.2.6	Desenho	79
5.2.7	Ambiente de realização da experiência	80
5.2.8	Processo	80
5.3	Resultados	82
5.3.1	Resultados de métricas de dimensão	83
5.3.2	Resultados de métricas de <i>performance</i>	93
5.3.3	Tempo de transmissão teórico	98
5.4	Discussão	100
5.4.1	Resultados da experiência	100
5.4.2	Desempenho dos formatos em relação ao espaço	100
5.4.3	Desempenho dos formatos em relação ao tempo	103
5.4.4	Discussão sobre a tempo de transmissão	105
5.4.5	Ameaças à validação	106
6	Conclusões e Trabalho Futuro	109
6.1	Conclusões	109
6.1.1	A avaliação qualitativa	110
6.1.2	A avaliação quantitativa	111
6.1.3	Avaliação Global	112

6.2 Trabalho futuro	112
A Anexo A - Tabela de resultados	121

Lista de Figuras

1.1	Exemplo da diversidade de tipos de forças no campo de batalha moderno	4
2.1	Meta-modelo de Classes do Cenário.	14
2.2	Transparente de Operações de uma operação de guerra convencional. . .	15
2.3	Linha de Fase.	16
2.4	Batalhão de Infantaria Mecanizado.	16
2.5	Transparente de operações da operação "Liberation of al-Khafji".[Wes08] .	18
2.6	Diagramas ilustrativos da codificação da informação presente num cenário nos diferentes padrões de comunicação.	20
2.7	Código com a representação de um Posto de Comando em formato NVG.	24
2.8	Código com a representação de um Posto Comando em formato SVG. . .	25
2.9	Código com a codificação de um Posto Comando em formato KML.	27
2.10	Código com a representação de um Posto de Comando, em formato GML.	29
2.11	Ficheiro de estilos no formato SLD para a aplicar aos dados no formato GML na sua visualização.	29
3.1	Métricas de XML apresentadas em [RL05].	36
4.1	Cenário utilizado para o caso de estudo	38
4.2	Cabeçalho do <i>XML Schema</i> do NVG	41
4.3	Código exemplificativo da definição de um Símbolo	43
4.4	Visualização do processamento do código 4.3	44
4.5	Definição do elemento <i>Path</i> e a sua respectiva visualização	44
4.6	Exemplo da utilização de RDF num ficheiro SVG	45
4.7	Exemplo da utilização de um recurso exterior ao ficheiro SVG	45
4.8	Exemplo da estrutura do elemento <i>kml:placemark</i>	51
4.9	GML como meta-linguagem na definição de esquemas de domínio de aplicação.	52
4.10	<i>XML Schema</i> definido para um domínio de aplicação concreto.	53

4.11	Exemplo do formato GML onde se representam ruas de uma cidade. . . .	53
5.1	Arquitetura da aplicação em Java, <i>Scenario Simulator</i>	69
5.2	Modelo Relacional da Base de Dados <i>ScenarioDB</i>	72
5.3	Modelo de partilha de um ficheiro XML do Produtor para o Consumidor.	74
5.4	Diagrama de aplicação da transformação XSL <i>Strip-space</i>	77
5.5	Diagrama da aplicação da operação de compressão <i>Zip Compression</i>	77
5.6	Diagrama da aplicação da operação de descompressão <i>UnZip Decompression</i>	78
5.7	Diagrama da aplicação da operação de validação XML <i>Validation</i>	78
5.8	Quadro resumo da aplicação das métricas.	78
5.9	Diagrama de Atividades da avaliação quantitativa.	83
5.10	Métricas de dimensão aplicadas a cenários com 333 elementos e sem Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	84
5.11	Métricas de dimensão aplicadas a cenários com 333 elementos e sem Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	84
5.12	Métricas de dimensão aplicadas a cenários com 342 elementos e com Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	85
5.13	Métricas de dimensão aplicadas a cenários com 342 elementos e com Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	85
5.14	Métricas de dimensão aplicadas a cenários com 64 atualizações e sem Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	86
5.15	Métricas de dimensão aplicadas a cenários com 64 atualizações e sem Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	86
5.16	Métricas de dimensão aplicadas a cenários com 64 atualizações e com Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	87
5.17	Métricas de dimensão aplicadas a cenários com 64 atualizações e com Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	87
5.18	Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados (KML em cima, NVG ao centro, SVG em baixo), sem Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	88
5.19	Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados (KML em cima, NVG ao centro, SVG em baixo), sem Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	89
5.20	Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados, com Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	90

5.21	Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados (KML em cima, NVG ao centro, SVG em baixo), com Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	90
5.22	Peso de cada formato de dados nas várias métrica de dimensão aplicadas a cenários com 64 atualizações e sem Medidas de Coordenação	91
5.23	Peso de cada formato de dados nas várias métrica de dimensão aplicadas a cenários com 64 atualizações e com Medidas de Coordenação	92
5.24	Métricas de <i>performance</i> aplicadas às operações de <i>Strip-spaces</i> , <i>Zip</i> , <i>UnZip</i> e <i>Validation</i> em cenários sem Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	93
5.25	Métricas de <i>performance</i> aplicadas às operações de <i>Strip-spaces</i> , <i>Zip</i> , <i>UnZip</i> e <i>Validation</i> em cenários sem Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	94
5.26	Métricas de <i>performance</i> aplicadas às operações de <i>Strip-spaces</i> , <i>Zip</i> , <i>UnZip</i> e <i>Validation</i> em cenários com Medidas de Coordenação para o padrão de comunicação <i>Publish-Subscribe</i>	94
5.27	Métricas de <i>performance</i> aplicadas às operações de <i>Strip-spaces</i> , <i>Zip</i> , <i>UnZip</i> e <i>Validation</i> em cenários com Medidas de Coordenação para o padrão de comunicação <i>Request-Response</i>	95
5.28	Distribuição do tempo consumido, em %, em cada operação por formato de dados.	95
5.29	Impacto da dimensão do ficheiro, dos diferentes formatos, nas operações de: <i>Strip-space</i> , <i>Zip</i> , <i>UnZip</i> e <i>Validation</i>	97
5.30	Variação das operações de <i>Strip-space</i> e <i>Validation</i> ao longo do aumento da dimensão dos ficheiros.	97
5.31	Variação das operações de <i>Zip</i> e <i>UnZip</i> ao longo do aumento da dimensão dos ficheiros.	98
5.32	Imagem do cenário c1 visualizado no Google Earth.	99
5.33	Tempo de transferência do cenário C1, para Largura de Banda diferentes.	106
A.1	Tabela com os resultados.	122
A.2	Tabela com os resultados (continuação).	123
A.3	Tabela com os resultados (continuação).	124
A.4	Tabela com os resultados (continuação).	125
A.5	Tabela com os resultados (continuação).	126
A.6	Tabela com os resultados (continuação).	127
A.7	Tabela com os resultados (continuação).	128
A.8	Tabela com os resultados (continuação).	129
A.9	Tabela com os resultados (continuação).	130
A.10	Tabela com os resultados (continuação).	131

A.11 Tabela com os resultados (continuação).	132
A.12 Tabela com os resultados (continuação).	133
A.13 Tabela com os resultados (continuação).	134
A.14 Tabela com os resultados (continuação).	135
A.15 Tabela com os resultados (continuação).	136
A.16 Tabela com os resultados (continuação).	137
A.17 Tabela com os resultados (continuação).	138

Lista de Tabelas

2.1	CrITÉrios de elegibilidade	22
2.2	Grelha de seleÇo dos formatos de dados	23
3.1	Documentos com mÉtricas para XML.	35
4.1	<i>XML Schema</i> do formato de dados NVG.	39
4.2	<i>XML Schema</i> do formato de dados SVG[Gro09].	42
4.3	<i>XML Schema</i> do formato de dados.[OGC08]	46
4.4	<i>Namespace</i> e os <i>Schemas</i> do formato de dados GML	51
4.5	Nmero de primitivas geomÉtricas por formato.	61
4.6	Matriz Resumo das CaracterÍsticas	64
5.1	Escales permitidos pelo <i>Scenario Generator</i>	69
5.2	Tecnologia e pacotes Java utilizados.	73
5.3	VariÁveis da experincia e respetivos valores testados.	79
5.4	Detalhes de configuraÇo dos recursos utilizados	80
5.5	ConfiguraÇo da estrutura dos elementos do Tipo Força	81
5.6	ConfiguraÇo das Medidas de CoordenaÇo	81
5.7	ConfiguraÇo <i>Scenario Genarator</i>	81
5.8	Tamanho dos ficheiros em cada formato, referente ao cenÁrio c1.	99
5.9	Tempo de transferncia do cenÁrio XPTO, para Largura de Banda diferentes.	99



Introdução

1.1 Motivação

A *North Atlantic Treaty Organization*¹ (NATO) está num período de transformação interna [webd], integrado no esforço das melhorias das suas capacidades. O programa *NATO Network Enabled Capability*[webe] (NNEC), em particular, visa fomentar a melhoria das capacidades tecnológicas no seio da aliança atlântica. É no contexto desse programa que a NATO desenvolveu um novo formato de dados, *NATO Vector Graphics*[NC310b] (NVG), para transferir informação georreferenciada entre sistemas de Comando e Controlo.

Existem vários formatos de dados georreferenciados, bastante divulgados e utilizados, que poderiam ser, do ponto de vista técnico, uma opção credível para a NATO, nomeadamente formatos abertos da *World Wide Web Consortium*² (W3C) e *Open Geospatial Consortium*³ (OGC). Contudo, a NATO optou pelo desenvolvimento de um novo formato de dados, o NVG.

Não se conhecendo qualquer estudo que sustente os benefícios e limitações deste formato e perante a diversidade de alternativas disponíveis, propõe-se nesta dissertação realizar uma avaliação comparativa do formato NVG com os seguintes formatos: o *Scalable Vector Graphics*⁴ (SVG), o *Keyhole Markup Language*⁵ (KML), e o *Geography Markup Language*⁶ (GML); através de um estudo qualitativo das características dos diferentes formatos, e de uma análise quantitativa realizada a partir de simulador de cenários realista

¹http://www.nato.int/cps/en/SID-F7AE6C8A-01A8B373/natolive/what_is_nato.htm

²<http://www.w3.org/>

³<http://www.opengeospatial.org/>

⁴<http://www.w3.org/Graphics/SVG/>

⁵<http://www.opengeospatial.org/standards/kml/>

⁶<http://www.opengeospatial.org/standards/gml>

com recurso a métricas de dimensão e performance.

Deste modo os principais contributos desta dissertação serão: (i) uma avaliação comparativa (qualitativa e quantitativa) dos vários formatos de dados georreferenciados; (ii) a especificação e implementação de um simulador de cenários realista do contexto militar. e (iii) a criação dum caso de estudo realista que permita comparar os diversos formatos.

1.2 Descrição do problema

1.2.1 NATO e novas tecnologias

Para fazer face aos desafios dos campos de batalha modernos, a NATO desenvolveu o programa *NATO Network Enabled Capability* (NNEC), que visa refletir a capacidade da NATO para certificar vários recursos, materiais e humanos, através de uma infraestrutura de informação [webd]. Este programa tem com principal objetivo a partilha de informação entre os países membros. É filosofia da NATO que a partilha de informação permitirá um melhor conhecimento da situação e tomadas de decisão mais rápidas que poderão salvar vidas e otimizar a utilização de recursos.

Um dos objectivos da NATO é atingir níveis de desempenho mais elevados. Esse desempenho resulta da existência de certas capacidades na NATO, a serem definidas através dos seguintes elementos: "Doctrine", "Organization", "Training", "Materiel", "Leadership development", "Personnel", "Facilities" e "Interoperability"[weba] (DOTMLPFI).

As novas tecnologias assumem um papel fundamental nessa transformação. A NATO preconiza desenvolver as suas capacidade ao nível das seguintes tecnologias: *Service-oriented architecture*⁷ (SOA), *Event-driven architecture* (EDA) e *Semantic Web*.

SOA é uma arquitetura orientada aos Serviços[web08], onde tudo são serviços. Serviços são módulos de negócio ou funcionalidades das aplicações que disponibilizam interfaces, e que são invocados via mensagens. Em SOA, recursos de *software* são empacotados como serviços bem definidos, auto-contidos, oferecendo funcionalidades padrões do negócio, independentes do estado ou contexto de outros serviços. Os serviços são disponibilizados através das suas interfaces, que o possibilita que o cliente do serviço tenha acesso ao serviço através da sua interface, mas não tenha acesso direto à implementação do serviço[CF09].

EDA é uma arquitetura voltada para os eventos(coisa notável que acontece dentro ou fora do nosso negócio que pode significar um problema para resolver ou uma oportunidade para fazer algo), na qual, quando um evento acontece, a arquitetura encarrega-se de o disseminar por todos os interessados (humanos ou autómatos). As partes interessadas avaliam o evento e podem ou não tomar uma dada ação. Uma ação em EDA pode ser uma invocação de um serviço, o desencadeamento de um processo de negócio e/ou eventos [BMMC06].

A *Web Semântica* é uma visão definida pelo inventor da *World Wide Web* (W3), *Tim*

⁷<http://www.soa-consortium.org/>

Berners-Lee que define a Web Semântica como "uma rede de dados que podem ser processados direta e indiretamente por máquinas." [BLHL01]. Hoje em dia existem diversos esforços no desenvolvimento deste novo conceito, destacamos o elevado número de grupos constituído para trabalhar no assunto⁸. No futuro, a *Web Semântica* permitirá aos sistemas inferir informação implícita acerca dos dados existentes na rede sem necessidade de interferência humana.

Foi neste contexto de transformação, que surgiu a necessidade de representação da informação militar georreferenciada contida nos diversos Transparentes de Operações (informação graficada em finas telas transparentes sobre uma carta militar) num formato digital, que permitisse, a troca de informação entre sistemas de Comando e Controlo (C2) [NC310a] . Para colmatar esta necessidade foi desenvolvido o formato de dados, em XML, designado de NATO Vector Graphics (NVG).

1.2.2 Sistemas de Comando e Controlo

Um sistema de C2 é um conjunto de equipamentos, métodos e procedimentos e, se necessário, pessoal, que permite que os comandantes e suas equipas para o exercício comando e controlo [(NS07)]. Os sistemas de C2 têm por objetivo otimizar o exercício da direção, do controlo e da coordenação das forças militares, com vista ao cumprimento de uma missão. Sempre que operam no cumprimento de uma missão, as forças militares estão subordinadas a um comandante, que exerce o comando e controlo dessas forças através de um sistema de C2 implementado para o efeito.

As unidades militares estão organizadas segundo uma estrutura hierárquica, dando origem a diferentes escalões de unidades. As unidades nos escalões mais elevados, designadas por Grandes Unidades têm na sua estrutura vários tipos de forças, como por exemplo: forças de manobra, forças de apoio de fogos, forças de apoio de serviços, etc; todas estas diferentes forças no terreno articulam-se por forma a que cada uma cumpra a missão que lhe foi incumbida e, no conjunto, a grande unidade de que fazem parte tenha sucesso (ver Figura 1.1). Para que tudo isto seja possível, são necessários sistemas de Comando e Controlo que permitam a partilha de informação entre todas as unidades.

Uma operação militar é na sua essência um sistema distribuído e heterogéneo de sistemas onde cada um tem uma visão parcial do conjunto que é necessário integrar, e onde o facto da existência de Forças Militares, que integram unidades pertencentes a diferentes países sob a égide da NATO, coloca um verdadeiro desafio à capacidade de interoperabilidade dos diferentes meios tecnológicos que atuam em conjunto e necessitam de partilhar informação no contexto militar.

Para ajudar a perceber o conteúdo desta dissertação existem dois conceitos essenciais a perceber: a) **Informação no Contexto Militar**, é informação cujo o objetivo é representar o Campo de Batalha, recorrendo a símbolos normalizados, que permitem a sua

⁸<http://www.w3.org/2001/sw/#publs>

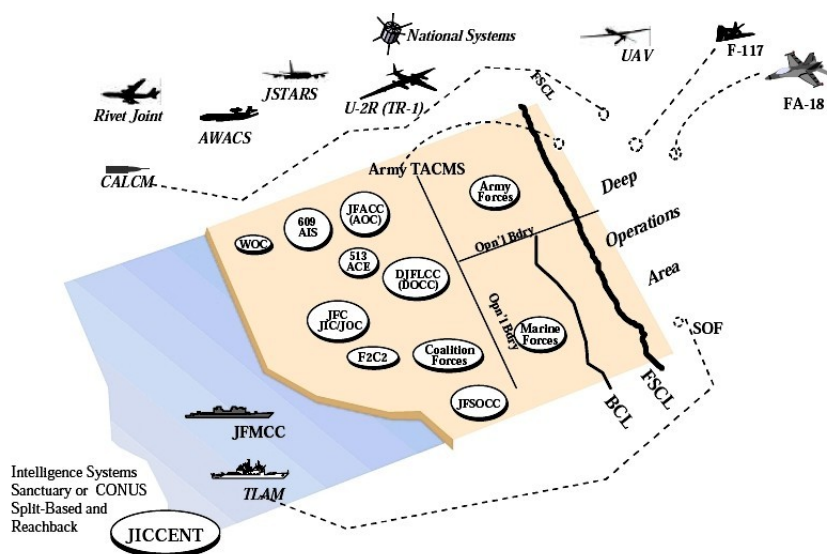


Figura 1.1: Exemplo da diversidade de tipos de forças no campo de batalha moderno [oD97]

visualização georreferenciada; b) **Simbologia Militar**, são símbolos gráficos (de duas dimensões e multicolor) normalizados, que representam informação no contexto militar de forma clara, precisa e concisa, diminuindo ao máximo a possibilidade de ambiguidades na interpretação da mesma.

Os conceitos que estão na base do Comando e Controlo têm vindo a evoluir ao longo dos tempos. Inicialmente, começaram por denominar-se "Command and Control"(C2), mas actualmente a sua designação NATO é, "Command, Control, Computers, Communication, Intelligence, Surveillance and Reconnaissance" (C4ISR)[NC310a]. Nesta dissertação, por simplicidade, vamos manter a designação inicial C2 para nos referirmos a estes conceitos.

Os sistemas C2 que dão suporte a este tipo de conceitos são extremamente complexos e respondem a requisitos cada vez mais exigentes a diversos níveis. A utilização de informação georreferenciada é vista como essencial para o cumprimento da missão, por conseguinte a partilha eficiente da mesma é um fator decisivo na consecução da vitória, pelo que a utilização de um bom formato de dados se revele de grande importância.

Informação georreferenciada é o resultado da associação efetuada entre uma dada informação e uma localização conhecida junto da superfície terrestre, num Sistema de Coordenadas de Referência.

É no **contexto de transformação da NATO**, que se abriu a oportunidade para participar nesse esforço, fazendo desta dissertação um possível contributo. Tendo conhecimento que ao nível da NATO, se desenvolveu **um novo formato de dados, NVG, para troca de informação entre Sistemas de Comando e Controlo**, importa conhecer as vantagens e desvantagens deste novo formato relativamente aos formatos já existentes e amplamente utilizados noutros contextos.

1.2.3 Questões de investigação

Existem no mercado várias soluções alternativas a este novo formato, que se encontram em uso com êxito, já há algum tempo, o que lhes confere um estado de maturação que permite, à partida, oferecer garantias que um novo formato, não pode oferecer. Os formatos alternativos oferecem ainda, a vantagem de serem *standards* e de terem, por isso, uma comunidade interessada em continuar o seu desenvolvimento e melhoramento.

Sabe-se que⁹, antes de se iniciar o processo de implementação do NVG, foram identificados dois potenciais formatos, o GML e o SVG. No entanto, ambos os formatos foram abandonados aparentemente pelas seguintes razões:

- Os formatos da OGC não endereçavam a simbologia militar de forma adequada;
- Utilizar a Linguagem de estilos da OGC, *Styled Layer Descriptor*[AC02] (SLD) para endereçar simbologia militar foi considerada uma opção muito complexa;
- Nos standards da OGC, cada tipo de item é tratado numa *layer*¹⁰ diferenciada, o que implica uma sobrecarga, visto que, para compor uma visualização é necessário sobrepor diferentes *layers*, o que pode ser computacionalmente muito pesado, para os recursos, tipicamente disponíveis em sistemas C2.
- A utilização do formato de dados GML, foi equacionada mas esta opção requeria a construção de um “transformador” (designado por *Feature Portrayal Service*) para um formato que pudesse ser visualizado, o que não apresentava grandes vantagens.
- SVG era um formato pouco adequado para a troca de informação georreferenciada entre sistemas de comando e controlo.

Não nos propomos avaliar a decisão da NATO, mas sim estudar opções e obter elementos que habilitem informar uma opinião acerca do NVG, comparativamente aos formatos alternativos. Nesse intuito, foram levantadas **duas grandes questões de investigação** que servirão de base a esta dissertação:

QI1 *Quais as diferenças entre o formato de dados NVG e os formatos de dados alternativos na partilha de informação georreferenciada entre sistemas de C2?*

QI2 *O formato de dados NVG permite melhor desempenho que os formatos de dados alternativos na partilha de informação georreferenciada entre sistemas de C2?.*

⁹Informação obtida a partir de contactos estabelecidos com os autores do NVG.

¹⁰No contexto desta dissertação, define-se *layer* como uma camada de informação que tem uma parte específica da imagem, que sobreposta com outras *layers* dará origem à imagem final.

1.3 Metodologia

Não sendo conhecidos, à data, outros estudos comparativos do formato NVG com outros formatos de dados, que responda às questões de investigação levantadas na secção anterior, propõe-se, com esta dissertação a realização de uma avaliação qualitativa ao nível das suas características explícitas na especificação de cada formato de dados para dar resposta à primeira questão de investigação, e uma avaliação quantitativa, comparando o NVG, num cenário realista, com os formatos dados que se assumam como alternativas credíveis para dar resposta à segunda questão de investigação. Os formatos de dados seleccionados para a referida comparação foram: NVG, SVG, KML e GML.

A sobreposição dos formatos de dados não é total no que diz respeito às suas características, objetivos e casos de uso. Entendeu-se que seria vantajoso para o estudo uma comparação que envolvesse formatos de dados onde fosse possível explorar todas as características que um possível formato alternativo possa conter, nomeadamente ao nível da representação e da visualização da informação georreferenciada.

A avaliação vai ser qualitativa, baseada nas características de cada formato, e quantitativa, contrastando os diferentes formatos num cenário realista.

No primeiro método, recorrer-se-á a uma análise sistemática das especificações dos diferentes formatos de dados seleccionados para a avaliação, por forma, a permitir uma avaliação comparativa que nos permita dar resposta à primeira questão de investigação (subsecção 1.2.3), segundo características consideradas importantes na codificação da informação georreferenciada no contexto da troca de informação entre sistemas de C2. Estas características são o resultado da nossa avaliação. Uma lista preliminar [Sal11] foi submetida ao escrutínio por parte de elementos da NATO, e o *feedback* recebido foi, por conseguinte, incorporado na **lista de caraterísticas** que aqui se apresentada. As características que importa analisar são as seguintes:

- Uso de *XML Schema* para definição do documento

Quando estamos a falar de troca de dados em XML[W3C] a linguagem de definição do próprios dados revela-se como um fator importante. Para o nosso estudo consideramos ser diferenciador o facto dos formatos de dados serem ou não definidos através da linguagem *XML Schema*. O uso da linguagem *XML Schema* é importante porque: a) a linguagem *XML Schema* que é a recomendação da W3C para a definição de documentos XML; b) a linguagem *XML Schema* é ela própria definida em XML e oferece um modelo de descrição mais rico e expressivo que o *Document Type Definitions*, permite mecanismos de restrições e herança, capacidade para lidar com vários *namespaces*, suporte para vários tipo de dados primitivos e definição de tipos de dados pelo utilizador que podem derivar de outros tipos de dados já existentes, tornam a *XML Schema*, segundo [Cag01], numa linguagem poderosa e flexível para expressar conteúdo permissível de documentos da classe XML.

- Sistema de Coordenadas de Referência optativo

Para a troca de informação geográfica é importante saber a que Sistemas de Coordenadas de Referência¹¹ (SCR) ela está associada. Existem diversos SCR definidos para georreferenciar dados, visto que, cada SCR é definido no contexto de um dado *Datum* (providencia um modelo matemático da terra, normalmente assente num elipsoide) importa saber qual a versatilidade dos diferentes formatos em possibilitar o uso de diferentes SCR, o que lhes permitirá maior mapeamento de informação proveniente de qualquer SCR, e ainda, de permitir ao utilizador escolher o SCR que mais lhe interesse; em oposição de um SCR pré-definido no formato.

- Utilização da terceira dimensão na localização de um ponto

Embora, tanto quanto se apurou, não exista nenhuma especificação definida pela NATO para integrar a terceira dimensão na visualização da sua simbologia militar, a utilização da terceira dimensão para a especificação de uma localização é importante, dado que nem todas as entidades ou eventos que queremos representar acontecem ao nível da superfície terrestre. Por exemplo, colocar uma aeronave no solo pousada ou no ar em movimento, embora a sua projeção na superfície terrestre seja a mesma, faz muita diferença. A utilização da terceira dimensão aumenta a expressividade do formato ao nível da localização, visto que permite discriminar duas entidades com o mesmo par de coordenadas (x,y), mas com altitudes diferentes. Esta situação, tem ainda maior relevo nos objetos existentes no Campo de Batalha que necessitam, de mais do que um simples ponto para a sua localização ser definida, um bom exemplo são os Corredores Aéreos¹².

- Suportar a utilização do Sistema Internacional de Unidades¹³

A utilização de unidades por parte dos formatos de dados pertencentes ao Sistema Internacional de Unidades (SI), permite usufruir do esforço realizado pelo *Bureau International des Poids et Mesures*¹⁴, em normalizar as unidade de medida para serem utilizados ao nível mundial, não obstante a existência de países que mantêm os seus próprios Sistema de Medidas, como é o caso dos Estado Unidos da América (*United States customary units*). A utilização do SI por parte dos formatos de dados é importante, visto que, a sua não utilização pode acarretar dificuldades de compreensão por parte do utilizador final dos seus dados, ou então acarreta um esforço extra de conversão das mesmas para o SI.

- Definição de Estilos

A definição de estilos (informação acessória, que serve para especificar como os dados serão visualizados) num formato de dados, não sendo uma característica crítica para a troca de informação entre os sistemas C2, visto que pode ser calculada

¹¹<http://www.isotc211.org/>

¹²Definem um volume de espaço, à semelhança de um paralelepípedo, destinado a passagem de aeronaves específicas, a sua utilização é restrita e necessita de ser coordenada

¹³*International System of Units*, é um documento publicado pelo *Bureau International des Poids et Mesures*

¹⁴<http://www.bipm.org/>

a partir de outros atributos pelo consumidor, é importante para se perceber até que ponto esse formato separa ou não a definição dos estilos da própria definição dos dados. Saber como é que a definição dos estilos é realizada, se é necessário definir um estilo por elemento ou se um mesmo estilo pode ser compartilhado por vários elementos. Estas diferentes abordagens têm vantagens e desvantagens[webb], logo, importa aqui analisar se o formato permite ou não a definição de estilos e de que forma o realiza.

- Uso de Simbologia Militar¹⁵

A utilização de simbologia militar é importantíssima no contexto desta avaliação, dado que ela permite uma grande expressividade no contexto militar, a sua utilização evita ambiguidades que poderiam ser fatais para as forças no terreno, e porque permitem uma grande simplicidade na especificação do formato de dados, logo, interessa saber se a linguagem na sua gênese prevê a utilização de simbologia militar. Ou seja, se dispõe de mecanismos criados nesse sentido. Qual a capacidade desses mecanismos, no endereçamento de simbologia multi-ponto, na utilização de diferente tipo simbologia e na possibilidade de integrarem novos símbolos.

- Elementos do tipo geométrico

Dado que, neste trabalho estamos a comparar a capacidade de troca de informação para ser visualizada geograficamente (a informação no contexto militar assenta principalmente no uso de símbolos militares), é absolutamente relevante analisar a capacidade de cada formato nesta área. Todos os formatos de dados têm as suas primitivas geométricas para descrever a geometria dos objetos ou entidades existentes na realidade. Assim sendo, interessa dar a conhecer quais são essas primitivas e compara-las de um ponto de vista qualitativo.

- Flexibilidade na definição dos dados

A definição dos dados responde a um conjunto de regras descritas para tal. Neste contexto, importa saber qual o grau de liberdade que é dado ao utilizador, para a definição dos seus próprios de dados.

- Uso de meta-informação

O uso de meta-informação, nos formatos de dados, revela-se extremamente importante no contexto de redes de computadores visto que a meta-informação permite catalogar os dados, através de um conjunto de características que permitam aos possíveis sistemas interessados saber da sua existência[Roc05]. Num cenário militar, onde existem muitas fontes de informação, e muitos sistemas interessados nessas fontes, é muito importante que os formatos tenham esta capacidade na sua definição.

¹⁵Simbologia Militar no contexto deste trabalho significa ter mecanismo de acesso a uma base de dados de símbolos militares normalizados no contexto da NATO.

- Extensibilidade do formato

Os Teatros de Operações militares estão em constante evolução, logo, é importante que o formato de dados tenha mecanismos que permitam fazer face a novas situações. Para este estudo interessa saber se o formato permite, de alguma forma, ser estendido no intuito de fazer face a novos desafios.

- Herança

Quanto mais rica for a codificação da realidade que queremos representar, mais verbosa ela se tornará e conseqüentemente mais pesada ao nível do sistema. A Herança é um mecanismo que, de certa forma, diminui essa verbosidade, dado que, se um elemento herda as características do pai, não necessita de voltar a especificá-las, tal como acontece na programação Orientada a Objetos. Neste contexto interessa saber os formatos que permitem ou não este mecanismo.

- Definição de dados pelo utilizador

A definição de dados pelo utilizador vem um pouco na continuação da “Extensibilidade do formato”, mas com carácter mais específico, no que diz respeito à definição de dados no próprio ficheiro de dados, pois o utilizador pode ter necessidade de representar algo novo ou diferente. Não sendo crucial num formato de dados é uma funcionalidade que pode ajudar numa situação nova em que o utilizador se encontre.

- Utilização de recursos externos

Um formato de dados tem de estar preparado para poder embeber recursos externos (informação que se encontra num local exterior ao ficheiro de dados, esta informação pode ser texto, imagem, vídeo, etc) dentro do seu formato. O acesso a recursos externos permitir-lhe-á usar recursos de que não dispõe localmente, de forma remota. A utilização de recursos já existentes poupa tempo e esforço ao utilizador/sistema que está a utilizar o formato, para transformar algo que já está codificado num dado formato, por exemplo numa imagem. Além disso, diminui significativamente a dimensão do ficheiro de dados.

- Controlo sobre a visualização

O controlo sobre a visualização revela-se de extrema importância quando estamos a visualizar cenários demasiado grandes para caberem no ecrã¹⁶, com uma resolução a que a informação a ser mostrada seja perceptível. Possuir o controlo sobre o nível de detalhe, a atribuir aos dados a serem visualizados numa dada imagem, é muito importante para permitir uma boa compreensão da informação que está a ser visualizada. Neste contexto, interessa saber que formatos estão apetrechados com mecanismos que lhe permitam efetuar este controlo.

¹⁶Ecrã ou outro dispositivo de visualização gráfica.

- Capacidade de 3-D

Num formato em que a visualização é o principal modo de receção dos dados por parte do utilizador, a capacidade de representação 3-D, pode fazer diferença. Também no contexto militar a representação 3-D tem crescido de importância. A modelação do terreno e a identificação de entidades são duas áreas onde este tipo de visualização apresenta grandes vantagens. Pese embora, a falta de doutrina que a NATO apresenta nesta área, a capacidade de representação tridimensional da realidade confere a um formato de dados um aumento da sua expressividade que tem de ser levada em linha de conta. Importa saber quais os formatos que têm esta característica.

- Gestão dos recursos

Carregar um recurso pode tornar-se numa operação muito dispendiosa para o sistema. Como foi referido no ponto anterior acerca da utilização de recursos, é importante que os formatos possibilitem o acesso a recursos externos, mas também, que os mesmos só sejam carregados quando forem mesmo necessários. Por defeito, aquando do carregamento inicial do ficheiro que contém os dados, esses recursos vão ser carregados, mesmo se por algum motivo não forem visualizados. Por exemplo, numa situação em que apenas se está interessado na visualização de parte da *picture*¹⁷, o sistema carrega desnecessariamente, toda a informação que está no ficheiro que contém a *picture* e todos os recursos externos existentes. Neste sentido, importa distinguir quais os formatos que têm mecanismos nos seus dados que permitam fazer a gestão dos recursos.

- Expressividade na representação de entidades geográficas

A expressividade (capacidade de detalhar algo) na representação de entidades geográficas é algo que tem de ser avaliado quando se trata de comparar diferentes formatos de dados que pretendem representar uma qualquer realidade geográfica. Quanto mais expressivo for esse formato, mais facilmente conseguirá fazer passar a mensagem correta dessa realidade. Logo, importa analisar a expressividade de cada um dos formatos.

O segundo método a ser utilizado é o da **Comparação quantitativa dos diferentes formatos num cenário** que visa dar resposta à segunda questão de investigação (subsecção 1.2.3). Este método requer a especificação e construção de um simulador de cenários, que permita a criação de cenários por forma à realização de testes aos diferentes formatos de dados. Paralelamente a este simulador, ter-se-á que especificar e implementar um sistema de recolha automática de dados, para se poder efetuar um estudo quantitativo dos diferentes formatos de dados.

O estudo quantitativo assenta nas seguintes análises, respetivamente: na análise da dimensão dos ficheiros produzidos por cada formato de dados a partir de um mesmo

¹⁷Designação da área visual que representa a situação em que estamos interessados.

cenário, na análise do tempo de transmissão necessário para cada formato de dados. Este estudo comparativo vai permitir perceber qual dos formatos possibilita melhor desempenho ou necessita de menos recursos.

1.4 Principais contribuições

As principais contribuições que se pretendem atingir com a realização desta dissertação são:

1. Comparação qualitativa dos formatos em avaliação, resultante de um estudo sistemático da especificação de cada um deles, segundo os mesmos critérios;
2. Desenho e implementação de um simulador de cenários militares;
3. Seleção de métricas para a avaliação dos vários formatos e implementação de um sistema para a recolha automática das referidas métricas;
4. Comparação quantitativa entre os diversos formatos de dados georreferenciados, num cenário realista.

2

Enquadramento

2.1 Cenário

No contexto desta dissertação, um **cenário** (ver figuras 2.1 e 2.2) é um conjunto de informações relativas a uma operação militar no *campo de batalha*¹ - por motivos de compreensão do conceito, para efeitos desta dissertação, o *campo de batalha* é o espaço físico onde se realiza uma operação militar - que se faz representar por uma imagem gráfica, designada por *Transparente de Operações*, conjugada com informação complementar associada a cada elemento do cenário. Cada cenário consiste num conjunto de elementos que podem ser de dois tipos: a) Medidas de Coordenação; b) Forças (ver figura 2.2).

Neste cenário encontram-se graficadas forças terrestres, meios aéreos e meios navais. No terreno e acima da linha "LD/LC"² encontram-se as nossas *forças terrestres* (Azul) pertencentes a uma unidade de escalão *Brigada Independente*³; abaixo dessa mesma linha temos três unidades de *forças terrestres* hostis (Vermelho) de escalão *Companhia*. No oceano Atlântico, à esquerda, temos mais acima um navio de guerra pertencentes às *nossas forças* e abaixo um navio mercante (Verde). Existem também várias *Medidas de Coordenação*, salientado-se o *Corredor Aéreo* que passa ao centro do *Campo de Batalha*, onde podemos ver, ao cimo, uma aeronave em deslocamento no corredor.

¹Segundo [Por05], "O campo de batalha moderno é não-linear, com áreas de operações contíguas ou não contíguas, incluindo, para além da vertente aérea, terrestre e marítima, igualmente a componente espacial, onde se desenrolarão os combates da informação, e o espectro eletromagnético. Inclui ainda as características do terreno, a população, as forças amigas e o adversário."

²Linha de Partida/Linha de Contacto das nossas forças com as *Forças Hostis*.

³Designa-se por *Brigada Independente* às unidades de escalão *Brigada* que não estão integradas no escalão imediatamente superior, *Divisão*. Estas *Brigadas* são reforçadas com meios de apoio logístico que lhes permite atuar isoladamente.

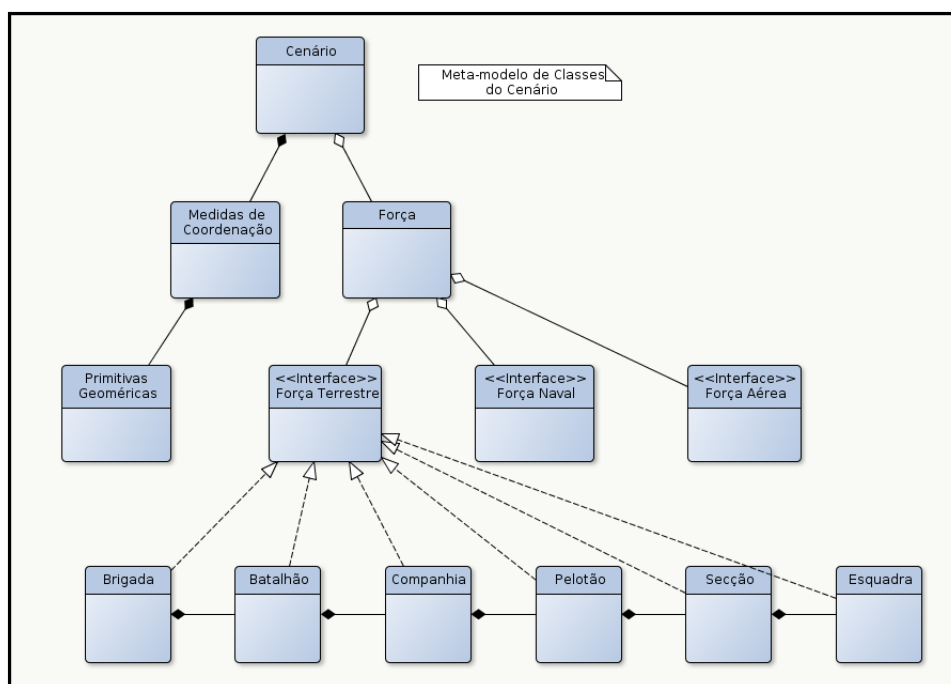


Figura 2.1: Meta-modelo de Classes do Cenário.

2.1.1 Medidas de Coordenação

As *Medidas de Coordenação*, no contexto militar, definem espaços no campo de batalha onde certas ações têm de ser ou não coordenadas entre as diversas unidades da Força. Estas medidas estabelecem normas a ser seguidas, que durante o período em que vigoram, facilitam as operações. As mesmas são usadas para diversos tipos de situação: para coordenar esforços entre as unidades, organizar a disposição das forças no terreno, permitir o acesso coordenado a recursos escassos ou partilhados, evitar o fratricídio entre as forças amigas, etc. Estas *Medidas de Coordenação* são representadas graficamente recorrendo a primitivas geométricas como linhas, pontos, polígonos, círculos, elipses, etc. Podemos visualizar na Figura 2.3 um exemplo de uma Medida de Coordenação. Uma Linha de Fase é uma *Medida de Coordenação* que permite coordenar ações, normalmente movimentos, entre várias unidades militares, por exemplo, todas as forças tem de estar posicionadas ao longo dessa linha na data e hora planeadas.

2.1.2 Força

Uma *Força Militar* (Força), no contexto militar, representa meios humanos e materiais normalmente divididos em três ramos: Força Terrestres, Força Aérea e Força Naval. As Forças Terrestres, são agrupadas em diferentes tipos de **Unidades Militares**: Infantaria, Cavalaria, Artilharia, Engenharia Militar, Transmissões, Serviço e Material e Administração. Estas unidades estão organizadas hierarquicamente por escalões (ver figura 2.1): Brigada (3-5 Batalhões), Batalhão (3-5 Companhias), Companhia (3-5 Pelotões), Pelotão

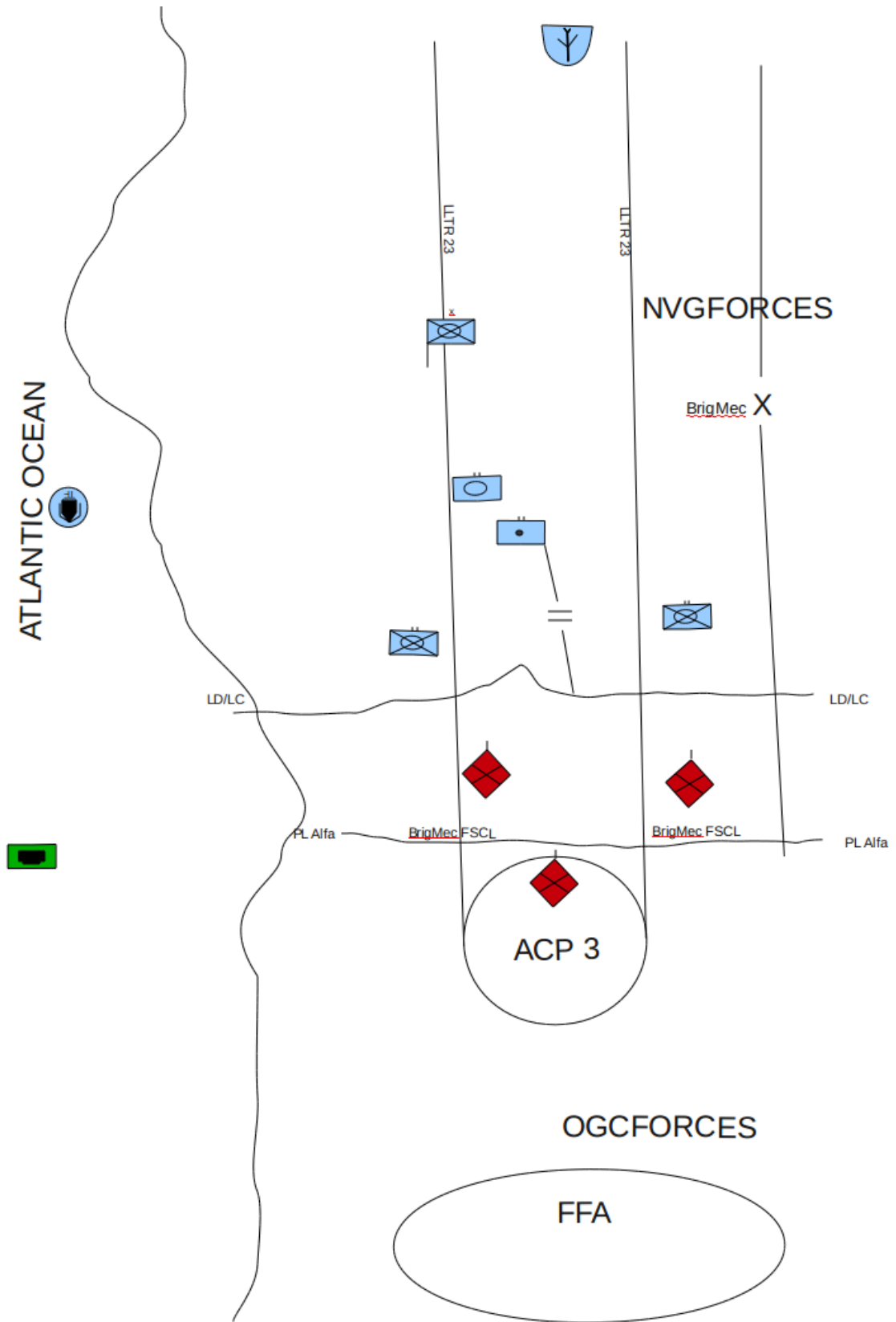


Figura 2.2: Transparente de Operações de uma operação de guerra convencional.

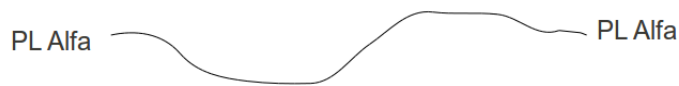


Figura 2.3: Linha de Fase.

(3-5 Secções), Secção (3-5 Esquadras), Esquadra (5 militares).

As *Unidades Militares* são representadas por uma simbologia normalizada em documentos próprios para o efeito, um exemplo desses documentos é o APP-6B[doc08], que contém os símbolos militares normalizados, para permitir uma interoperabilidade conjunta entre das forças, na NATO. A simbologia utilizada assenta em símbolos gráficos a duas dimensões onde a cor dos mesmos é utilizada para distinguir as diferentes partes envolvidas: a) Azul, representa as nossas forças; b) Verde, representa as Forças Amigas; c) Vermelho, representa as Forças Hostis. Estes símbolos recorrem à utilização de pequenas imagens e texto para a representação da informação militar de forma clara, precisa e concisa do Campo de Batalha. Podemos observar um exemplo de um símbolo que representa uma unidade militar, na Figura 2.4.

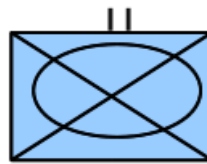


Figura 2.4: Batalhão de Infantaria Mecanizado.

2.1.3 Evolução do Cenário

Importa referir que todos os elementos que aparecem num cenário são georreferenciados, ou seja, têm sempre associados a eles a posição em que se encontram. Ao longo da Campanha Militar (tempo que decorre desde o início do conflito até ao seu final) o Campo de Batalha evolui fruto das mais diversas razões (unidades que se deslocam, unidades que são destruídas, etc). Logo, um cenário, uma vez criado, tem de ser regularmente atualizado para estar sempre sincronizado com a realidade dos acontecimentos do Campo de Batalha.

Durante estas atualizações, se tivermos em consideração uma taxa de atualização por segundo, a relação de alteração entre os elementos do tipo *Medidas de Coordenação* e elementos do tipo Força, é significativa. Naturalmente que esta taxa dependerá de tipo de cenário para tipo de cenário mas, a título de exemplo, as *Medidas de Coordenação* têm duração por vezes de várias horas ou até dias, enquanto que nos elementos do tipo Força a informação que lhes está associada (localização, velocidade, direção, etc), é frequentemente atualizada.

Nesta dissertação, a complexidade do cenário, é resultante do número de elementos

do tipo *Medidas de Coordenação*. A dimensão do cenário depende e dimensão da Força e do escalão que queremos trabalhar como elemento do tipo Força.

2.1.4 Batalha de al-Khafji

Para ilustrar de forma mais clara um cenário típico usaremos como referência a *Batalha de al-Khafji*, ocorrida na primeira Guerra do Golfo, entre as forças da Coligação e as forças do Iraque, em Janeiro de 1991[Wes08]. Segundo [Wes08], a *Batalha de al-Khafji* foi uma tentativa do então presidente do Iraque, Saddam Ussein, precipitar uma guerra terrestre com as forças da Coligação. *Ra's al-Khafji* é uma cidade na Arábia Saudita a 7 quilómetros do Kuwait, onde existe um porto com importantes refinarias de petróleo. O ataque foi concretizado na noite de 29 para 30 de Janeiro de 1991 pelas forças iraquianas. A reconquista da cidade foi efetuada por parte das forças da Coligação de 30 para 31 de Janeiro. Ao longo da experiência que vamos realizar, usaremos a operação que ficou conhecida por "Liberation of al-Khafji"(ver figura 2.5) **como referência para a dimensão das forças** a colocar num cenário geograficamente localizado em Portugal, de modo a garantir que estamos a trabalhar com um cenário de dimensão realista.

2.2 Estratégias de Comunicação

A informação presente nos cenários referenciados em 2.1, é conseguida através da integração da informação partilhada entre os diversos sistemas de C2 presentes no Campo de Batalha. A comunicação efetuada entre estes sistemas pode seguir dois padrões de comunicação diferentes:

- **Request-Response**, também conhecido como *Request-reply*, é um padrão de troca de mensagens em que um solicitante envia uma mensagem de pedido a um sistema solicitador que recebe e processa o pedido, que em última análise, retorna uma mensagem como resposta. Este é um simples, mas poderoso padrão de comunicação que permite que duas aplicações tenham uma comunicação com dois sentidos num só canal. Este padrão é especialmente comum nas as arquiteturas cliente-servidor[HW03].
- **Publish-Subscribe**, é um paradigma em que as aplicações comunicam através de troca de mensagens. A aplicação que envia a mensagem não especifica o recetor da mensagem. Em vez disso, a aplicação publica a mensagem num *middleware* que fica responsável pela interação. Por esta razão as aplicações que publicam mensagens designam-se por *publisher*. Se uma aplicação está interessada em receber mensagens dum dado tipo, então essa aplicação subscreve (*subscribe*) esse tipo de mensagem no *middleware*, ou seja, regista os seus interesses. Quando *publisher* envia uma mensagem de um dado tipo, o *middleware* envia uma cópia dessa mensagem para todas as aplicações que tenham subscrito esse tipo de mensagem [GA04].

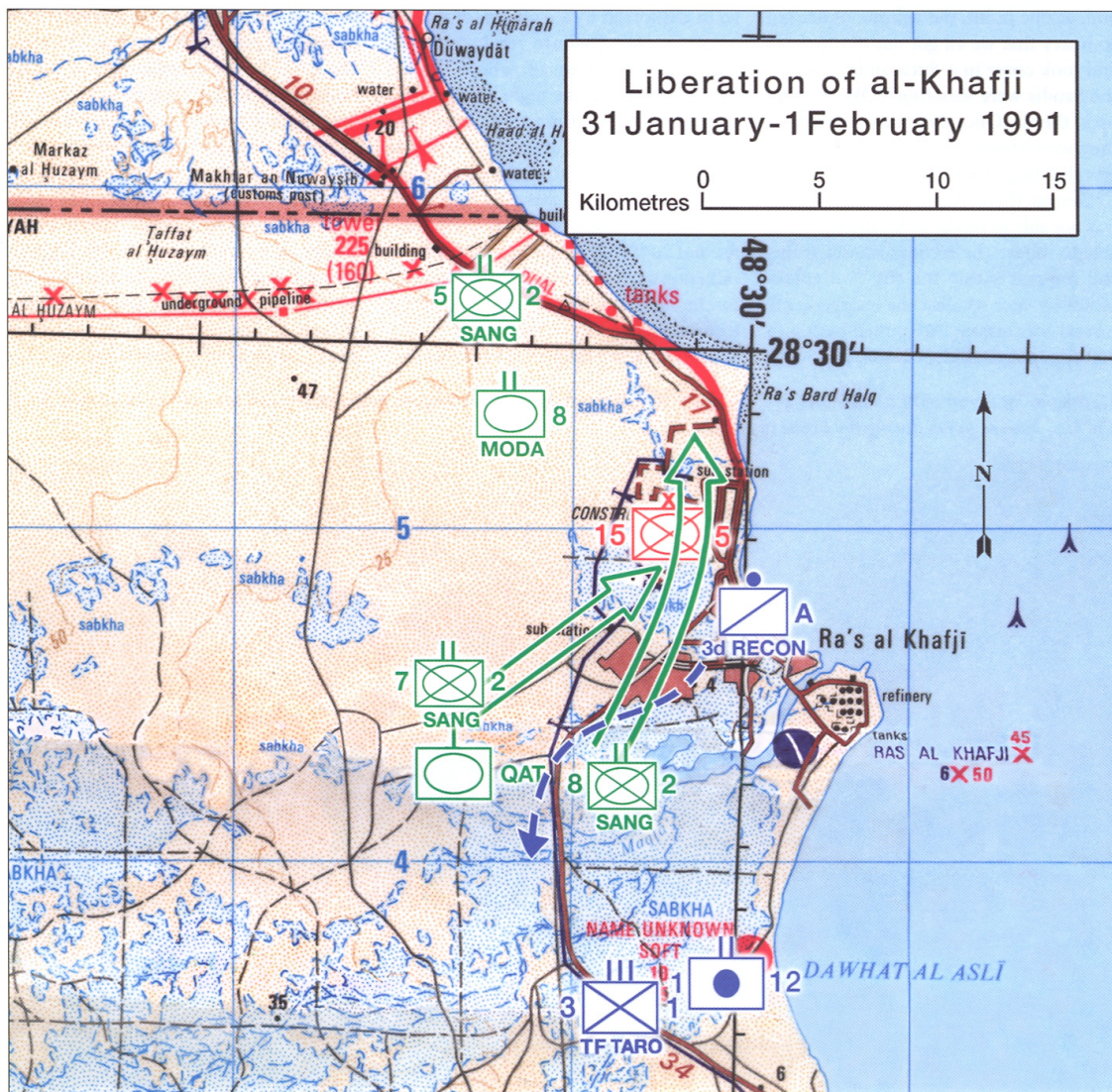


Figura 2.5: Transparente de operações da operação "Liberation of al-Khafji". [Wes08]

A utilização destes padrões poderá ter impactos diferentes, sobretudo ao nível do desempenho, em cada um dos formatos de dados em apreciação. Logo, é necessário avaliar qual o comportamento de cada formato nos diferentes padrões de comunicação, na geração do cenário inicial e durante o ciclo de atualizações do mesmo.

2.2.1 Request-Response

Fruto da decisão de desenho tomada, neste padrão de comunicação cada vez que o Cliente faz um pedido ao Servidor este responde-lhe com toda a informação que tem no momento referente a esse cenário, não se preocupando se a informação que está a enviar é repetida. Ou seja, o Servidor envia ao Cliente todos os elementos que constituem o cenário: os que sofreram e os que não sofreram atualizações. Neste contexto, parece evidente que em cenários mais complexos (número elevado de elementos do Tipo de *Medidas de Coordenação*) o formato com melhor desempenho na codificação de medidas de

coordenação leva vantagem em relação aos restantes formatos, dado que os elementos do tipo *Medidas de Coordenação* tem um custo quer em espaço quer em número de nós muito superior aos elementos do tipo *Força*.

2.2.2 *Publish-Subscribe*

Neste padrão de comunicação o Cliente apenas recebe do Servidor os elementos do cenário que sofreram alguma alteração. O Cliente fica à espera que o servidor lhe envie as atualizações do cenário inicial. Deste modo os ficheiros que circulam na rede são de tamanhos relativamente menores que os do padrão anterior. Para cenários em que os elementos do tipo *Medidas de Coordenação* sejam alterados com uma frequência baixa, relativamente aos elementos do tipo *Força*, os ficheiros que circulam na rede são constituídos com apenas elementos do tipo *Força*.

2.2.3 Mapeamento de cenários em ficheiros

Para melhor se perceber a importância do impacto da utilização dos padrões de comunicação atrás referidos, apresentamos na figura 2.6 três diagramas onde tentamos de uma forma genérica mostrar as diferenças entre os padrões no mapeamento de cenários em ficheiros. No primeiro diagrama titulado por cenário, apresentamos um retângulo (retângulo da esquerda) titulado por *Cenário Inicial*, que representa um cenário no momento após ter sido constituído, ou seja, as forças militares estão prontas para iniciar a operação militar. Os restantes retângulos titulados por *T1*, *T2* e *T3*, representam os vários estados do cenário nos instantes *t1*, *t2* e *t3*, decorrentes da evolução do *Cenário Inicial*. Os quadrados numerados dentro de cada estado do cenário (*Cenário Inicial*, *T1*, *T2* e *T3*) representam os elementos que constituem o cenário. A semântica das cores utilizada nos elementos foi a seguinte: a) a azul, os elementos que estão no estado inicial ou num estado em que não sofreram alterações; b) a verde, os elementos que sofreram alterações na atualização do cenário no estado em que se encontram; c) a encarnado, os elementos que ficaram destruídos e foram eliminados. Nos dois diagramas seguintes intitulados de *Request-Response* e *Publish-Subscribe*, apresenta-se uma visualização genérica da diferença no mapeamento em ficheiros dos elementos existentes no Cenário, durante a troca de informação entre sistema de C2, aquando da utilização do padrão de comunicação *Request-Response* e do padrão de comunicação *Publish-Subscribe*, respetivamente. Nesses dois diagramas existem quatro retângulos titulados por **Ficheiro Inicial**, **Ficheiro 1**, **Ficheiro 2** e **Ficheiro 3** que mapeiam os respetivos estados do cenário (*Cenário Inicial*, **T1**, **T2**, e **T3**), Ficheiro Inicial mapeia **Cenário Inicial**, e assim sucessivamente. Em cada ficheiro é representado o mapeamento de um determinado elemento através de um retângulo com o número do elemento.

Como se pode observar no diagrama *Request-Response*, todos os elementos existentes no cenário são mapeados e enviados no ficheiro, enquanto que, no diagrama *Request-Response*, após um mapeamento integral do cenário inicial, só os elementos que sofreram

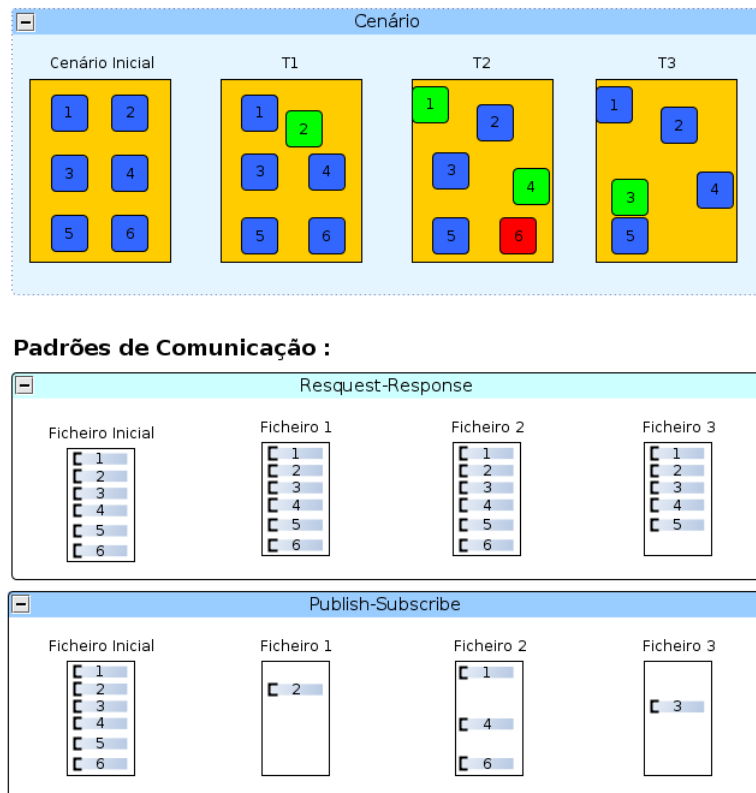


Figura 2.6: Diagramas ilustrativos da codificação da informação presente num cenário nos diferentes padrões de comunicação.

alteração nesse estado são mapeados e enviados.

2.3 Cenários **com** e **sem** Medidas de Coordenação

Por norma um cenário é composto por elementos do tipo Medidas de Coordenação e do tipo força, no entanto, neste estudo vamos estudar as duas modalidades: cenários **com** e **sem** Medidas de Coordenação. Um cenário **sem** Medidas de Coordenação é um cenário que apenas contém elementos do tipo Força. A diferença entre um cenário **sem** Medidas de Coordenação e um cenário **com** Medidas de Coordenação esta no número de elementos XML.

As razões que nos levaram a tomar esta decisão foram essencialmente as seguintes:

- Os utilizadores dos formatos podem estar interessados em utilizar os formatos em cenários sem Medidas de Coordenação, por diversas possibilidades de: a) poderem assim trabalhar com tamanhos de ficheiro mais pequenos; b) apenas estarem interessados na monitorização das posições das forças no terreno; c) trabalhar com um protocolo que envia as Medidas de Coordenação apenas de hora a hora mas envia os elementos do tipo Força de segundo em segundo, entre outras;
- Para se perceber qual o impacto das Medidas de Coordenação na dimensão do

ficheiro final e na *performance* de cada formato no processo de transferência do ficheiro.

2.4 Formatos de Dados e Critérios de Elegibilidade

Para a divulgação e massificação do uso de informação geográfica (IG), em muito contribuiu a *Open Geospatial Consortium* (OGC)⁴, organização criada em 1994. A OGC tem hoje mais de 400 membros, e é responsável pelas principais normas de formatos de dados para codificação de informação georreferenciada de utilização livre. No entanto, existem inúmeros formatos de dados proprietários e de domínio público criados para fazer face a necessidades específicas das organizações que os sustentam. Por condicionamentos temporais optamos por pré-selecionar um conjunto de formatos de dados que pela sua divulgação, maturidade, importância e aceitação na comunidade, poderiam constituir-se uma boa alternativa ao NVG. Os formatos pré-selecionados para análise foram:

- ESRI *Shapefile*⁵, formato de dados desenvolvido pela Environmental Systems Research Institute⁶ (ESRI);
- *Geography Markup Language*[Gro07] (GML), formato de dados *standard* da OGC;
- *Keyhole Markup Language*[Wil08] (KML), formato de dados *standard* da OGC;
- *Scalable Vector Graphics*[Gro09] (SVG), formato de dados *standard* da World Wide Web Consortium⁷ (W3C);
- *GeoTIFF*⁸, formato de dados criado pelo Dr. Niles Ritter na NASA - Jet Propulsion Labs;
- *Intergraph Raster File Format*⁹ (INGR), formato de dados desenvolvido pela Intergraph¹⁰.
- *Joint Consultation, Command and Control Information Exchange Data Model* (JC3IEDM) [Mem09], formato de dados desenvolvido pelo Multilateral Interoperability Programme¹¹ (MIP) no seio da NATO.

Para este estudo comparativo de formatos de dados georreferenciados com o formato de dados *NATO Vector Graphics* (NVG), desenvolvido pela NATO, com o propósito de efetuar a partilha de informação georreferenciada entre sistemas de C2, foram selecionados formatos de acordo com os seguintes critérios de elegibilidade:

⁴<http://www.opengeospacial.org>

⁵<http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>

⁶<http://www.esri.com/>

⁷<http://www.w3.org/>

⁸<http://www.remotesensing.org/geotiff/spec/geotiffhome.html>

⁹<http://oreilly.com/www/centers/gff/formats/ingr/index.htm>

¹⁰<http://www.intergraph.com/>

¹¹<https://mipsite.lsec.dnd.ca/Pages/Default.aspx>

Tabela 2.1: Critérios de elegibilidade

Critérios de elegibilidade	Av. Qualitativa	Av. Quantitativa
Capacidade de codificar informação georreferenciada	Sim	Sim
Utilização livre e código aberto	Sim	Sim
Amplamente divulgado e com aceitação global	Sim	Sim
Sustentado por uma organização internacional credível	Sim	Sim
Usa XML como linguagem de codificação	Sim	Sim
Formato de dados vetorial	Sim	Sim
Schema definido e disponível	Não	Sim

Como se observa na tabela 2.1, a Avaliação Qualitativa (AQL) e Quantitativa (AQt) têm em comum todos os critérios de elegibilidade com a exceção do último, *Schema* definido e disponível.

Capacidade de codificar informação georreferenciada A seleção deste critério, no contexto de uma avaliação sobre formato de dados para troca de informação georreferenciada, apresenta-se como obrigatória uma vez que existem formatos de dados que não contêm esta capacidade.

Utilização livre e código aberto Uma vez que queremos estudar alternativas ao formato de dados NVG passíveis de serem utilizadas pela NATO têm de ser obrigatoriamente de utilização livre e se queremos avaliar esse formato têm de ter o código aberto para que possa ser estudado e debatido por nós e restante comunidade.

Amplamente divulgado e com aceitação global Neste estudo apenas estamos interessados em estudar alternativas que representem um valor acrescentado para além das características intrínsecas ao formato de dados, ou seja, eliminem o esforço de manutenção do formato por parte do utilizador (leia-se NATO), ofereçam garantias de utilizar um formato estável por já se encontrar num estado de maturidade avançada, e que facilitem a sua utilização pela existência de diversas ferramentas disponíveis para trabalhar com o respetivo formato.

Uso da XML como linguagem de marcação dos dados O uso da linguagem *eXtensible Markup Language* (XML) da W3C, como Linguagem de Definição de Dados (LDD) é um factor decisivo no contexto da interoperabilidade. O uso do XML como formato de dados para a troca de informação é fundamental, pelas seguintes razões: a) a disponibilidade e ampla utilização de especificação de suporte e ferramentas para o XML que permite aos utilizadores da linguagem perder mais tempo na definição de dados que no processo de aprendizagem de novas ferramentas ou linguagens; b) XML é um padrão aberto, em formato texto, facilmente convertível em outros formatos, de fácil leitura e

processamento; c) o XML permite mecanismos de pesquisa muito mais eficientes, o que possibilita o aumento do reaproveitamento dessa informação[eJAR07]; d) o XML permite grande flexibilidade quando utilizado corretamente, que contribui para um aumento da sua extensibilidade.

Formato de dados vetorial A importância deste critério prende-se com a nossa intenção de comparar formatos de dados do mesmo tipo que o NVG. Como o NVG é um formato de dados do tipo vetorial todos os formatos alternativos são do tipo vetorial.

Schema definido e disponível Este critério é o único que apenas se aplica à avaliação quantitativa. Este critério é importante para a avaliação quantitativa porque neste tipo de avaliação precisamos de instanciar os formatos de dados para os podermos analisar quantitativamente. Para se proceder a uma instanciação e validação do formato de dados necessitamos de um *schema* desse formato definido e disponível, logo, a inexistência desse *schema* ou a indisponibilidade de acesso ao mesmo, inviabiliza o formato de participar numa avaliação desse tipo.

Tabela 2.2: Grelha de seleção dos formatos de dados

Critérios de legibilidade	GML	KML	SVG	GeoTIFF	Shapefile (ESRI)	INGR	JC3IEDM
Capacidade de codificar informação georreferenciado	Sim	Sim	Sim	Sim	Sim	Sim	Sim
Utilização livre e código aberto	Sim	Sim	Sim	Sim	Sim	Sim	Não
Amplamente divulgado e com aceitação global	Sim	Sim	Sim	Sim	Sim	Não	Não
Sustentado por uma organização internacional credível	Sim	Sim	Sim	Sim	Sim	Sim	Sim
Uso da XML como linguagem de marcação dos dados	Sim	Sim	Sim	Não	Não	Não	Não
Formato de dados vetorial	Sim	Sim	Sim	Não	Sim	Não	Não
Schema disponível na Internet que permita a validação dos dados	Não	Sim	Sim	Não	Não	Não	Não

Como se pode observar na tabela 2.2, quatro dos formatos pré-selecionados falharam em pelo menos um dos critérios de seleção da avaliação qualitativa e quantitativa, tendo por isso sido excluídos deste estudo: GeoTIFF, ESRI *Shapefile*, INGR e JC3IEDM. O formato de dados GeoTIFF foi eliminado por não ser um formato dados vetorial e por não recorrer à linguagem XML para a sua codificação. O formato ESRI *Shapefile* foi eliminado, por também não recorrer à linguagem XML para a sua codificação. O formato INGR foi eliminado por não recorrer à linguagem XML para a sua codificação, não ser um formato vetorial, e não ser um formato de dados com uma grande utilização fora da comunidade que utiliza produtos da *Intergraph*. E por último, formato JC3IEDM embora sendo um formato de dados desenvolvido no seio da NATO, falhou os mesmos critérios de elegibilidade que o formato de dados INGR, e ainda falha no critério da *Utilização livre e código*

aberto uma vez que não está disponível para utilização por parte da comunidade.

2.5 Formatos de dados selecionados

Nesta dissertação propomos uma comparação qualitativa entre quatro formatos de dados georreferenciados: NVG, SVG, KML e GML; e uma comparação quantitativa entre três dos anteriores quatro formatos: NVG, SVG e KML. Como se pode observar na tabela 2.2 o GML não preencheu todos os requisitos necessários para tornar possível uma comparação quantitativa com os restantes formatos.

2.5.1 NATO Vector Graphics[NC310b] (NVG)

O NVG surgiu da necessidade de representação da informação militar contida nos diversos Transparentes de Operações num formato digital de fácil codificação, para troca de informação entre sistemas de C2.

Estrutura tipo do ficheiro Na figura 2.7 apresenta-se a codificação de *Posto de Comando de Brigada* no formato NVG, com a qual se pretende mostrar a forma como se estrutura este formato de dados.

```

1 <nvg version="1.4.0" classification="NOT UNCLASSIFIED">
2 <point label="PostoComandoBrig" symbol="app6a:sfg*vciz---*****" x="-9.41745" y="39.00267"/>
3 <metadata><![CDATA[<h4>PostoComandoBrig</h4> <ul class="textattributes">
4 | | | | | <li><strong><span class="atr-name">name</span>:</strong> <span class="atr-value">PostoComandoBrig</span></li>
5 | | | | | </ul>]]></metadata>
6 </nvg>

```

Figura 2.7: Código com a representação de um Posto de Comando em formato NVG.

2.5.1.1 Justificação da escolha

O formato de dados NVG, é um formato de dados que foi desenvolvido pela NATO para o domínio específico da troca de informação georreferenciada entre sistema de C2. Por esta razão, é o mais natural de todos os candidatos a participar neste estudo, dado ser o formato atualmente adotado.

2.5.1.2 Objetivos do NVG

O formato NVG foi desenvolvido com o objetivo principal de permitir as seguintes possibilidades [NC310b]:

- Providenciar uma especificação simples para a codificação de informações do Campo de Batalha que permita uma visão georreferenciada;
- Providenciar suporte para a Simbologia Militar;

- Permitir efetuar pesquisa aprofundada da informação do Campo de Batalha por forma a obter mais informação que aquela que está contida no formato de dados NVG.

2.5.1.3 Caso de Uso do NVG

O principal Caso de Uso do formato de dados NVG é o de apoiar a recolha de informações de várias fontes no Campo de Batalha com o propósito de sobrepor essa informação num dispositivo de visualização de informação georreferenciada [NC310b]. Este Caso de Uso permite o conhecimento detalhado da situação, recorrendo à acumulação de fontes de informação existentes no Campo de Batalha, com visões parciais.

2.5.2 Scalable Vector Graphics[Gro09] (SVG)

O SVG é um formato aberto, que começou a ser desenvolvido no ano de 1998 com a criação do grupo de trabalho *SVG Working Group*, na W3C. Este formato surgiu depois outros formatos como *Precision Graphics Markup Language* (PGML) e o *Vector Markup Language* (VML) terem sido propostos na W3C. Mas foi em 2001 que o SVG se tornou uma recomendação da W3C, tendo sofrido uma revisão com a saída da versão vigente SVG 1.1 em Janeiro de 2003 Neste momento está para consulta pública o documento *W3C Working Draft* com uma proposta para a edição da versão SVG 1.2.

O formato SVG é um formato que, originalmente, não foi concebido com o fim específico de trabalhar com informação georeferenciada. No entanto, existem estudos que consideram este formato como um formato adequado para SIG e aplicações de mapeamento [DGB06].

A georreferenciação neste formato efetua-se da seguinte forma, existe uma imagem vetorial SVG, que tem o seu canto superior esquerdo referenciado através de um par de coordenadas segundo um Sistema de Coordenadas de Referência, e todos os restantes elementos/entidades existentes no ficheiro têm as suas coordenadas referentes à origem de coordenadas da imagem que é efetivamente o canto superior esquerdo.

Estrutura tipo do ficheiro Na figura 2.8 apresenta-se a codificação de *Posto de Comando de Brigada* no formato NVG, com a qual se pretende mostrar a forma como se estrutura este formato de dados.

```

1 <?xml:version="1.0" standalone="no"?>
2 <svg width="500" height="414" xmlns:xlink="http://www.w3.org/1999/xlink"
3   contentStyleType="text/css" viewBox="0 0 800 600" xmlns="http://www.w3.org/2000/svg" version="1.0">
4   <title>PostoComandoBrig</title>
5   <metadata><![CDATA[<h4>PostoComandoBrig</h4>.<ul class="textattributes">
6     <li><strong><span class="atr-name">name</span>:</strong> <span class="atr-value">PostoComandoBrig</span></li>
7     </ul>]]></metadata>
8   <image x="200.0" y="260.0" fill="none" width="57" xlink:href="NVG/simbolos/PostoComandoBrig.jpg"
9     height="64" stroke="none" preserveAspectRatio="none"/>
10 </svg>

```

Figura 2.8: Código com a representação de um Posto Comando em formato SVG.

2.5.2.1 Justificação da escolha

O SVG foi selecionado para fazer parte deste estudo comparativo de formatos alternativos ao NVG, porque, para além de cumprir os critérios de seleção referidos em 2.4, foi o formato adotado para servir de base ao desenvolvimento do NVG, e constitui-se como uma alternativa relevante para a publicação de informação georreferenciada na Internet, principalmente por manter a geometria das entidades geográficas[Far07].

2.5.2.2 Objetivos do SVG

O formato de dados SVG, segundo a W3C, é uma linguagem de marcação que permite descrever gráficos a duas dimensões utilizando para isso o XML. O SVG disponibiliza ao utilizador três diferentes tipos de objetos gráficos: formas gráficas vetoriais (círculos, retângulos, polígono, linhas e curvas), texto e imagens. Estes objetos gráficos podem ser agrupados, transformados e animados.

2.5.2.3 Caso de Uso do SVG

- Descrição de gráficos de duas dimensões;
- Disponibilizar ao utilizador formas gráficas vetoriais, texto e imagens;
- Fornecer ao utilizador as operações de agrupar, transformar e animar para atuar sobre os objetos gráficos.

2.5.3 *Keyhole Markup Language*[Wil08] (KML)

O formato de dados KML foi criado pela Google, para o geo-browser¹² GoogleEarth, da mesma empresa. Em 2007 foi criado um grupo de trabalho para submeter a especificação da KML 2.2 à aprovação da OGC, para fazer parte dos seus standards. Esta aprovação apenas veio a acontecer em Abril de 2008.

O formato de dados KML, é desde 2008 um dos *standards* da organização *Open Geospatial Consortium* (OGC) designado oficialmente por *OpenGIS® KML Encoding Standard* (OGC KML). KML é uma gramática XML para codificar e transportar representação de informação geográfica para ser visualizada num geo-browser[webc].

O KML é um formato de dados que integra a definição dos objetos, o estilo da sua visualização e a sua representação. Segundo a OGC a KML é uma linguagem de marcação em XML, vocacionada para visualização geográfica, incluindo a anotação de mapas e imagens[Wil08].

Estrutura-tipo de um ficheiro Na figura 2.9 apresenta-se a codificação de *Posto de Comando de Brigada* no formato KML, com a qual se pretende mostrar a forma como se estrutura este formato de dados.

¹²Geo-browser é uma aplicação do lado do cliente que corre no computador do utilizador e que permite a pesquisa e visualização de informação georreferenciada.

```

1 <?xml:version="1.0" encoding="UTF-8"?>
2 <kml xmlns="http://www.opengis.net/kml/2.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://www.opengis.net/kml/2.2 http://schemas.opengis.net/kml/2.2.0/ogckml22.xsd">
4   <Document xmlns:atom="http://purl.org/atom/ns#">
5     <name>PostoComandoBrig</name>
6     <Placemark id="PostoComandoBrig.1">
7       <name><![CDATA[PostoComandoBrig.1]]></name>
8       <description><![CDATA[<h4>PostoComandoBrig</h4> <ul class="textattributes">
9         <li><strong><span class="atr-name">name</span>:</strong> <span class="atr-value">PostoComandoBrig</span></li>
10        </ul>]]></description>
11       <LookAt>
12         <longitude>-9.417448471568914</longitude> <latitude>39.002656723606094</latitude>
13         <heading>10.0</heading>
14         <tilt>10.0</tilt>
15         <range>700</range>
16       </LookAt>
17       <Style>
18         <IconStyle>
19           <color>ff0000ff</color> <colorMode>normal</colorMode>
20           <Icon><href>http://192.168.1.66:8080/geoserver/data/data/TO/PostoComandoBrig/PostoComandoBrig.jpg</href></Icon>
21         </IconStyle>
22         <LabelStyle><color>00ffffff</color></LabelStyle>
23       </Style>
24       <Point>
25         <coordinates>-9.417448471568914,39.002656723606094</coordinates>
26       </Point>
27     </Placemark>
28   </Document>
29 </kml>

```

Figura 2.9: Código com a codificação de um Posto Comando em formato KML.

2.5.3.1 Justificação da escolha

O KML foi seleccionado para este estudo, porque, para além de cumprir os critérios de seleção referidos em 2.2, é um formato de dados georreferenciados de referência, adotado pelos *geobrowsers* de utilização gratuita: Google Earth, Google Maps e Google Maps para telemóvel; que o tornam num dos formatos mais conhecidos e mais utilizados.

2.5.3.2 Objetivos do KML

Os objectivos, da OGC e da Google[Wil08] com esta normalização da especificação da KML, segundo a especificação oficial, são os seguintes:

- Disponibilizar um padrão internacional de uma linguagem para expressar anotação e visualização geográfica para as aplicações *geobrowsers*, nomeadamente: *earth browser (3D)* e *mobile maps (2D)*;
- O KML ficar alinhado com as melhores práticas e normas internacionais, possibilitando desta forma uma maior absorção e interoperabilidade das implementações de *geobrowser*.

2.5.3.3 Casos de Uso da KML

Segundo a especificação da OGC a linguagem KML pode ser utilizada para:

- Efetuar anotações na superfície terrestre;
- Especificar ícones e rótulos para identificar os locais da superfície do planeta;
- Criar diferentes posições de câmara para especificar vistas diferentes para as *features* do KML;

- Definir as sobreposições de imagens para anexar à Terra ou ao ecrã¹³;
- Definir estilos para especificar KML a apresentação das *features*;
- Escrever descrições em HTML das *features* do KML, incluindo hiperlinks e imagens incorporadas;
- Organizar KML *features* em hierarquias;
- Localizar e atualizar documentos KML localmente obtidos ou a partir de locais remotos na rede;
- Definir a localização e orientação de textura de objetos 3D.

2.5.4 *Geography Markup Language*[Gro07] (GML)

O formato de dados *Geography Markup Language* (GML), é uma especificação aberta da OGC, para codificar informação georreferenciada. O GML tem duas vertentes, serve como um linguagem modelo para sistemas geográficos, e serve como um formato aberto de troca de informação geográfica na Internet. Este formato de dados é uma norma da OGC desde 2000, e da *International Standard Organization* (ISO 19118) desde o ano de 2007.

O GML é uma meta-linguagem que tenta ser o mais completa possível, o que a torna algo complexa para ser compreendida e utilizada. É na tentativa de combater esta dificuldade que aparecem os perfis (*GML profile*). Os perfis definidos em GML, mais não são que restrições ao vasto conjunto de tipos de dados definidos nos esquemas padrões do GML, com o objetivo de ajudar o GML como formato *standard*.

O GML codifica as propriedades de objetos do mundo real como elementos (exemplo: a geometria de uma casa, ou a cor da casa, etc). A codificação em elementos é realizada por forma a que as propriedades de um objeto sejam sempre elementos filhos do elemento que representa o objeto[Trn05]. Este formato, obriga que as entidades sejam declaradas como um tipo ao qual estão associados atributos, e os elementos que se referem a atributos das entidades terão de ser obrigatoriamente filhos do elemento que define essa entidade[Trn05]. Mantendo esta regra simples o formato permite uma fácil pesquisa e interpretação da informação no mundo da *Web Semântica*.

Para proceder à utilização da *Geography Markup Language* é necessário desenvolver uma esquema de aplicação do domínio especificado em *XML Schema* segundo duas formas: a) utilizando as regras especificadas na ISO 19109 para *applications schemas* em UML¹⁴ (É um *standard* desenvolvido e mantido pelo *Object Management Group*) com o objetivo principal de desenvolver uma linguagem de modelação para a área de engenharia de software orientada a objetos.; b) ou recorre às regras para aplicação de esquemas GML especificados na ISO 19109 para a criação de um esquema de aplicação GML diretamente em *XML Schema*.

¹³Isto quer dizer às coordenadas do ecrã, segundo o sistema de eixos do ecrã que esteja a servir para visualização.

¹⁴<http://www.uml.org/>

Estrutura-tipo de um ficheiro Na figura 2.9 apresenta-se a codificação de *Posto de Comando de Brigada* no formato GML, com a qual se pretende mostrar a forma como se estrutura este formato de dados. O GML separa a representação dos dados da visualização dos mesmos, pelo que, necessitamos de um ficheiro do tipo *.gml* para representação dos dados e de um ficheiro do tipo de estilos *.sld*¹⁵ (*Styled Layer Description*) para dizer a forma como os dados devem ser visualizados (ver figura 2.11).

```

1 <wfs:FeatureCollection numberOfFeatures="0" timeStamp="2010-12-29T17:39:10.923Z"
2 xsi:schemaLocation="http://localhost:8080/geoserver/web/TO http://localhost:8080/geoserver/wfs?service=WFS&version=
3 1.1.0&request=DescribeFeatureType&typeName=TO%3APostoComandoBrig
4 http://www.opengis.net/wfs http://localhost:8080/geoserver/schemas/wfs/1.1.0/wfs.xsd">
5 <gml:featureMembers>
6 <TO:PostoComandoBrig gml:id="PostoComandoBrig.1">
7 <gml:name>PostoComandoBrig</gml:name>
8 <TO:the_geom>
9 <gml:Point>
10 <gml:pos>-9.417448471568914 39.002656723606094</gml:pos>
11 </gml:Point>
12 </TO:the_geom>
13 <TO:name>PostoComandoBrig</TO:name>
14 </TO:PostoComandoBrig>
15 </gml:featureMembers>
16 </wfs:FeatureCollection>

```

Figura 2.10: Código com a representação de um Posto de Comando, em formato GML.

```

2 <StyledLayerDescriptor version="1.0.0" xmlns="http://www.opengis.net/sld" xmlns:ogc="http://www.opengis.net/ogc"
3 xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4 xsi:schemaLocation="http://www.opengis.net/sld http://schemas.opengis.net/sld/1.0.0/StyledLayerDescriptor.xsd">
5 <NamedLayer>
6 <Name>PostoComandoBrig</Name>
7 <UserStyle>
8 <Name>PostoComandoBrig</Name>
9 <Title>simbolo Posto Coamdo Principal de Brigada</Title>
10 <Abstract>A sample of how to use an SVG based symbolizer</Abstract>
11 <FeatureTypeStyle>
12 <Rule>
13 <Title>PostoComandoBrig</Title>
14 <PointSymbolizer>
15 <Graphic>
16 <ExternalGraphic>
17 <OnlineResource xlink:type="simple" xlink:href="PostoComandoBrig.svg" />
18 <Format>image/svg+xml</Format>
19 </ExternalGraphic>
20 <Size>
21 <ogc:Literal>20</ogc:Literal>
22 </Size>
23 </Graphic>
24 </PointSymbolizer>
25 </Rule>
26 </FeatureTypeStyle>
27 </UserStyle>
28 </NamedLayer>
29 </StyledLayerDescriptor>

```

Figura 2.11: Ficheiro de estilos no formato SLD para a aplicar aos dados no formato GML na sua visualização.

2.5.4.1 Justificação da escolha

O GML foi escolhido para este estudo, porque para além de cumprir todos os critérios de legibilidade referidos em 2.1 para a Avaliação Qualitativa é, também, o formato de dados georreferenciados de referência da OGC para a codificação de informação georreferenciada. No entanto, este formato de dados não fará parte da Avaliação Quantitativa pelo motivo, intrínseco ao formato, da necessidade de definição de um *Application Schema*.

¹⁵<http://www.opengeospatial.org/standards/sld>

Tanto quanto se sabe, não existe nenhum *Application Schema standard* da OGC definido para este Fragmento, logo, a definição desse documento pela nossa parte constituiria sempre uma ameaça a validação dos resultados obtidos.

2.5.4.2 Objetivos do GML

O formato de dados GML, segundo a especificação da OGC e da ISO 19100, tem os seguintes objectivos :

- Oferecer uma estrutura aberta e independente de fornecedor para a descrição de esquemas de aplicação geo-espacial para o transporte e armazenamento de informação geográfica em XML;
- Permitir que perfis suportem subconjuntos apropriados das capacidades do GML;
- Suportar a definição de esquemas de aplicação geo-espacial para domínios especializados;
- Permitir a criação e a manutenção de esquemas de aplicação ligados a um conjunto de dados;
- Suportar ao armazenamento e transporte de *schemas* de aplicação e conjunto de dados;
- Aumentar a capacidade das organizações para compartilhar esquemas aplicação e as informações que eles descrevem.

2.5.4.3 Casos de Uso do GML

O formato de dados foi desenvolvido com o intuito de oferecer à comunidade as seguintes capacidades:

- Transporte de informação;
- Armazenamento de dados;
- Modelação geo-espacial de informação na web e uma linguagem de implementação.

Estas capacidades fazem do formato de dados GML, um formato muito dinâmico e versátil que levaram a muitas empresas na área de Sistemas de Informação Geográfica, incluindo a *Intergraph*¹⁶ (empresa que ocupa o 67º lugar no *ranking* das maiores empresas de desenvolvimento de software¹⁷) desenvolveram um componente para importar e exportar dados no formato GML [SY08].

¹⁶<http://www.intergraph.com/>

¹⁷<http://www.softwaretop100.org>



Trabalho Relacionado

Nesta dissertação pretendemos realizar comparações relativas ao tamanho dos dados e à performance. As referências para comparações entre formatos de dados georeferenciados não são abundantes, e o seu foco é baseado em comparações qualitativas dos formatos.

Neste capítulo vamos abordar a literatura disponível relativamente à temática da avaliação de formatos de dados (georeferenciados e não georeferenciados). Para melhor apresentarmos toda a informação recolhida, subdividimos este capítulo em três secções: Avaliação qualitativa de formatos de dados, avaliação quantitativa de formatos de dados e métricas de XML.

3.1 Avaliação qualitativa de formatos de dados georeferenciados

Em [dOLJ02], encontramos uma proposta de formato de intercâmbio de dados georeferenciados, GeoBR. Este estudo realiza uma comparação deste novo formato com um dos formatos de dados *standard* da OGC, o GML. A comparação tem por base um levantamento das principais características do formato GML e das suas principais fraquezas, como é o caso da necessidade de investimento de tempo para a criação de um *XML schema* de domínio da aplicação ou a lacuna que existe ao nível semântica dos dados definidos em GML, entre outras que foram entretanto colmatadas nas versões posteriores (3.2.1 é a versão atual). Segundo os autores estas fraquezas seriam colmatadas com este novo formato. O que importa aqui realçar não é tanto o novo formato de dados proposto pelo autor, mas sim, a necessidade que o autor teve da comparação do seu formato com uma referência, no contexto dos formatos de dados georeferenciados, que é o formato GML. As comparações com o formato de dados GML continuaram em [RGCM04], onde é apresentado um outro formato para solucionar os problemas da interoperabilidade da

informação recolhida por *Earth Scientists*, o qual é comparado com diversas soluções possíveis de formatos existentes na comunidade, entre os quais está o GML. Esta comparação foca-se no carácter diferenciado dos principais objetivos de cada formato, enquanto que na nossa dissertação foi produzida uma lista de características importantes a analisar nos diversos formatos para nos ajudar a sistematizar o processo e por forma a não deixar de lado por descuido ou esquecimento algum ponto importante.

Ron Lake, presidente da *Galdos Systems Inc*¹, um dos principais (ou mesmo o principal) contribuintes para o desenvolvimento do GML na OGC, apresenta numa revista da especialidade, *GeoWorld*, dois artigos [[Lak08b](#)] e [[Lak08a](#)] em que efetua uma comparação das principais diferenças entre o formatos de dados GML e o formato de dados KML na construção de um *Schema* para expressar um modelo de dados. Recorrendo a um exemplo prático enumera uma série de diferenças entre os dois formatos, por exemplo, ambas as linguagens permitem a extensão dos dados, no entanto o KML não permite a utilização de *tipos complexos*, nem *tipos enumerados* para além de também a noção de *namespace* neste *schema*. *Ron Lake*, conclui que KML e GML tem objetivos diferentes, GML está principalmente vocacionada para a representação de informação georreferenciada, enquanto que KML está principalmente vocacionada para a apresentação de informação georreferenciada. O estudo apresentado por *Ron Lake*, serve-se de exemplos bem definidos para ilustrar as diferenças entre os dois formatos de dados, também nós aplicamos esta técnica, sempre que isso nos foi possível, pois pensamos ser uma forma muito positiva de explicar as diferenças.

Também o trabalho apresentado em [[KKLL09](#)], volta a comparar estes dois formatos de dados, KML e GML, no âmbito da modelação de objetos georreferenciados 3-D, argumentando que a sua proposta de modelo dados 3-D alternativo, designado por *prism model*, vem resolver os problemas que os modelos KML e GML apresentam. Os critérios de comparação utilizados foram os seguintes: o poder expressivo, a dimensão dos dados produzida e o grau de dificuldade da implementação para cada modelo. O resultado da comparação feita entre o KML e GML no âmbito da modelação 3-D, foi o seguinte: o model 3-D KML, sendo de fácil implementação e produzindo pequenas quantidades de informação, apresenta uma expressividade insuficiente; o modelo 3-D GML, em oposição ao KML, é um modelo muitíssimo completo e enorme expressividade, mas que produz muita quantidade de informação e de difícil implementação. Este estudo [[KKLL09](#)], tal como a nossa avaliação qualitativa, segue critérios pré-definidos para realizar a comparação entre os diversos formatos, no entanto, ao contrário da nossa avaliação qualitativa, não apresenta no documento a forma de como chegou a essas conclusões critérios. Pensamos ser importante apresentar ao leitor a justificação dos resultados.

No que diz respeito à temática da visualização/apresentação, [[San08](#)] apresenta um estudo que aborda as capacidades do formato KML na temática dos mapas nas vertentes: Símbolos proporcionais, Imagens de icons escaláveis, Desenho de polígonos regulares, Objetos 3-D *Collada*, Gráficos, *Choropleth maps*, *Prism maps*, Mapas animados, Imagens

¹<http://www.galdosinc.com/>

Raster e Títulos dos mapas e Legendas dos mapas. Os formatos alternativos ao KML considerados em [San08] foram o KML, SLD(*Styled Layer Descriptor*) e SE (*Symbol Encoding*), e o SVG, no entanto, todos eles sofrem de um problema comum, não são formatos completos como o KML. O GML é uma boa linguagem para codificação e armazenamento de informação georreferenciada mas falta-lhe a capacidade da visualização. O SLD e SE são duas linguagens de estilização da informação, mas falta-lhes a capacidade de representação da informação. O SVG é uma linguagem com uma expressividade gráfica muito maior que o KML, mas é apenas 2-D e não incorpora conceitos para a navegação.

A forma de abordar a avaliação é sempre uma questão importante para o resultado da mesma, a forma que achamos mais profícua e clara e a que podemos observar em [Ols09], onde se efetua uma comparação entre o *CityGML*, novo formato de dados para troca de informação de modelos de cidades em 3-D *standard* da OGC, desde 2008, e o formato de dados KML. À semelhança do que foi realizado na nossa avaliação qualitativa, também esta comparação foi realizada de acordo com critérios previamente selecionados, Geometria, Georreferenciamento, Topologia, Semântica e Incorporação. Para cada critério foi estabelecida a seguinte taxonomia: básico, sofisticado, muito sofisticado. Seguindo o mesmo traço, aparece o estudo apresentado em [DLR11], um estudo que analisa e compara diversos *standards*, com o pressuposto de que um formato ser reconhecido como *standard*, significa que têm qualidade e é eficiente. Neste contexto, realiza também uma comparação entre os *standards* de Sistemas Informação Geográfica, apresentando como critério de seleção é o nível de popularidade dos formatos, em que apenas os mais utilizados são contemplados, entre os quais estão presentes o GML, KML e SVG. A comparação dos vários formatos neste estudo, à semelhança do anterior, segue critérios definidos antecipadamente. Sendo estes dois documentos muito interessante no ponto de vista global, no que diz respeito à comparação poderia ser mais esclarecedora para o leitor, se explicassem as razões que levaram à escolha dos critérios apresentados e não de outros critérios quaisquer.

Interessa registrar a importância que os autores dos estudos apresentados vêm nos formatos de dados: GML, KML e o SVG. São formatos de dados incontornáveis quando se fala de Sistemas de Informação Geográfica. No que diz respeito à forma como as avaliações qualitativas são realizadas, como podemos observar existem várias formas de o fazer, sendo que, na nossa opinião a avaliação comparativa com o levantamento inicial dos critérios é a mais indicada.

3.2 Avaliação quantitativa de formatos de dados

Na literatura disponível, não encontramos nenhuma comparação quantitativa de formatos georreferenciados. No entanto existem diversos exemplos de comparações de formatos de dados (não georreferenciados) usados para garantir a interoperabilidade entre

aplicações. Por exemplo, [NN09] compara dois diferentes formatos de intercâmbio de informação, XML e JSON, no intuito de determinar qual dos formatos é a opção mais adequada no contexto do intercâmbio de informação. A comparação utiliza como critérios: a utilização de recursos e a *performance* no tempo gasto na transferência da informação relativa das aplicações que usam o XML e JSON.

Um outro estudo [FSo07], realiza uma comparação quantitativa aos mesmos formatos de dados que a comparação anterior, mas neste estudo foi adicionado mais um formato de dados, YAML (Yet Another Markup Language). A comparação destes 3 formatos assentou em critérios semelhantes: o uso do recurso memória e a *performance*. Neste estudo verifica-se um elevado número de ameaças à validação dos resultados, uma vez que os dados gerados para servirem de teste não são exatamente os mesmos e são utilizados *parser* diferentes para realizar a mesma operação de *parsing*.

No mesmo contexto de formato de dados, mas com finalidades diferentes temos [GT11], cujo objetivo é comparar, novamente os formatos de intercâmbio de dados: XML, JSON e *Protocol Buffers*, para medir o impacto dos mesmos no consumo da bateria e da *performance* nos *Smartphones*. Os autores deste estudo elegeram vários critérios de comparação, à semelhança do que é realizado na nossa avaliação quantitativa, também eles analisam o tempo despendido em determinadas operações e o tamanho das instâncias que cada formato de dados produz. Mas as semelhanças não se ficam por aqui, pois também neste estudo se introduziu a compressão do formato para tirar vantagem de um menor número de *bytes* a serem trocados.

Num contexto diferente, mas com o mesmo objetivo de comparar formatos de dados diferentes que servem para realizar o mesmo fim, Rick Jelliffe, o famoso inventor da *Schematron schema language*², compara dois formatos de dados para troca de documentos, *Open Document Format* (ODF) e *MS Office Open XML*(MSOOX). Da mesma forma que nós utilizamos o mesmo cenário (a mesma informação de teste) para mapear nos diferentes formatos que estamos a comparar, também ele compara a mesmo documento (a mesma informação de teste) nos dois diferentes formatos, recorrendo a métricas de tamanho, métricas de *performance* e a métricas complexidade de XML, para perceber qual é o melhor formato de dados para ser usado como *standard office XML format*³.

3.3 Métricas de XML

Quando decidimos realizar uma avaliação comparativa dos diversos formatos de dados, procedeu-se ao levantamento das métricas a aplicar. Visto que estávamos a trabalhar com formatos de dados que utilizam o XML como linguagem de marcação, foi realizado um levantamento e análise sobre as *Métricas de XML* existentes na literatura disponível.

Constatou-se que existem muitos documentos disponíveis sobre métricas de XML, [Spa10], [RL05], [KSH02], [BM09], [ZZ03], [LHC06], [MSbY04], [fXS06] e [PSH10] são

²<http://www.schematron.com/>

³http://www.oreillynet.com/xml/blog/2006/08/comparing_xml_office_document_3.html

alguns dos principais. Cada documento, por norma, propõe diversas novas métricas de XML, em consequência, existe um elevado número de métricas XML disponíveis para ser utilizadas.

Em [Zha08] são apresentados de forma sistematizada os diversos documentos que abordam as métricas de XML. Existem vários tipos de métricas: métricas de Complexidade, métricas de Estrutura, métricas de Qualidade e métricas relacionada com facilidade de Manutenção. Apresenta-se na tabela 3.1⁴ um breve resumo das principais contribuições de cada documento para as métricas de XML.

Tabela 3.1: Documentos com métricas para XML.

Documento	Principais contribuições
[KSH02]	São propostas cinco métricas para avaliar o <i>Schema</i> dos documentos XML.
[MS04]	Propõe onze métricas para medir a qualidade e complexidade do XML <i>Schema</i> .
[RL05]	Apresenta conceitos essenciais para análise de esquemas XML com base em métricas de código de software. E analisa as informações quantitativas e qualitativas para XML <i>Schema</i> .
[QS05]	Centra-se na determinação da complexidade dos documentos XML com base nas características sintáticas e estruturais diferentes.
[LHC06]	Propõe algumas métricas para grandes propriedades estruturais do XML, especialmente os aninhamento de entidades e um-para-muitos relacionamentos.
[Vis06]	Propõe um conjunto de métricas para XML <i>Schema</i> para medir as propriedades estruturais.
[BM07]	Propõe uma métrica baseada na arquitetura interna dos componentes XSD e considera a complexidade dos seus componentes de construção.

O estudo da complexidade dos diferentes formatos, levanta uma série de questões, nomeadamente:

- **Qual a melhor forma para comparar a complexidade dos *Schemas* dos vários formatos definidos à custa de *W3C XML Schema* e *DTD*?**
- **Quais as métricas a utilizar nessa comparação? e Qual o significado prático de um melhor ou pior resultado relativo de complexidade?**

Estas questões que fogem ao âmbito do nosso estudo.

Na avaliação quantitativa dos diferentes formatos de dados que realizamos, utilizamos métricas de Tamanho: *File size* e *Number of all XML Nodes* aplicadas a documentos

⁴Esta tabela é parte da tabela *Summary of Research on XML Schema and DTD* apresentada em [Zha08].

<i>XML-agnostic schema size</i>	
File size	KB or LOC
<i>XSD-agnostic schema size</i>	
Number of all XML nodes	#NODE
Number of all XML nodes for annotation	#ANN
<i>XSD-aware counts</i>	
Number of global, local, or all element declarations	#EL _g , #EL _l , #EL
Number of global, local, or all complex-type definitions	#CT _g , #CT _l , #CT
Number of global, local, or all simple-type definitions	#ST _g , #ST _l , #ST
Number of global, local, or all model-group definitions	#MG _g , #MG _l , #MG
Number of global, local, or all attribute-group definitions	#AG _g , #AG _l , #AG
Number of global, local, or all attribute declarations	#AT _g , #AT _l , #AT
#EL _g + ... + #AT _g	#GLOBAL
<i>McCabe complexity for XSD</i>	
McCabe cyclometric complexity	MCC
<i>Depth and breadth of content models</i>	
Code-oriented and instance-oriented breadth	
Code-oriented and instance-oriented depth	

Figura 3.1: Métricas de XML apresentadas em [RL05].

XML para analisarmos e compararmos os ficheiros produzidos por cada formato de dados, como se pode observar, apresentadas na figura 3.1, .

4

Avaliação Qualitativa

4.1 Introdução

O objetivo principal deste capítulo não é substituir-se à especificação do formato de dados, mas antes **apontar, de forma sistemática e segundo determinadas características descritas no capítulo 1, se o formato tem ou não a capacidade apontada na característica em causa e de que forma ele a suporta**. Alerta-se para o facto de que neste capítulo as fontes escolhidas para servirem de análise, foram respetivamente a documentação oficial com a especificação de cada um dos formatos. Dado a esse facto, tudo o que neste capítulo, diga respeito à especificação dos formatos de dados, tem por base a documentação oficial da especificação do respetivo formato.

4.2 Exemplo prático

Antes de se iniciar o estudo aprofundado das especificações de cada formato de dados, procedeu-se à criação de um caso de prático, que permitisse servir de âncora ao longo do estudo sistemático dos diversos formatos. A experiência adquirida com este caso de estudo foi muito enriquecedora e revelou-se muito importante no levantamento das características a analisar dos diversos formatos de dados, para a comparação qualitativa das especificações dos formatos.

O exemplo prático utilizado assentou num cenário convencional no ambiente militar, no qual se opõem duas forças no terreno (*OGCForces* e *NVGForces*), com localização na área a norte da Ericeira junto ao oceano Atlântico (ver Figura 4.1). O significado da simbologia militar utilizada é irrelevante para o nosso estudo, contudo podemos dizer que os símbolos a verde são *forças amigas*, a vermelho *forças inimigas* e a verde elementos

identificados como *elementos conhecidos*.

O cenário foi georreferenciado recorrendo a um editor de SIG, o Udig¹. Depois, utilizando um servidor de SIG, o GeoServer², converteu-se o cenário georreferenciado em formato Shapefile³, para os diferentes formatos: KML, GML, SVG; para o formato NVG, a situação foi um pouco mais difícil, pois nenhum dos servidores de SIG disponíveis na Internet trabalham com esse formato, o que levou a que esse ficheiro em formato NVG fosse produzido num editor de texto.

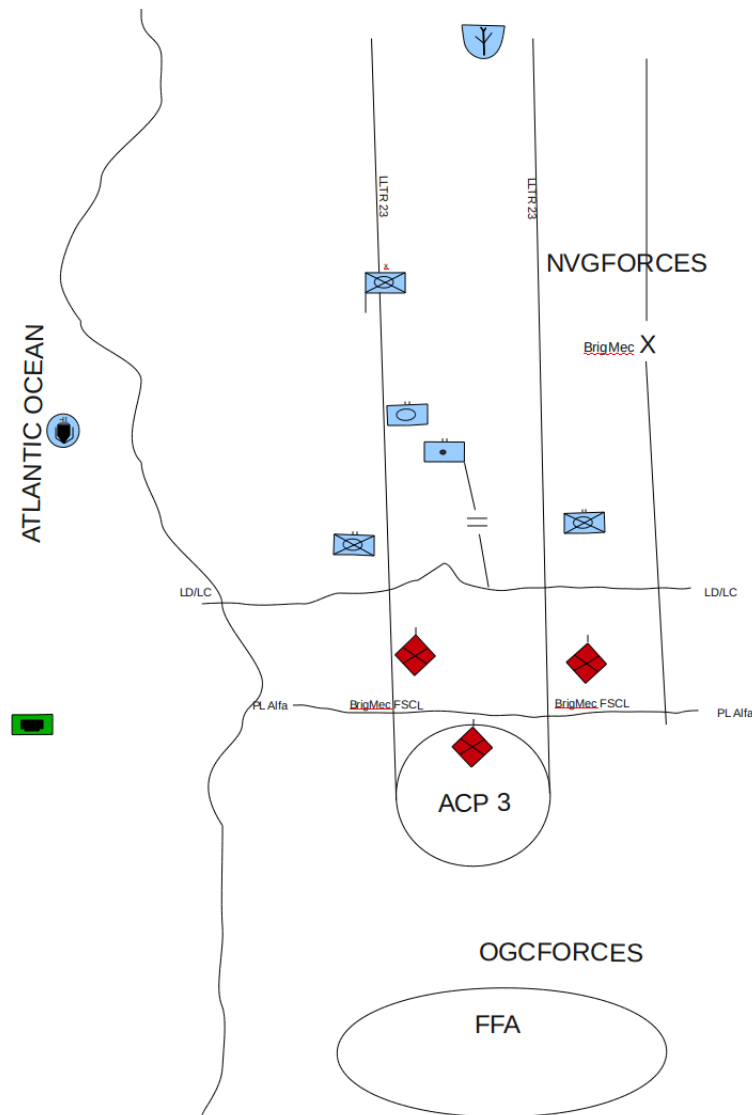


Figura 4.1: Cenário utilizado para o caso de estudo [oD97]

¹<http://udig.refrations.net/>

²<http://geoserver.org/display/GEOS/What+is+Geoserver>

³Formato desenvolvido pela Esri com a extensão .shp

4.2.1 NATO Vector Graphics[NC310b]

Uso de XML Schema para definição do documento O NVG utiliza o *XML Schema* para definir uma estrutura de *tags* com elementos e atributos aninhados de acordo com:

Namespace	http://tide.act.nato.int/schemas/2009/10/nvg
Schema	http://tide.act.nato.int/schemas/2009/10/nvg/nvg.data.1.5.xsd

Tabela 4.1: *XML Schema* do formato de dados NVG.

A versão atual do *XML Schema* que define o formato de dados NVG é *nvg.data.1.5.xsd*. Este *XML Schema* tem a particularidade de ser relativamente pequeno em relação aos restantes, com 526 linhas de código.

Sistema de Coordenadas de Referência optativo O formato de dados NVG não permite qualquer opção no que se refere ao seu Sistema de Coordenadas de Referência. O seu sistema de coordenadas é fixo: *World Geodetic System*⁴ (WGS84). Neste sistema, as unidades posicionais são expressas em graus decimais, onde a parte inteira é interpretada como graus e a parte decimal interpretada com minutos e segundos.

Utilização da terceira dimensão na localização de um ponto O sistema de Coordenadas Reais é expresso em graus e segundo dois eixos: Longitude e Latitude. Os valores da Longitude variam de -180° a 180° e os valores da Latitude variam de -90 a 90. Não existe a possibilidade de introdução do valor para a terceira dimensão. Este formato não permite a utilização da terceira dimensão na localização de um ponto.

Suportar a utilização do Sistema Internacional de Unidades O formato de dados NVG utiliza para a medição dos valores da velocidade a unidade de medida, “nó”; e para a medição do valor dos ângulos planos a unidade de medida “grau”. Estas unidades de medida não pertencem ao sistema internacional (SI) de medidas. As medidas correspondentes no SI, são: grau - radiano; nó - m/s. Conclui-se que o formato NVG não suporta o SI.

Definição de Estilos No formato NVG, a informação referente ao estilo atribuído aos dados, não está separada dos dados. O estilo a aplicar aos dados é especificado recorrendo ao uso de estilos CSS⁵ (Cascading Style Sheet), e cada estilo é associado a um único elemento. Existem três tipos de estilos que podem ser utilizados para atribuir estilo ao elementos, são eles: *Line Style* atribui um estilo a uma dada linha, *Area Style* atribui um estilo a uma dada área, e *Text Style* atribui um estilo a um dado texto. Conclui-se que o NVG não separa dos dados, a informação referente ao estilo.

⁴<http://earth-info.nga.mil/GandG/wgs84/index.html>

⁵<http://www.w3.org/Style/CSS/>

Uso de Simbologia Militar Este formato permite grande facilidade e simplicidade na utilização de simbologia militar necessária aos ambientes militares. O elemento `<point>` tem um atributo *symbol* que é usado para associar uma dada figura, à localização especificada pelo ponto. O formato NVG suporta os seguintes standards de simbologia militar⁶: *Allied Procedural Publication 6A (APP-6A)*[NC308a], *Allied Procedural Publication 6B (APP-6B)*[NC308b] e o *MIL-STD-2525B Common Warfighting Symbology*[NC309]. O NVG é um formato que permite a utilização, quer de símbolos baseados num ponto apenas, quer de símbolos que para a sua definição necessitam de vários pontos.

O NVG suporta ainda a criação de novos símbolos que venham a ser necessários, usando o atributo *symbol*, que permite a utilização de novos documentos com Simbologia Militar. Para além da utilização de símbolos pré-concebidos nos standards apresentados, o NVG permite ainda a utilização de recursos externos para um dado símbolo com recurso ao valor *icon:url*.

Elementos do tipo geométrico O formato de dados NVG tem 21 tipos de elementos diferentes, dez dos quais dedicados à representação de elementos geométricos. Os elementos geométricos que fazem parte do formato de dados NVG são os seguintes: `<point>`, `<multipoint>`, `<circle>`, `<ellipse>`, `<polyline>`, `<arrow>`, `<corridor>`, `<polygon>`, `<arc>`, `<arcband>`. Vale a pena referir, pela sua especificidade as seguintes primitivas:

- `<arrow>` elemento que define uma seta usada para representar movimento nesse sentido;
- `<corridor>` elemento que define um corredor⁷ para ser apresentado nas coordenadas geo-espaciais dadas;
- `<arcband>` elemento que representa uma área contida entre dois círculos concêntricos e duas radiais desses círculos, compreendido entre um determinado ângulo inicial e ângulo final;

Flexibilidade na definição dos dados No formato de dados NVG, a flexibilidade na definição dos dados é conseguida através do elemento `<content-item>`, que é um elemento que permite utilizar dentro dele elementos pertencentes a outros *namespace* que não o do NVG.

Uso de meta-informação Esta possibilidade é materializada pelo elemento `<metadata>`, que associa meta-informação ao elemento associado. A definição dos dados, para descrever meta-informação por parte do utilizador, segue determinadas recomendações que

⁶Nos últimos anos tem havido por parte do comando da NATO um esforço de integração e normalização ao nível da simbologia a utilizar na NATO, desse esforço resultou a adoção como norma, dos seguintes documentos: APP-6A, o documento oficial da NATO para a simbologia militar; o APP-6B, uma nova versão do AAP-6A; e o MIL-STD-2525B é o documento oficial para a simbologia militar dos Estados Unidos da América, e também é muito utilizado na NATO.

⁷Um corredor é uma medida de coordenação que representa um volume do espaço aéreo no Campo de Batalha, que permite a coordenação de trânsito aéreo no seu interior.

indicam que os dados devem ser definidos utilizando para isso formatos baseados em XML *Schemas*⁸ ou DTD⁹, e que recorram ao uso de entidades definidas no espaço de nomes `xmlns:dcterms="http://purl.org/dc/terms/"` como podemos ver no XML *Schema* definido para o formato NVG versão 1.5.0[NC310b].

```
<?xml:version="1.0" encoding="utf-8"?>
<xsd:schema xmlns="http://tide.act.nato.int/schemas/2009/10/nvg" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:dcterms="http://purl.org/dc/terms/" targetNamespace="http://tide.act.nato.int/schemas/2009/10/nvg"
  elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.5.0">
  <xsd:annotation>
    <xsd:documentation>
      Schema for NVG Data Format v1.5
      See http://tide.act.nato.int/mediawiki/index.php/NATO_Vector_Graphics_%28NVG%29_Data_Format
      for NVG Data Format document on TIDEpedia
    </xsd:documentation>
  </xsd:annotation>
  <xsd:import namespace="http://purl.org/dc/terms/" schemaLocation="metadata.xsd"/>
  <xsd:include schemaLocation="nvg.types.1.5.xsd"/>
  ...
```

Figura 4.2: Cabeçalho do XML *Schema* que define o formato de dados NVG 1.5.0

No formato NVG, foi também criado um elemento específico para permitir introduzir semântica ao nível dos dados. O elemento `<a>` possui um atributo especial `<href>` que aponta para um local que contém conteúdo HTML, e através deste mecanismo pode-se introduzir semântica num dado elemento, definida num dado URL.

Extensibilidade do formato O formato de dados NVG permite ser estendido ao nível dos elementos. A extensibilidade ao nível do elemento é conseguida recorrendo a diferentes formas:

- Estender os atributos de um qualquer elemento recorrendo ao atributo opcional `{any}`;
- Enviar meta-informação não prevista no formato de dados recorrendo ao elemento opcional `<ExtendedData>`;
- Usar o valor `icon:url` no atributo `symbol`, para endereçar um recurso que não faça parte da simbologia normalizada.

Herança O formato NVG não incorpora o conceito de herança entre os seus elementos.

Definição de dados pelo utilizador O formato de dados NVG é um formato que define os seus dados à custa de um XML *Schema* perfeitamente definido. No entanto, possibilita a utilização de um elemento `<Schema>` que permite ao utilizador definir novos elementos do tipo simples, definidos em `xsd:anySimpleType`.

⁸<http://www.w3.org/XML/Schema>

⁹<http://www.w3.org/QA/2002/04/valid-dtd-list.html>

Utilização de recursos externos Este formato de dados permite a utilização de recursos externos para a visualização de um dado símbolo com a atribuição do valor *icon:url* onde o *url* especifica o caminho do recurso externo. No entanto, o formato de dados NVG, apenas admite recursos externos do tipo imagens no formato *GIF*¹⁰ ou *PNG*¹¹.

Controlo sobre a visualização O formato NVG não prevê qualquer tipo controlo sobre a visualização dos seus elementos.

Capacidade de 3-D O formato NVG não prevê qualquer tipo de especificação nem visualização 3-D dos seus elementos.

Gestão dos recursos No formato NVG não se prevê nenhum mecanismo que permita ajudar na gestão dos recursos a ser visualizados.

Expressividade no domínio dos Sistemas de Comando e Controlo A expressividade na representação de entidades geográficas no formato NVG é a soma: da expressividade dada pelo uso da Simbologia Militar, das suas primitivas geométricas para representação de entidades geométricas, do elemento *<text>* para representar texto, e do elemento *ExtendedData* para descrever mais informação acerca da entidade em questão.

4.2.2 Scalable Vector Graphics

4.2.2.1 Formato de Dados[Gro09]

Uso de XML Schema para definição do documento A linguagem de definição de dados utilizada pelo formato de dados SVG é o *eXtensible Markup Language* (XML), da W3C, de acordo com: tabela 4.2 (DTD): A utilização de DTD na definição de dados, facilita a com-

Namespace	http://www.w3.org/2000/svg
DTD	http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd

Tabela 4.2: XML Schema do formato de dados SVG[Gro09].

preensão humana dos mesmos, mas introduz limitações na expressividade dos próprios dados relativamente ao XML Schema, como por exemplo, na definição do número de instâncias que cada elemento permite. A validação de um documento definido por um DTD é sempre limitada. Ao contrário do XML Schema, o DTD não utiliza o XML para se auto definir, mas, embora ainda se encontre muito disseminado por diversos documentos na Internet, não é norma, e por conseguinte, a sua utilização é desencorajada.

O DTD usado na definição do formato SVG é o *svg11.dtd*, e apresenta uma dimensão relativamente grande, 5588 linhas de código.

¹⁰Graphics Interchange Format

¹¹Portable Network Graphics

Sistema de Coordenadas de Referência optativo O formato de dados SVG, recorre às recomendações da OGC, *Definition of Coordinate Reference System*¹², para especificar o Sistema de Coordenadas de Referência (SCR) utilizado na geração do mapa. A definição do SRC é realizada no elemento *metadata* e consiste numa descrição *Resource Description Framework*¹³ (RDF) do mesmo. O mapeamento é realizado da seguinte forma: o primeiro eixo descrito no SRC do mapa corresponde ao eixo das abcissas (x) do SVG; o segundo eixo descrito no SRC do mapa corresponde ao eixo das ordenadas (y) do SVG. É de notar, que o sistema de eixos no formato SVG tem o ponto central no topo superior esquerdo, sendo que, a coordenada y cresce para baixo. Esta particularidade obriga a efetuar transformações no próprio sistemas de coordenadas. Do apurado, conclui-se que o formato permite usar diferentes SRC, dando opção ao utilizador de qual o SRC a utilizar.

Utilização da terceira dimensão na localização de um ponto Não permite a utilização da terceira dimensão. Os dados provenientes deste formato apenas se aplicam a um plano.

Utilização de unidades do Sistema Internacional de Unidades Este formato permite a utilização de unidades de medida dentro do Sistema Internacional de Unidades (SI), oferecendo no entanto, ao utilizador, a possibilidade de utilizar unidades fora do SI.

Definição de Estilos Este formato permite a aplicação de estilos recorrendo à linguagem de estilo *Cascading Style Sheets* (CSS). A técnica utilizada para introduzir estilos, recorre ou à atribuição de valores a atributos CSS que estão associados a um elemento gráfico ou a utilização de classes CSS, definida em ficheiros externos ao código. Portanto, o formato SVG permite a definição de estilos, e permite separar a visualização da representação.

```

1 <svg x="200" contentType="text/ecmascript" y="200".
2   xmlns:xlink="http://www.w3.org/1999/xlink"
3   baseProfile="full" zoomAndPan="magnify" contentType="text/css".
4   viewBox="0 0 520 520" preserveAspectRatio="xMidYMid meet".
5   xmlns="http://www.w3.org/2000/svg" version="1.0">
6   <defs>
7     <symbol id="symbolImage" overflow="visible">
8       <image x="20" y="20" width="300" height="80".
9       xlink:href="/home/mestre/desktop/NVG/simbolos/PostoComandoBrig.jpg"/>
10    </symbol>
11  </defs>
12  <g id="PostoComandos">
13    <use id="Tactico" x="100" y="200" xlink:href="#symbolImage"/>
14    <use id="Principal" x="150" y="100" xlink:href="#symbolImage"/>
15  </g>
16 </svg>

```

Figura 4.3: Código exemplificativo da definição de um Símbolo

¹²<http://www.opengeospatial.org/standards>

¹³<http://www.w3.org/RDF/>

Uso de Simbologia Militar Não possui qualquer tipo de mecanismo específico para o uso da simbologia militar, no entanto esta incapacidade pode ser simulada (com bastantes limitações) com a utilização do seu elemento `<def>` em conjugação do seu elemento `<symbol>` que permite definir um símbolo recorrendo a importação de uma imagem, como podemos ver no seguinte exemplo, o código definido na figura 4.3 dá origem à visualização da figura 4.4. Este mecanismo é baseado na localização de símbolos baseado num ponto.

Elementos do tipo geométricos O SVG tem disponíveis 6 elementos geométricos básicos e mais um elemento geométrico especial que é o elemento `path`. Os seis elementos básicos são os seguintes: `Rectangle`, `Circle`, `Line`, `Polyline`, `Polygon` e `Ellipse`. Recorrendo a estas seis primitivas em conjunto com a primitiva `Text`, pode-se construir qualquer elemento gráfico, como por exemplo um símbolo militar.

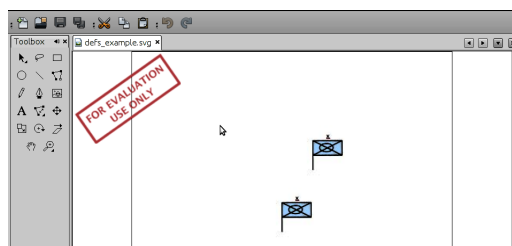


Figura 4.4: Visualização do processamento do código 4.3

O elemento geométrico `Path`, é especial dado que através dele podemos substituir qualquer um dos 6 elementos básicos. Este elemento pode conter uma série de diferentes capacidades que o tornam muito versátil e poderoso, nomeadamente aceitação de comando que permitem desenhar, *cubic splines*, *quadratic splines*, *elliptical curves* e *holes*; que permitem de forma aninhada novos comandos *sub-path*, *lineTo*, etc. Como podemos visualizar no seguinte exemplo 4.5.

```

1 <svg viewBox="0 0 320 220">
2 <rect width="320" height="220" fill="white" stroke="black" />
3 <g transform="translate(10 10)">
4 <g stroke="red" stroke-width="3" fill="lime">
5 <path d="M 0 112
6 L 20 124 L 40 129 L 60 126 L 80 120 L 100 111 L 120 104
7 L 140 101 L 164 106 L 170 103 L 173 80 L 178 60 L 185 39
8 L 200 30 L 220 30 L 240 40 L 260 61 L 280 69 L 290 68
9 L 288 77 L 272 85 L 250 85 L 230 85 L 215 88 L 211 95
10 L 215 110 L 228 120 L 241 130 L 251 149 L 252 164 L 242 181
11 L 221 189 L 200 191 L 180 193 L 160 192 L 140 190 L 120 190
12 L 100 188 L 80 182 L 61 179 L 42 171 L 30 159 L 13 140Z"/>
13 </g>
14 </g>
15 </svg>

```

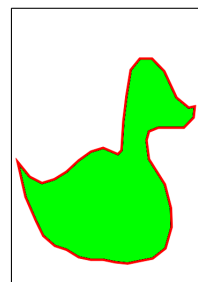


Figura 4.5: Definição do elemento `Path` e a sua respectiva visualização

Flexibilidade na definição dos dados Este formato não prevê nenhum mecanismo de flexibilidade na definição dos seus dados para além da sua leve e permissiva estrutura.

Uso de meta-informação Este formato recorre ao elemento `<metadata>` para introduzir semântica nos seus dados. Este elemento pode ser introduzido, em qualquer lugar do

formato. O elemento permite a utilização de qualquer linguagem de meta-informação compatível com XML. Podemos ver um exemplo como utilizar o a linguagem RDF para introduzir semântica nos dados na figura 4.6.

```

1 <svg width="4in" height="3in" version="1.1"
2   xmlns="http://www.w3.org/2000/svg">
3   ...
4   <metadata>
5     <rdf:RDF
6       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
7       xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
8       xmlns:dc="http://purl.org/dc/elements/1.1/">
9     <rdf:Description about="http://example.org/myfoo"
10      dc:title="MyFoo Financial Report"
11      dc:language="en">
12     <dc:creator>
13     <rdf:Bag>
14     <rdf:li>Irving Bird</rdf:li>
15     <rdf:li>Mary Lambert</rdf:li>
16     </rdf:Bag>
17     </dc:creator>
18     </rdf:Description>
19     </rdf:RDF>
20   </metadata>
21 </svg>

```

Figura 4.6: Exemplo da utilização de RDF num ficheiro SVG
[Gro09, text]

Extensibilidade do formato A extensibilidade no SVG efectua-se por duas formas: através da possibilidade de definição de *namespaces* exteriores em qualquer parte do conteúdo do ficheiro SVG, com a conseguinte utilização dos elementos constantes nesses *namespaces*; através da possibilidade de estender o próprio DTD que define a linguagem SVG no interior do ficheiro SVG.

Herança nos dados Este formato não prevê nenhuma forma de herança entre os dados do formato.

Utilização de recursos externos A utilização de recursos externos é utilizada no elemento `<image>` para carregar uma imagem exterior ao ficheiro, e é possibilitada pelo recurso aos elementos existentes no *namespace* `xmlns:xlink="http://www.w3.org/1999/xlink"`, nomeadamente o elemento `xlink:href`.

```

1 <image x="20" y="20" width="300" height="80"
2   xlink:href="http://jenkov.com/images/layout/top-bar-logo.png" />
3

```

Figura 4.7: Exemplo da utilização de um recurso exterior ao ficheiro SVG
[webf]

Controlo sobre a visualização O formato de dados SVG permite um elevado controlo na visualização gráfica dos seus elementos, conseguido recorrendo à introdução de código no interior do ficheiro que é interpretado por uma linguagem de scripting do tipo *ecmascript*¹⁴, compatível com *XML/DOM*¹⁵, que lhe permite responder a eventos e interação com o utilizador. Este formato possui ainda elementos de controlo definidos ao nível da animação da própria imagem. Para além disso, permite ainda definir o detalhe gráfico com que os objectos serão visualizados, recorrendo à definição do elemento *viewBox*.

Capacidade de 3D O formato SVG não prevê qualquer tipo de especificação nem visualização 3D dos seus elementos.

Gestão dos recursos O SVG não permite nenhuma gestão ao nível dos recursos.

Expressividade no domínio dos Sistemas de Comando e Controlo Este formato não possui elementos que lhe permitam descrever entidades geográficas para além das suas primitivas geométricas.

4.2.3 KML

Uso de XML Schema para definição do documento Como se pode ler em [Wil08] no paragrafo 5.3, KML usa a linguagem *W3C XML Schema* para descrever a gramática que definem as instâncias de dados KML.

O formato de dados KML utiliza uma estrutura de tags com elementos e atributos aninhados servindo-se da linguagem de definição de dados *eXtensible Markup Language* (XML), padrão da W3C, de acordo com o seguinte:

Namespace	http://www.opengis.net/kml/2.2
Schema	http://schemas.opengis.net/kml/

Tabela 4.3: XML Schema do formato de dados.[OGC08]

Na definição do *schema* para o KML, foram usados vários mecanismos à disposição na linguagem *XML Schema*, como é o caso da herança, e dos grupos de substituição. A versão mais atual do *XML Schema* para o KML é a *ogckml22.xsd*.

Todos os elementos são *case-sensitive* e devem ser introduzidos pela ordem correta indicada na especificação. Uma instância do KML é processada da mesma forma que os documentos HTML são processados nos *browsers*. Pode também ser apresentado sob a forma de um ficheiro do tipo *.kmz* que mais não é que uma versão comprimida do ficheiro *.kml* com algoritmo de compressão *Zip*¹⁶.

¹⁴<http://www.ecmascript.org/docs.php>

¹⁵<http://www.w3.org/DOM/>

¹⁶<http://www.pkware.com/>

Sistema de Coordenadas de Referência optativo As duas primeiras componentes, latitude e longitude são definidas pelo *World Geodetic System*¹⁷ de 1984 (WGS84). A componente vertical, altitude, é medida recorrendo ao *WGS84 EGM96 Geoid vertical datum*¹⁸. Estas são definidas através do *GML Coordinate Reference System*[Gro07] com o identificador *LonLat84_5773*. Este formato tem um sistema de coordenadas fixo que não permite ao utilizador optar por um Sistemas de Coordenadas de Referencia diferente.

Utilização da terceira dimensão na localização de um ponto O formato KML usa um sistema de coordenadas geográficas com capacidade de utilização da terceira dimensão, no entanto, o registo da terceira dimensão (altitude) é de carácter opcional. As suas coordenadas são referenciadas pelos seguintes valores: longitude, latitude e altitude, por esta ordem.

Suportar a utilização do Sistema Internacional de Unidades O formato de dados KML utiliza para a medição do valor dos ângulos, a unidade de medida o “Grau”. Esta unidade de medida não pertence ao sistema internacional de medidas. Concluindo-se que este formato não suporta o Sistema Internacional de Unidades.

Definição de Estilos Existem dois tipos de elementos com os quais podemos atribuir estilos, são eles: *kml:Style* e *kml:StyleMap*. Na linguagem KML temos duas formas de aplicar cada estilo:

- *inline style* aplica-se um estilo a um único elemento;
- *shared style* define-se um estilo, atribui-se-lhe um identificador e utiliza-se esse estilo em vários elementos referenciando o identificador desse estilo.

O formato KML não separa os dados da sua representação.

Uso de Simbologia Militar Neste formato de dados não está prevista nenhuma simbologia militar pré-definida. No que toca a simbologia no formato KML existe um elemento *kml:Placemark* do tipo *kml:AbstractFeatureGroup* que, associado a um outro elemento *kml:Point*, pode ser desenhado com um *icon* para marcar o elemento *kml:Placemark* na visualização geográfica, esse *icon* é um subelemento que pode ser personalizado através da utilização de outro elemento *href* que efetua a importação de um imagem disponível na *Web*. O elemento *kml:Placemark*, por sua vez, permite através do elemento *<kml:description>*, introduzir no marcador criado, qualquer informação do tipo HTML (introduzir imagens, conteúdo de páginas web, etc). O KML, é um formato que utiliza a localização baseada num ponto apenas.

¹⁷<http://earth-info.nga.mil/GandG/wgs84/index.html>

¹⁸<http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm96/egm96.html>

Elementos do tipo geométricos O elementos geométricos definidos neste formato são os seguintes: *kml:MultiGeometry*, *kml:Point*, *kml:LineString*, *kml:LinearRing*, *kml:Polygon*, *kml:Model*. As primitivas que se destacam pela diferença neste formato são:

kml:LineString Define uma lista de um ou mais segmentos de recta contínuos. Este elemento permite a utilização de um sub-elemento *kml:extrude* que possibilita colocar segmentos de recta afastados da superfície terrestre e unir a mesma com a sua projecção na superfície terrestre desenhando assim algo parecido com uma cerca.

kml:LinearRing Define um elemento *kml:LineString* fechado sem que qualquer das linhas definidas se cruzem.

kml:Polygon O elemento *kml:Polygon* obriga à definição de um anel de contorno exterior recorrendo ao elemento *kml:outerBoundaryIs* e zero ou mais anéis interiores, *kml:innerBoundaryIs*. Cada a anel é definido à custa do elemento *kml:LinearRing*.

kml:Model Especifica a localização e orientação de uma textura aplicada a um recurso 3D. A estrutura e a aparência da textura do objecto 3D não é definida pela especificação.

Flexibilidade na definição dos dados Como foi referido, o formato de dados KML é um formato onde existem tipos de dados abstratos. Estes tipos de dados abstratos são muito úteis porque, entre outras coisas, permitem a utilização de **grupos de substituição**. Ou seja, para cada elemento do tipo abstrato pertencente ao núcleo do formato de dados, existem diversos elementos diferentes que, dentro do mesmo grupo de substituição, o podem substituir. Por exemplo, o elemento *kml:AbstractFeatureGroup* representa um grupo de substituição para objecto do tipo *kml:AbstractFeatureType*, logo todos os tipos de dados definidos com o tipo *kml:AbstractFeatureType* poderão ser utilizados onde apareça a utilização do elemento *kml:AbstractFeatureGroup*.

A seguir apresenta-se o resumo dos elementos KML que servem como cabeça de um grupo de substituição para o seu correspondente tipo (entre parênteses). Esta hierarquia permite identificar na estrutura de um qualquer ficheiro KML, onde se situa um dado elemento e que tipos de elementos podemos esperar dentro do mesmo. A lista, segundo a especificação é a seguinte:

- *kml:AbstractObjectGroup* (*kml:AbstractObjectType*)
 - *kml:AbstractFeatureGroup* (*kml:AbstractFeatureType*)
 - *AbstractContainerGroup* (*kml:AbstractContainerType*)
 - * *AbstractOverlayGroup* (*kml:AbstractOverlayType*)
 - * *AbstractGeometryGroup* (*kml:AbstractGeometryType*)
 - *AbstractStyleSelectorGroup* (*kml:AbstractStyleSelectorType*)
 - *AbstractSubStyleGroup* (*kml:AbstractSubStyleType*)
 - * *AbstractColorStyleGroup* (*kml:AbstractColorStyleType*)

- AbstractLatLonBox (kml:AbstractLatLonBoxType)
- AbstractViewGroup (kml: AbstractViewType)
- AbstractTimePrimitiveGroup (kml:AbstractTimePrimitiveType)

Para além desta funcionalidade, existe ainda uma outra que permite flexibilizar a definição dos dados designada por **Entidades de Substituição**. As entidades de substituição são utilizadas como um modelo de mecanismo de substituição nos elementos: *kml:BalloonStyle* e *kml:text*. Ou seja, podemos ver estas entidades de substituição como variáveis colocadas no código que depois tomarão o seu devido valor definido num elemento do tipo *kml:schema* associado ao *kml:AbstractFeatureGroup*.

Uso de meta-informação Neste formato, o uso de meta-informação é realizado pelo elemento *kml:Description*, que permite ao utilizador inserir informação semântica acerca do elemento *kml:Placemark* que estamos a referir. Esta informação pode ser introduzida recorrendo à linguagem de marcação *HyperText Markup Language*¹⁹ (HTML), da W3C.

Extensibilidade do formato A extensão na linguagem KML é feita de duas formas, por Composição e por Herança. A extensão por Composição é realizada recorrendo à substituição de elementos abstratos, cujo o tipo também ele é abstrato. Os elementos abstratos funcionam como *placeholders* para elementos que os substituirão no *XML schema* definido. A extensibilidade por Herança é conseguida atribuindo aos filhos características dos pais.

Herança nos dados O formato de dados KML utiliza o mecanismo de herança entre os seus dados. No formato KML alguns tipos de elementos são derivados de outros tipos os quais são apelidados de pai destes. Um elemento derivado herda todos os elementos dos seu pai e ainda pode adicionar os seus próprios elementos.

Definição de dados pelo utilizador O formato de dados KML permite que o utilizador defina os seus próprios dados. Esta possibilidade é materializada pelo elemento *kml:ExtendedData*, e oferece três formas:

- Adicionar um elemento arbitrário não tipificado do tipo par (chave/valor) recorrendo ao *kml:Data element*;
- Adicionar instâncias de campos tipificados definidos no elemento *kml:Schema* definido pelo utilizador;
- Incluir qualquer conteúdo definido num outro espaço de nomes que o espaço de nomes definido pelo KML.

Utilização de recursos externos Este formato de dados permite a utilização de recursos externos. Este acesso é realizado recorrendo ao uso do elemento *kml:NetworkLink* do tipo *kml:Abstract FeatureGroup* que permite referenciar um recurso local ou remoto na rede.

¹⁹<http://www.w3.org/XML/>

Controlo sobre a visualização O formato de dados KML permite um controlo efectivo sobre a forma como a visualização se irá efectuar. Nesse intuito, foi criado um elemento *kml:Region* que define o controlo da visualização e do nível de detalhe a mostrar. Para isso, *kml:Region* especifica um outro elemento *kml:LatLonAltBox* que descreve a área de interesse definida por coordenadas geográficas e altitude. Além disso, *kml:Region* contém como filho o elemento *kml:Lod*, responsável por definir o alcance (distancia) da validade do elemento *kml:Region* associado nos termos da dimensão do tela projectada. Um elemento *kml:Region* fica activo, quando o valor do elemento *bounding box* está dentro da visão do utilizador, e quando os requisitos especificados no *kml:Lod* estão presentes (Lod - level of detail).

Capacidade de 3-D A capacidade 3-D está presente no formato KML, quer ao nível da definição da localização com a utilização da altitude nas suas coordenadas; quer ao nível da visualização com a utilização de um sistema de interpolação tridimensional. A definição, entre outros elementos, do elemento geométrico *kml:Model* que permite aplicar uma textura a um objeto tridimensional, e do elemento *kml:Camera* que especifica a posição e orientação de uma câmara virtual, segundo a qual visualizamos os objetos.

Gestão dos recursos No formato de dados KML existe uma preocupação com a gestão dos recursos disponíveis. O carregamento de recursos através do *kml:Link* ou do *kml:Icon*, só é efectuado quando o elemento *kml:Region* estiver no modo *active*, pois de outro modo era estar a realizar um operação, pesada para o sistema, em que os recurso obtidos não iriam ser utilizados.

Expressividade no domínio dos Sistemas de Comando e Controlo A expressividade no formato de dados KML para representar entidades reside na utilização do elemento *kml:PlaceMark*, que tem a estrutura apresentada na Figura 4.8. Nesta estrutura existem alguns elementos que merecem destaque pela sua função na construção do elemento *kml:Placemark*, são eles:

- *LookAt* Indica o local na superfície terrestre de onde estamos a observar;
- *Point* Indica o local na superfície terrestre da entidade que estamos a descrever;
- *StyleURL* Refere um estilo a atribuir à visualização do entidade que estamos a expressar, que pode existir remotamente;
- *Description* Serve para descrever a entidade, e permite recorrer à linguagem HTML para o fazer

Este formato não tem nenhum mecanismo que lhe confira uma maior expressividade no domínio dos Sistemas de Comando e Controlo que noutra qualquer domínio.

```

<kml:Placemark
id="ID [0..1]"
targetId="NCName [0..1]"
  <kml:ObjectSimpleExtensionGroup> ... </kml:ObjectSimpleExtensionGroup> [0..*]
  <kml:name> ... </kml:name> [0..1]
  <kml:visibility> ... </kml:visibility> [0..1]
  <kml:open> ... </kml:open> [0..1]
  <atom:author> ... </atom:author> [0..1]
  <atom:link> ... </atom:link> [0..1]
  <kml:address> ... </kml:address> [0..1]
  <xal:AddressDetails> ... </xal:AddressDetails> [0..1]
  <kml:phoneNumber> ... </kml:phoneNumber> [0..1]
  <kml:snippet> ... </kml:snippet> [0..1]
  <kml:description> ... </kml:description> [0..1]
  <kml:AbstractViewGroup> ... </kml:AbstractViewGroup> [0..1]
  <kml:AbstractTimePrimitiveGroup> ... </kml:AbstractTimePrimitiveGroup> [0..1]
  <kml:styleUrl> ... </kml:styleUrl> [0..1]
  <kml:AbstractStyleSelectorGroup> ... </kml:AbstractStyleSelectorGroup> [0..*]
  <kml:Region> ... </kml:Region> [0..1]
  <kml:ExtendedData> ... </kml:ExtendedData> [0..1]
  <kml:AbstractFeatureSimpleExtensionGroup> ...
</kml:AbstractFeatureSimpleExtensionGroup> [0..*]
  <kml:AbstractFeatureObjectExtensionGroup> ...
</kml:AbstractFeatureObjectExtensionGroup> [0..*]
  <kml:AbstractGeometryGroup> ... </kml:AbstractGeometryGroup> [0..1]
  <kml:PlacemarkSimpleExtensionGroup> ... </kml:PlacemarkSimpleExtensionGroup>
[0..*]
  <kml:PlacemarkObjectExtensionGroup> ... </kml:PlacemarkObjectExtensionGroup>
[0..*]
</kml:Placemark>

```

Figura 4.8: Exemplo da estrutura do elemento *kml:placemark* [Wil08, pág. 58]

4.2.4 GML[Gro07]

Uso de XML Schema para definição do documento Segundo [Gro07], GML é uma gramática XML escrita em XML Schema para a descrição de esquemas de aplicação, bem como o transporte e armazenamento de informações geográficas. O GML especifica a codificação XML de uma série de classes conceituais definidas na série ISO 19100 da *International Standards* e da *OpenGIS Abstract Specification* em conformidade com estes standards e especificações[Gro07]:

Namespace	http://www.opengis.net/gml/3.2
Schemas	http://schemas.opengis.net/gml

Tabela 4.4: *Namespace* e os *Schemas* do formato de dados GML

Para se utilizar o GML como formato de dados, um utilizador começa por definir a sua própria linguagem para descrever os seus dados, utilizando para isso as regras definidas na linguagem (meta-linguagem) GML. Ou seja, como podemos observar pela figura 4.9, inicialmente define-se o esquema em XML, recorrendo ao XML schema para o “domínio concreto de aplicação”, *Application Schema*. O esquema agora definido, alberga os elementos, os atributos, e a estrutura usados num documento que define os dados de um determinado domínio de aplicação[Far07, pág. 21]. Por fim, são criados os documentos XML de acordo com o XML schema de aplicação definido.

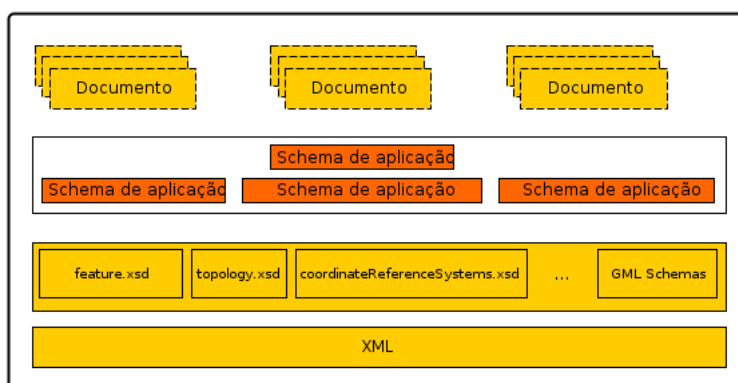


Figura 4.9: GML como meta-linguagem na definição de esquemas de domínio de aplicação.

Para ajudar na definição de esquemas específicos de domínio de aplicação, foi realizado um grande esforço em desenvolver padrões que contemplem a maioria das necessidades existentes na comunidade que utiliza informação georreferenciada. Os padrões principais de esquemas GML publicados pela OGC são: *BasicTypes*, *Topology*, *Coordinate Reference Systems*, *Temporal Information and Dynamic Feature*, *Definitions and Dictionaries*, *Metadata* e *Coverages*.

Para definir os esquemas de aplicação (ficheiros .xsd com o formato dos dados) deve ter-se em atenção as seguintes regras:

- Todos os tipos utilizados na definição de dados, são subtipos dos correspondentes tipos definidos na meta-linguagem GML;
- O esquema de aplicação tem de importar o(s) esquema(s) do GML que contém os tipos de dados que forem referenciados, ou esquemas que referenciem estes;
- Os esquemas de aplicação não pode efetuar qualquer alteração no que diz respeito ao nome, definição ou tipo de dados aos elementos do GML;
- As definições dos tipos de dados abstratos de dados podem ser estendidas ou restringidas;
- Para que os dados estruturados no formato GML sejam processados, estes devem ser acompanhados do esquema de aplicação que os define;
- Os esquemas de aplicação não podem utilizar, como *targetNamespace*, o *XML Namespace* <http://www.opengis.net/gml/3.2>;

Para facilitar a compreensão da utilização dos esquemas de aplicação apresenta-se, de seguida um exemplo muito simples de um esquema de aplicação, o qual pretende representar as ruas de uma cidade (ver Figura 4.10). Como se observa no *XML Schema* apresentado na figura 4.10, os Tipos de Dados aqui definidos são, obrigatoriamente, sub-Tipos

```

1  <?xml:version="1.0" encoding="UTF-8"?>
2  <xs:schema targetNamespace="http://www.Salvador.pt" xmlns="http://www.Salvador.pt"
3  xmlns:gml="http://www.opengis.net/gml" xmlns:xs="http://www.w3.org/2001/XMLSchema"
4  elementFormDefault="qualified" attributeFormDefault="unqualified">
5  <xs:import namespace="http://www.opengis.net/gml"
6  schemaLocation="/home/mestre/desktop/GML/feature.xsd"/>
7  <xs:element name="City" type="gml:AbstractFeatureCollectionType"
8  substitutionGroup="gml:_FeatureCollection">
9  </xs:element>
10 <xs:element name="Street" type="StreetType" substitutionGroup="gml:_Feature"/>
11 <xs:complexType name="StreetType">
12 <xs:complexContent>
13 <xs:extension base="gml:AbstractFeatureType">
14 <xs:sequence>
15 <xs:element name="name" type="xs:string" maxOccurs="1"/>
16 <xs:element name="width" type="xs:integer"/>
17 <xs:element ref="gml:centerLineOf"/>
18 </xs:sequence>
19 </xs:extension>
20 </xs:complexContent>
21 </xs:complexType>
22 </xs:schema>

```

Figura 4.10: XML Schema definido para um domínio de aplicação concreto.

dos Dados definidos nos XML Schemas definidos para o GML pela OGC. Por exemplo, na linha sete, o elemento *City* é do tipo *gml:AbstractFeatureCollectionType* que está definido no ficheiro *feature.xsd*. Depois de existir um esquema definido, poderemos criar diversas instâncias do mesmo. No entanto, para efeitos de completude do exemplo, apresenta-se um ficheiro .gml com uma possível instância do esquema definido em 4.11.

```

1  <City xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.Salvador.pt"
2  xmlns:gml="http://www.opengis.net/gml"
3  xsi:schemaLocation="http://www.Salvador.pt/home/mestre/desktop/GML/schema_gml.xsd">
4  <gml:boundedBy>
5  <gml:Envelope>
6  <gml:pos>100 100</gml:pos>
7  <gml:pos>230 250</gml:pos>
8  </gml:Envelope>
9  </gml:boundedBy>
10 <gml:featureMember>
11 <Street gml:id="street 1">
12 <name>Av. Fontes Pereira de Melo</name>
13 <width>50</width>
14 <gml:centerLineOf>
15 <gml:LineString gml:id="line 2">
16 <gml:pos>130 200</gml:pos>
17 <gml:pos>200 220</gml:pos>
18 </gml:LineString>
19 </gml:centerLineOf>
20 </Street>
21 </gml:featureMember>
22 <gml:featureMember>
23 <Street gml:id="street 2">
24 <name>Av. da Liberdade</name>
25 <width>100</width>
26 <gml:centerLineOf>
27 <gml:LineString gml:id="line 3">
28 <gml:pos>130 100</gml:pos>
29 <gml:pos>130 200</gml:pos>
30 </gml:LineString>
31 </gml:centerLineOf>
32 </Street>
33 </gml:featureMember>
34 </City>

```

Figura 4.11: Exemplo do formato GML onde se representam ruas de uma cidade.

Sistema de Coordenadas de Referência optativo O formato de dados GML, não especifica nenhum sistemas de coordenadas obrigatório. Em vez disso deixa a opção ao utilizador de escolher o Sistema de Coordenadas de Referência (SCR). Essa opção é realizada através da utilização do atributo obrigatório *srsName*, em todos os tipos de dados

geométricos. Existem duas formas de indicar qual o SCR: indicando um número ou indicando um nome; se indicarmos um número ele servirá para através de um servidor de dicionário de SCR dizer qual o nome do documento que indica as características do SCR e o Datum utilizado.

Para além do utilizador poder facilmente escolher um SCR dos muitos já existentes na meta-linguagem do GML, este pode sempre definir o seu próprio Sistema de Coordenados de Referencia recorrendo ao *CRS Schemas*.

Utilização da terceira dimensão na localização de um ponto O formato GML usa um sistema de coordenadas geográficas com capacidade de utilização da terceira dimensão. As coordenadas são referenciadas por tuplos de valores. O registo do valor da terceira dimensão opcional.

Utilização de unidades do Sistema Internacional de Unidades No que diz respeito à utilização de unidade de medida no GML, este tem predefinido um tipo *BaseUnit* que contém as unidades de medida do Sistema Internacional, embora existam outros tipos também definidos. No entanto, deixa ao critério do utilizador a opção de utilizar o tipo de unidade de medida ou, se este assim o pretender, definir as suas próprias unidades de medida.

Definição de Estilos O formato de dados GML não especifica qualquer definição de estilos para os seus dados. O GML não se preocupa com a visualização dos mesmos, por isso deixa essa tarefa para linguagens de estilos como é o caso da *Styled Layer Descriptor* (SLD) da OGC ou XSLT da W3C.

Uso de Simbologia Militar O GML, tanto quanto se conhece à data, não existe nenhum *XML Schema standard* da OGC definido que contemple qualquer mecanismo específico para lidar com a simbologia militar.

Elementos do tipo geométrico Os elementos primitivos do tipo geométrico do formato GML, espalhado pelas suas quatro dimensões (0,1,2,3), são:

Dimensão 0 *Point* representa um ponto num sistema geográfico.

Dimensão 1 *Curve* representa um linha curva continua, tem um comprimento mensurável num qualquer sistema de coordenadas;

LineString representa uma curva especial, que consiste em segmentos interpolados linearmente entre cada dois pontos de coordenadas;

ArcString representa um segmento de curva definido com 3 pontos de coordenadas, que é utilizada interpolação *CircularArc3Points*;

Arc representa um arco com apenas 3 pontos de controlo;

Circle é um arco cujo início coincide com o final;

B-Spline é uma curva seccional mente polinomiais ou racionais paramétricos, descritos em termos de pontos de controle;

Bézier são *splines* polinomiais que usam polinômios de *Bézier* ou *Bernstein* para efeitos de interpolação.

GeodesicString é uma sequência de segmentos geodésicos.

Geodesic é um segmento de curva que utiliza a interpolação geodésica;

Dimensão 2 *Polygon* é uma superfície definida por um único *patches*²⁰ de superfície. A sua fronteira é coplanar e utiliza a interpolação planar no seu interior;

LinearRing é definido por no mínimo 4 tupulos de coordenadas e utilizam a interpolação linear entre eles. A primeira coordenada deve ser coincidente com a última;

Surface é composta de uma ou mais *patches* de superfície;

Ring é uma sequência de linhas que representam a fronteira de uma superfície;

PolyhedralSurface é uma superfície composta de *patches* polígonos conectados ao longo das suas curvas de fronteira comum;

TriangulatedSurface é uma superfície poliédrica que é composta apenas por triângulos;

Tin é uma superfície triangulada que usa o algoritmo *Delauny* ou um algoritmo similar;

PolygonPatch é um *patche* de superfície que é definido por um conjunto de curvas de contorno e uma superfície subjacente a qual estas curvas aderem. As curvas são complanares e o polígono usa interpolação planar na sua interior;

Triangle representa um triângulo como um *patche* de superfície com a fronteira exterior é um *LinearRing*. O número de pontos na *LinearRing* têm de ser 3;;

Rectangle representa um rectangle como um *patche* de superfície com a fronteira exterior é um *LinearRing*. O número de pontos na *LinearRing* têm de ser 4;

Cone representa um cone (implementa a *ISO 19107 GM_Cone*);

Cylinder representa um cilindro (implementa a *ISO 19107 GM_Cylinder*);

Sphere representa uma esfera (implementa a *ISO 19107 GM_Sphere*);

Dimensão 3 *Solid* é a base para a geometria a 3 dimensões (implementa a *ISO 19107 GM_Solid*). É definido por uma superfície exterior e uma superfície interior;

Shell representa um único componente conectado de um contorno sólido (conforme especificado na norma *ISO 19107:2003*).

²⁰A tradução para português pode ser feita para remendo, mancha ou fragmento.

Para além destes tipos primitivos GML, foi definido a noção de *Coverages(Grigs)*, que permite, entre outras coisas, uma forma de representação de imagens *raster* (imagens construídas a partir de mapa de pontos) no formato GML.

Flexibilidade na definição dos dados O GML é um formato onde existem tipos de dados abstratos. Estes tipos abstratos de dados são muito úteis porque, entre outras coisas, permitem a utilização de **grupos de substituição** (principais grupos de substituição: *gml:AbstractObject*, *gml:AbstractGML*, *gml:AbstractFeature*, *gml:AbstractValue*, *gml:AbstractCoverage*, *gml:AbstractTopology* e *gml:AbstractCRS*). Ou seja, para cada elemento do tipo abstrato que pertence ao núcleo do formato de dados, existem diversos elementos diferentes que, dentro do mesmo grupo de substituição, o podem substituir.

Uso de meta-informação O GML prevê o uso de meta-informação para atribuir um caráter semântico aos seus dados. Isto é conseguido com a inclusão, na linguagem, de um tipo abstrato de dados *AbstractMetadataPropertyType*. Todos os tipos de dados definidos, que derivem deste tipo, têm a possibilidade de incluir meta-informação nos dados. Para além deste mecanismo explícito, pode-se dizer que a forma como a informação é codificada permite uma inferência semântica dos próprios dados, visto que, as entidades e respectivos atributos são descritos de acordo com o modelo atributo/valor, que é uma cópia do conceito do modelo classe/propriedade do *Resource Description Framework*²¹ (RDF).

Extensibilidade do formato Como foi explicado no início da subsecção 4.2.4, para se codificarem os dados em formato GML, o utilizador tem de criar um esquema de aplicação em *XML Schema* para definir o tipo de dados que quer ver representados, recorrendo à derivação de tipos de dados já existentes. Estando do lado do utilizador a definição dos seus próprios formatos de dados, este mecanismo permite uma extensibilidade enorme. O formato permite ainda a utilização dos grupos de substituição que permitem, por sua vez, uma nova extensibilidade ao nível do esquema de aplicação definido. A extensibilidade do formato também se faz através da herança entre os elementos.

Herança nos dados O formato de dados GML utiliza o mecanismo de herança entre os seus dados. No formato GML alguns tipos de elementos são derivados de outros tipos os quais são apelidados de pai destes. Um elemento derivado herda todos os elementos dos seu pai e ainda pode adicionar os seus próprios elementos.

Definição de dados pelo utilizador A definição de dados pelo utilizador no formato GML, como foi explicado no paragrafo “Extensibilidade do formato”, é realizada no momento que este procede a definição do esquema de aplicação para os seus dados. Recorrendo ao *XML Schema*, o utilizador define os seus novos tipos de dados que derivam

²¹<http://www.w3.org/RDF/>

sempre, em última instância, de um tipo de dados definido num esquema normalizado do GML.

Utilização de recursos externos No GML a utilização de recursos externos é permitida recorrendo aos componentes existentes no *XLink*. O elemento *gml:AssociationAttributeGroup* é o tipo de dados definido que dá suporte ao uso do *XLinks*²² como método para indicar o valor da propriedade referenciada no GML. O valor dessa propriedade é retornado pelo recurso indicado no atributo *xlink:href*.

Controlo sobre a visualização Não existe no GML nenhum mecanismo de controlo sobre a visualização visto que o formato não se preocupa com a visualização dos seus dados, deixando isso para outros formatos, nomeadamente o SVG, da W3C.

Capacidade de 3-D O formato de dados GML, tem capacidade de 3-D na representação de objectos. Esta capacidade é introduzida pelo elemento *Solid*, que é a base da tridimensionalidade no GML.

Gestão dos recursos Não efectua qualquer gestão de recursos uma vez não efectua a visualização dos mesmos.

Expressividade no domínio dos Sistemas de Comando e Controlo É um objectivo da OGC que o formato de dados GML, se torne no *standard* mais completo na representação da realidade geográfica. Com esse intuito tem desenvolvido uma serie de *XML Schemas* com centenas de tipos de dados diferentes. Além disso, o GML permite que um “Esquema de Aplicação” possa derivar de outros “Esquemas de Aplicação” e assim conseguir uma maior especialização do objecto a representar. O GML apresenta-se como um formato com grande potencial de expressividade no domínio dos sistemas de C2, contudo os esquemas definidos pela OGC não contemplam nenhuma expressividade extra neste domínio.

4.3 Comparação qualitativa dos formatos

4.3.1 Introdução

Nesta secção proceder-se-á a uma análise critica acerca dos formatos em cada uma das caraterísticas listadas para efetuar a comparação qualitativa. Onde serão apontados os formatos que possuem a caraterísticas em causa e quais as vantagens que apresenta comparativamente aos que não a possuem. Nos casos em que todos os formatos apresentam a caraterística, então sublinhar-se-á o(s) formato(s) que se destaca(m) pela positiva ou pela negativa nessa mesma caraterística. Existem ainda alguma caraterísticas nas quais

²²XML Linking Language da W3C, <http://www.w3.org/TR/xlink/>

se apresentam uma taxonomia para distanciar os vários formatos dentro da mesma característica.

4.3.2 Análise das características

- Uso de *XML Schema* para definição do documento
 - Os formatos, que utilizam *XML Schema* são o NVG, KML e GML. Vantagens: a) utilização uma especificação para a definição do documentos *standard* da W3C; b) uma maior expressividade na representação dos dados; c) possibilidade de recorrer a qualquer dos muitos mecanismo implementados pela linguagem, tais como, restrições, grupos de substituição, a herança e derivação de tipos de dados; d) trabalhar e validar vários *namespaces* no mesmo documento; e) ganho na flexibilidade e extensibilidade do formato ao poder definir tipos de dados que derivam de outros tipos de dados. Com a utilização de grupos de substituição e herança verifica-se um aumento da facilidade de manutenção do código. Desvantagens: A utilização dos vários mecanismos avançados disponíveis na linguagem, podem tornar a mesma demasiado complicada.
 - O formato SVG utiliza o *Document type Definitions* para definir os seus documentos. Vantagens: menor complexidade na especificação. Desvantagens: menor expressividade e não utiliza a linguagem XML na definição do modelo de dados.
- Sistema de Coordenadas de Referência optativo
 - Os formatos, que permitem trabalhar com diferentes SCR são o GML e o SVG. Vantagens: uma maior versatilidade na representação de informação proveniente de qualquer SCR. Desvantagens: acréscimo de algumas linhas de código no formato SVG, dado que, o SCR tem de ser definido dentro do próprio ficheiro; no GML, sempre se utilizam coordenadas é necessário indicar qual é o SCR a usar, pois não existe um SCR por defeito.
 - Os formatos KML e NVG, não permitem o uso de diferentes SCR, estão obrigados a usar apenas o SCR WGS84.
- Utilização da terceira dimensão na localização de um ponto
 - Os formatos que permitem a utilização da terceira dimensão são: o GML e o KML. Vantagens: maior precisão na localização de entidades ou eventos a representar; permite explorar a terceira dimensão, o que aumenta o domínio de valores disponível para definir coordenadas; maior expressividade na localização de entidades ou eventos. Por exemplo, é possível localizar, sem ambiguidade, duas entidades que tenham as mesmas coordenadas (x,y) mas em altitudes diferentes. Desvantagens: maior verbosidade do código.

- Os formatos NVG e SVG não permitem o uso da terceira dimensão.
- Suportar a utilização do Sistema Internacional de Unidades
 - Apenas os formatos: GML e o SVG dão suporte à utilização do SI. O formato GML, permite ao utilizador optar por um tipo específico “BaseUnit” (conjunto de unidades de medida predefinido) no qual todas as unidades de medida pertencem ao SI. No entanto o utilizador pode, se assim o pretender, escolher unidades de medida fora do SI; O SVG permite ao utilizador, para cada dimensão, escolher a sua unidade de medida sendo que uma das opções é o SI. Vantagens: os dados são mais compatíveis; os dados são de mais fácil entendimento, não necessitando de ser convertidos. Desvantagens: existem zonas do globo onde não impera a utilização do SI, como é o caso dos Estado Unidos da América onde vigora um sistema de Unidades nativo, em que o uso do SI pode dificultar o entendimento.
 - Os formatos KML e NVG obrigam à utilização de unidades fora SI.
- Definição de Estilos
 - Os formatos que possuem um mecanismo próprio para atribuir estilos aos dados são: o KML, o NVG e o SVG. Dentro deste grupo podemos ainda dividi-lo em dois subgrupos:
 - * os que podem definir um estilo e partilha-lo por diversos elementos, KML e o SVG;
 - Vantagens de partilhar estilos: diminui-se a verbosidade do formato, pois um dado estilo só precisa se ser especificado uma vez por documento;
 - Desvantagens: aumento da complexidade do código.
 - * os que não podem partilhar estilos entre elementos, o NVG.
 - Vantagens: Apresentam-se ao utilizador como um formato de pacote completo (com capacidade de definição dos objectos, definição do estilo da sua visualização e da sua representação), não obrigando o utilizador a recorrer a um outro formato de dados para realizar a visualização dos dados; consome recursos na operação de “transformação” dos dados para um formato passível de ser visualizado; não é necessária uma tecnologia adicional para realizar essa operação. Por exemplo, o GML, para que os dados possam ser visualizados, usa tecnologia XSLT²³ da W3C para efectuar a transformação dos seus dados para o formato SVG. Desvantagens: maior complexidade do código; mistura a representação de dados, com a forma de como os dados vão ser visualizados.

²³*eXtensible Stylesheet Language Transformations*

- Os formatos que permitem separar a visualização dos dados a representar são: o GML e o SVG. O GML não possui um mecanismo próprio para atribuir estilos aos dados. E o SVG permite a utilização de classes de estilos CSS definidas em ficheiros externos. O SVG tem a vantagem de poder definir um ficheiro com um estilo e aplicá-lo a vários ficheiros de dados.
 - O formato que não possui um mecanismo próprio para atribuir estilos é o GML.
- Uso de Simbologia Militar
 - Apenas o formato NVG tem mecanismos específicos para tratar a Simbologia Militar. Vantagens: o formato tira partido mais facilmente da expressividade da simbologia militar para representação do contexto militar; facilidade no uso da simbologia militar, com recurso a um servidor de Simbologia Militar²⁴; código mais simples e menos verboso; o formato fica imune às alterações que ocorram ao nível da simbologia. Desvantagens: o formato fica dependente do funcionamento de um servidor de Simbologia Militar.
 - Os formatos SVG, KML e GML, não possuem qualquer mecanismo específico para o uso de Simbologia Militar. No entanto, todos eles têm mecanismos que lhes permitem recorrer aos símbolos militares sob forma de imagem, se estas imagens estiverem disponíveis como um recurso. Vantagens: Independência do formato em relação a servidor de Simbologia Militar. Desvantagens: Produção de ficheiro de tamanho maior, devido à necessidade de utilização de primitivas geométricas para representar outros símbolos que não símbolos *point-based*, como é o caso da maior parte das Medidas de Coordenação que são símbolos *multi-point based*. O sistema que utiliza o formato fica dependente dos símbolos por ele definidos, o que representa um grande esforço de manutenção do próprio sistema, visto que, caso um símbolo sejam criado ou alterado tem de se mexer no código da aplicação que utiliza o formato.

- Elementos do tipo geométrico

No que se refere a capacidade geométrica de cada um dos formatos, conclui-se que:

- Quantitativamente A ordenação dos formatos por ordem decrescente de primitivas geométricas é a seguinte: GML com 25, NVG com 10, SVG com 7, KML com 6.
- Qualitativamente, todos permitem uma grande expressividade geométrica, sendo que, o formato NVG se destaca pela existência de primitivas geométricas vocacionadas para o contexto militar, sendo as primitivas *arrow* e *corridor*

²⁴Os servidores de Simbologia Militar permitem através da utilização de um identificador de símbolo e da utilização de atributos configurar um dado símbolo padrão, num símbolo específico que representa a nossa realidade

Formato de dados	Nº de primitivas geométricas
NVG	10
SVG	7
KML	6
GML	25

Tabela 4.5: Número de primitivas geométricas por formato.

exemplos disso. O formato GML, destaca-se dos restantes formatos pela sua enorme variedade de primitivas geométrica, que lhe conferem uma expressividade superior aos restantes. O KML, destaca-se pela negativa, por não possuir as primitivas geométricas, *Circle* e *Ellipse* existentes nos restantes formatos, que diminuiu a sua expressividade geométrica relativamente aos restantes.

- Flexibilidade na definição dos dados

Quanto à flexibilidade na definição dos dados em cada um dos vários formatos, conclui-se que, embora sendo uma informação importante, não há diferenças significativas entre as capacidades dos vários formatos.

- Uso de meta-informação

- Todos os formatos permitem o uso de meta-informação nos seus dados. No entanto, o formato KML é o único que não permite recorrer ao uso de elementos do vocabulário definidos no *Dublin Core Metadata Element Set*²⁵ na descrição da meta-informação. Esta limitação é uma desvantagem para o KML, visto que a utilização *Dublin Core Metadata Element Set* é uma norma bastante divulgada e aceite no que diz respeito à descrição de elementos na Internet.

- Extensibilidade do formato

- Todos os formatos prevêm a possibilidade de extensão dos seus formatos de dados.
- O formato GML destaca-se dos restantes nesta característica, uma vez que como já foi referido, neste formato, é o utilizador que define os seus próprios tipos de dados, à custa da derivação dos tipos dados definidos nos esquemas normalizados do GML. Logo, a extensibilidade do formato apresenta-se como algo intrínseco ao formato GML.

- Herança nos dados

- Os formatos que permitem herança entre os dados são: KML e GML;
- Os formatos que não permitem herança entre os dados são: NVG e SVG. Vantagens: menor verbosidade do código e maior facilidade na definição dos tipos

²⁵<http://www.dublincore.org/documents/dces/>

de dados, pois não é necessário estar a reescrever novamente as propriedades detidas pelo elemento pai. Esta propriedade no formato de dados representa um ganho na facilidade de manutenção do próprio formato. Desvantagens: aumento da complexidade da especificação.

- Definição de dados pelo utilizador
 - Todos os formatos possuem mecanismos para permitir a definição de novos tipos de dados pelo utilizador do formato de dados.
 - Nesta característica o formato GML destaca-se de todos os outros, visto que a definição de dados por parte do utilizador do formato de dados é obrigatória aquando da definição do Esquema de Aplicação. O que é uma vantagem para quem sabe trabalhar com a linguagem *XML Schema* e é um problema para quem não domina esta linguagem, porque vai ter muitas dificuldades para definir o Esquema de Aplicação.

Vantagens: a definição de dados por parte do utilizador do formato, permite uma grande flexibilidade ao formato para fazer face a novas necessidades. Desvantagens: a definição de novos tipos de dados pode, se isso não for acautelado, por em causa os objetivos para que o modelo de dados que foi definido inicialmente.

- Utilização de recursos externos
 - Todos os formatos permitem a utilização de recursos externos;
 - O formato de dados NVG, ao contrário dos restantes formatos, apenas pode referenciar recursos externos que sejam imagens no formato *GIF* ou *PNG*. Embora sendo uma garantia que os recursos referenciados são apropriados para circular numa rede de computadores (por exemplo: na Internet), é no entanto, uma grande limitação na utilização de recursos externos.
 - O formato KML não suporta a importação de imagens em formato *SVG*, o que é uma limitação, no entanto, possui a capacidade de escalar imagens do tipo *Raster*.
- Controlo sobre a visualização
 - Os formatos que permitem controlo sobre a visualização dos dados são: o KML e o SVG. No entanto, o formato KML, destaca-se por utilizar mecanismos que lhe permitem definir o controlo e grau de detalhe a visualizar, o qual elimina entidades cuja visualização seria impercetível dado à escala. O formato SVG, destaca-se por permitir um controlo efetivo sobre os dados, ao nível da visualização das entidades e da animação de imagens, recorrendo à linguagem de programação *ecmascript*. Vantagens: maior controlo sobre a visualização dos dados; permite um controlo sobre seleção da informação existente

nos dados a ser apresentada. Por exemplo, no formato KML pode-se definir o detalhe da informação a ser mostrada, mediante a qualidade com que pode ser visualizada, o que potencia uma enorme melhoria na apresentação da informação codificada nos dados. Visualizar os dados sempre da mesma forma, independente das capacidades dos dispositivos onde estes são visualizados ou do interesse do utilizador final seria uma desvantagem. Desvantagens: maior complexidade do código; maior verbosidade. No caso do SVG é necessário conhecer a linguagem *ecmascript* para poder realizar o controlo da visualização.

- Os formatos: NVG e GML; não permitem nenhum controlo sobre a visualização dos dados.

- Capacidade de 3D

- Os formatos com capacidade 3D são: o KML e o GML. Vantagens: maior capacidade de representação de entidades do mundo real; maior detalhe na representação de entidades. Por exemplo, na representação do terreno é possível ter uma perceção mais verdadeira da realidade, nomeadamente em relação às elevações. Desvantagens: maior complexidade do código; maior verbosidade;
- Os formatos: NVG e SVG; não possuem capacidade de 3D nos seus formatos.

- Gestão dos recursos

- Apenas um formato permite a gestão de recursos, o KML. Vantagens: ganhos em performance na leitura dos dados, pois, neste formato os recursos externos apenas são carregados aquando da sua efetiva visualização. Não permitindo o carregamento de recursos externos (que podem representar um grande consumo de recursos da aplicação: banda-larga, cpu, memória, etc) sem que estes depois nunca venham a ser realmente visualizados. Desvantagens: maior complexidade do código.
- Os formatos que não permitem a gestão de recursos são: o NVG, o SVG e o GML.

- Expressividade no domínio dos Sistemas de Comando e Controlo

- O formato NVG apresenta-se com mecanismos específicos vocacionados para o domínio dos Sistemas de Comando e Controlo, o que lhe conferem grande expressividade nesse domínio. Por exemplo, as primitivas *corridor* e *arrow*. No entanto, tem lacunas na representação tridimensional dos, na utilização da terceira dimensão para localização e não permite a utilização de diferentes SCR.
- O GML é um formato com maior potencial para poder definir todo tipo de dados, mas no core do GML não existe nenhum esquema acerca deste domínio.

Pelo que, mesmo assim dispõe da capacidade tridimensional, da utilização da terceira dimensão para localização e das suas 25 primitivas geométricas. E ainda, permite a utilização de diferentes SCR.

- No formato KML a sua expressividade no domínio de C2 é a dada pelo elemento genérico *kml:Placemark* e as suas primitivas geométricas genéricas. Permite a representação tridimensional e a utilização da terceira dimensão para localização. No entanto, não permite a utilização de diferentes SCR.
- O formato SVG, é o menos expressivo de todos no domínio de C2, pois as suas primitivas são muito básicas e vocacionadas para o grafismo. Pode utilizar diferentes SCR. No entanto, tal como o NVG, não possui capacidade tridimensional e não utiliza a terceira dimensão para localização.

Como forma de sintetizar e resumir toda a informação relativa às características dos vários formatos apresenta-se uma matriz resumo: 4.3.2

Características	NVG	SVG	KML	GML	Imp
Uso de XML Schema para definição do documento	Sim	Não	Sim	Sim	2
Sistema de Coordenadas de Referência optativo	Não	Sim	Não	Sim	1
Utilização da terceira dimensão na localização de um ponto	Não	Não	Sim	Sim	2
Suportar a utilização do Sistema Internacional de Unidades	Não	Sim	Não	Sim	1
Definição de Estilos	Sim	Sim	Sim	Não	1
Uso de Simbologia Militar	Sim	Não	Não	Não	3
Elementos do tipo geométricos a)	10	7	6	25	2
Flexibilidade na definição dos dados	Sim	Sim	Sim	Sim	1
Uso de meta-informação	Sim	Sim	Sim	Sim	2
Extensibilidade do formato	Sim	Sim	Sim	Sim	1
Herança nos dados	Não	Não	Sim	Sim	1
Definição de dados pelo utilizador	Sim	Sim	Sim	Sim	1
Utilização de recursos externos	Sim	Sim	Sim	Sim	2
Controlo sobre a visualização	Não	Sim	Sim	Não	1
Capacidade de 3D	Não	Não	Sim	Sim	1
Gestão dos recursos	Não	Não	Sim	Não	1
Expressividade na representação de entidades geográficas b)	2	4	3	1	2

Tabela 4.6: Matriz Resumo das Características

Nesta matriz²⁶ temos na primeira coluna a lista das características a avaliar, nas colunas seguintes os vários formatos de dados e na última temos o grau de importância da característica no âmbito da troca de informação georreferenciada entre sistemas de C2

²⁶ a) Número de primitivas geométricas b) 1- mais expressivo e 4- o menos expressivo

no contexto militar (1-Importante, 2-Muito importante 3-Muitíssimo importante). Nas linhas número 8, 10 e 17 recorreremos à quantificação de valores para melhor diferenciarmos os respetivos formatos. Na linha 8, apresentamos o número de primitivas geométricas de cada formato. Na linha 17, apresentamos o valor correspondente ao pódio dos formatos que possuem maior expressividade na representação de entidades geográficas, sendo que a primeira posição foi ocupada pelo formato de dados com maior expressividade, e assim sucessivamente.

4.4 Desafios em aberto

Nesta avaliação qualitativa foi abordado um conjunto de características eleitas como as mais importantes acerca de cada formato no contexto de troca de informação entre sistemas de C2 em contexto militar, contudo, persistem ainda algumas questões que numa próxima oportunidade será interessante de analisar:

- Agora que já conhecemos o tipo de SCR que cada formato de dados pratica seria interessante analisar nos formatos que permitem mais do que SCR: qual a precisão na conversão de coordenadas entre os diversos SRC, qual a dimensão dos clientes desses formatos relativamente aos clientes que apenas utilizam um único SRC.
- Agora que possuímos o conhecimento de que todos os formatos têm capacidade de codificar meta-informação seria interessante analisar as primitivas que cada formato disponibiliza e os mecanismos que são utilizados no processo de envio da meta-informação, para se perceber se a meta-informação é toda enviada no primeiro contacto entre os sistemas (tornando o início da comunicação lento) ou tem mecanismos que lhe permite enviar meta-informação quando requerida.
- Agora ficamos a saber que todos os formatos de dados permitem o acesso a recursos externos, seria interessante saber até que ponto os formatos de dados tem capacidade de receber um URI (Uniform Resource Identifiers) e descobrir a localização do recurso de forma autónoma.
- Agora que já percebemos que mecanismos existem em cada formato de dados para realizar o controlo sobre a visualização dos próprios dados, seria interessante debater a questão de que lado deve estar a decisão do controlo da visualização dos dados, do lado do produtor do dados incluindo especificação no formato que permita isso mesmo, ou do lado cliente permitindo assim a quem visualiza o poder escolher a informação que deseja visualizar.

Os desafios aqui deixamos em aberto constituem-se como boas oportunidades para futuros trabalhos a serem desenvolvidos no contexto desta avaliação qualitativa.



Avaliação Quantitativa

5.1 Introdução

Esta avaliação quantitativa foi realizada com recurso a uma experiência *in vitro*, em resposta à segunda pergunta de investigação: **O formato de dados NVG permite melhor desempenho que os formatos de dados alternativos, na partilha de informação georreferenciada entre sistemas de C2?**, e por forma a complementar a avaliação qualitativa apresentada no capítulo 4. Os formatos de dados propostos para esta comparação quantitativa são: NVG, SVG e o KML.

Esta experiência foi realizada com recurso a uma aplicação desenvolvida para o efeito, *Scenario Simulator*, que integra um simulador de cenários militares, *Scenario Generator*, um transformador da informação contida nos cenários em ficheiros XML nos três diferentes formatos, *XML Transformer*, e um leitor e registador de métricas ao longo do processo de transferência de um ficheiro do Produtor para o Consumidor.

A informação a ser trocada consiste em cenários que evoluem ao longo do tempo. Em particular, estamos interessados na comparação dos tamanhos dos ficheiros a serem trocados, e na *performance* dos ficheiros durante todo o processo da troca dos mesmos, considerando os padrões de comunicação: *Request-Response* e o *Publish-Subscribe*.

Os valores obtidos pelos vários formatos de dados nesta avaliação quantitativa, serão sempre analisados em termos relativos e nunca de forma individual, dado que, o que se pretende é comparar o comportamento relativo de cada formato de dados em relação aos restantes. Para garantirmos resultados justos e credíveis, todos os formatos foram sujeitos aos mesmos testes e nas mesmas condições.

5.2 Planeamento da experiência

5.2.1 Objetivos

Para dar resposta à segunda questão de investigação foram levantados dois objetivos que se procuram atingir com esta experiência:

Obj1 **Análise** dos formatos de dados NVG, SVG e KML **a fim de** realizar uma avaliação quantitativa entre os diversos formatos de dados **em relação à** dimensão dos ficheiros produzidos no mapeamento dos cenários gerados **do ponto de vista** dos protocolos de comunicação *Publish-Subscribe* e *Request-Response* **no contexto** da troca de informação georreferenciada entre sistemas de C2 em cenários militares.

Obj2 **Análise** dos formatos de dados NVG, SVG e KML **a fim de** realizar uma avaliação quantitativa entre os diversos formatos de dados **em relação à** *performance* obtida durante o processo de partilha dos mesmos **do ponto de vista** dos protocolos de comunicação *Publish-Subscribe* e *Request-Response* **no contexto** da troca de informação georreferenciada entre sistemas de C2 em cenários militares.

5.2.2 Unidades Experimentais

A unidade experimental na nossa experiência é o *Cenário*, referido em 2.1. O *Cenário* é composto por elementos de dois tipos: *Força* e *Medidas de Coordenação*. Os elementos do tipo *Força* representam forças militares georreferenciadas. Cada elemento deste tipo tem três valores associados que se podem alterar ao longo da vida do *Cenário* a que pertencem: *localização*, *velocidade de movimento*, *orientação do movimento*, *estado operacional da força*. Os elementos do tipo *Medidas de Coordenação*, representam as medidas implementadas para facilitar a ação de comando e controlo sobre elementos do tipo *Força*. Os elementos do tipo *Medidas de Coordenação* são representados por um grupo de primitivas geométricas que tipicamente incluem: *Pontos*, *Círculos*, *Elipses* e *Polígonos*.

5.2.3 Ferramentas de Apoio

Para realizar esta experiência foi desenvolvida uma aplicação em *Java*, **Scenario Simulator**, a qual permite comparar a *performance* e o tamanho dos ficheiros na troca de informação georreferenciada entre sistemas de C2.

5.2.3.1 Scenario Simulator

Esta aplicação permite criar diferentes tipos de cenários através da alteração das configurações do *Cenário*. Permite também criar diferentes cenários do mesmo tipo através da introdução de valores iniciais da posição dos elementos, diferentes para cada cenário. O *Scenario Simulator* permite gerar um *Cenário*, realizar a sua evolução ao longo do tempo, guardar toda a informação produzida numa base de dados e por fim transformar

essa informação nos três formatos diferentes (NVG, SVG e KML), em dois padrões de comunicação: *Request-Response* e *Publish-Subscribe*. A arquitetura do *Scenario Simulator* é apresentada na figura 5.1.

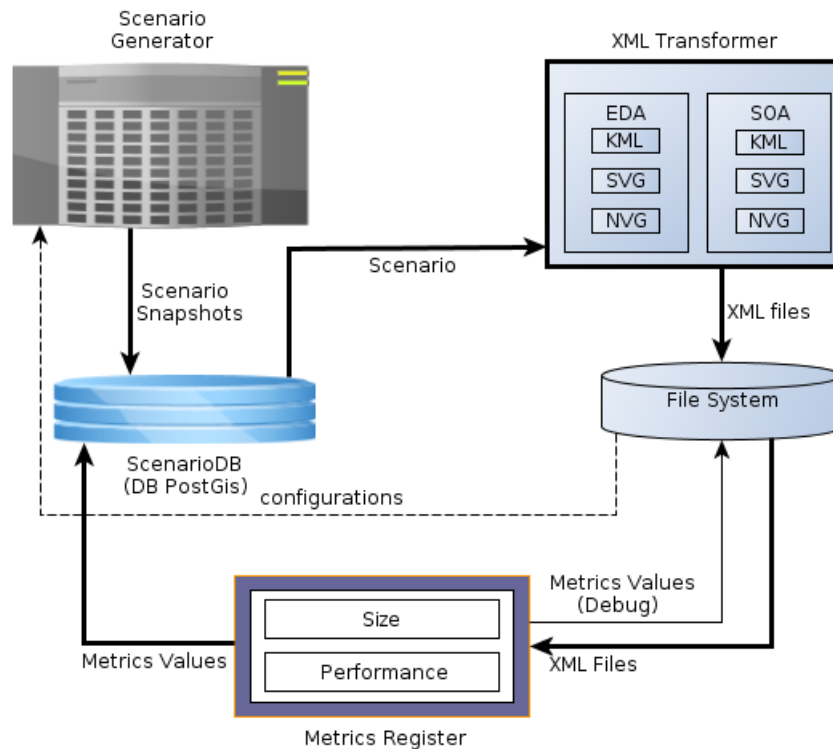


Figura 5.1: Arquitetura da aplicação em Java, *Scenario Simulator*

O *Scenario Generator* gera e anima os cenários de forma não determinística. Um cenário é criado de acordo com as configurações presentes no ficheiro de configuração existente em disco, com a especificação da constituição da Força. Mediante a especificação da Força, é o escalão em que o cenário vai ser gerado (ver tabela 5.1), que vai especificar o número de elementos a considerar na geração do cenário .

Tabela 5.1: Escalões permitidos pelo *Scenario Generator*.

Brigada/Brigada Independente
Batalhão/Grupo
Companhia/Esquadrão
Pelotão
Secção
Esquadra/Equipa

Após criar um cenário inicial, um novo *snapshot* (representa o estado de todos os elementos que constituem o cenário num dado momento) é gerado periodicamente. Essencialmente, a animação consiste na atualização das características dos elementos pertencentes ao cenário, recorrendo a um intervalo de probabilidade para uma alteração ocorrer, parametrizável pelo administrador do *Scenario Generator*. Existem quatro parâmetros deste tipo:

P1 - probabilidade de alteração da direção de movimento dos Elementos do tipo Força;

P2 - probabilidade de diminuição da capacidade operacional dos Elementos do tipo Força;

P3 - probabilidade de deslocamento dos Elementos do tipo Força para uma nova posição;

P4 - probabilidade de alteração dos Elementos do tipo Medidas de Coordenação;

O parâmetro P1 permite influenciar as alterações à direção do movimento a seguir pelos elementos do tipo *Força*. Desta forma pode-se simular operações mais ou menos complexas ao nível do movimento das forças no terreno.

O parâmetro P2, permite influenciar a velocidade com que os elementos do tipo *Força* perdem capacidade operacional para combater até ficarem inoperacionais. Desta forma pode-se simular a variação de intensidade dos combates nas operações.

O parâmetro P3, permite influenciar a alteração de posição dos elementos do tipo *Força* durante toda a operação. Desta forma pode-se simular a forma como as forças no terreno se posicionam para fazer face aos condicionamentos táticos de cada operação.

O parâmetro P4, permite influenciar a alteração dos elementos do tipo *Medidas de Coordenação* durante toda a operação. Desta forma pode-se simular as diversas alterações que existem ao nível tático na coordenação das forças no terreno.

A soma dos valores atribuídos a P1, P2 e P3 em comparação com o valor atribuído a P4, permite transpor para o cenário o dado empírico, no qual as medidas de coordenação têm uma taxa de variação muito mais reduzida que a da força. A título de exemplo, uma medida de coordenação pode ser mantida sem sofrer alterações por período de horas ou mesmo dias, por oposição de uma força em deslocamento está em constante mudança de posição.

Para cada valor que pode ser alterado numa atualização do cenário, é gerado um número pseudo-aleatório no intervalo de [0-100]. Se esse valor se encontrar acima de um valor pré-fixado então essa alteração ocorre segundo as regras pré-estabelecidas.

Cada *snapshot* é guardado numa base de dados em *PostgreSQL*¹ com as extensões para *PostGis*².

¹<http://www.postgresql.org/>

²<http://postgis.refrains.net/>

O *ScenarioDB*, é uma base de dados *PostgreSQL* com extensão para dados geográficos, *PostGis*, que serve de repositório permanente dos cenários criados pelo *Scenario Generator*, a fim de, permitir que, uma vez um cenário gerado, seja possível consultar toda a evolução desse cenário, desde o seu estado inicial até ao estado final da simulação. A salvaguarda da evolução de um cenário é essencial para a sua posterior consulta, uma vez que a evolução dos cenários é não determinista, o que na prática impediria o utilizador de gerar dois cenários iguais.

Esta base de dados é também um repositório do valor das métricas calculadas para cada cenário gerado. Assim sendo, o elemento *ScenarioDB*, constitui-se como um repositório dos dados e dos resultados da experiência de fácil consulta e rápido acesso.

O **XML Transformer** é o responsável pela transformação da informação guardada na base de dados referente a cada cenário, para ficheiros XML, nos três diferentes formatos: NVG, SVG e KML. O *XML Transformer*, para cada formato de dados gera dois tipos de documentos: documentos XML para serem usados no padrão de comunicação *Request-Response* e os documentos XML para serem utilizados no padrão de comunicação *Publish-Subscribe*. A modelação do *XML Transformer*, foi alvo de especial esforço para garantir uma grande flexibilidade ao seu código quer na introdução de novos formatos de dados, quer a introdução de novos padrões de comunicação, para facilitar futuras réplicas da experiências com novo formatos ou padrões de comunicação. Necessitando de um esforço de integração muito simples graças a esta flexibilidade extra.

O **File System** serve para guardar os ficheiros XML gerados pelo *XML Transformer*, para guardar os ficheiros referentes às métricas (para detecção de erros) e para guardar os ficheiros de configuração do *Scenario Simulator*.

O **Metrics Register** é o elemento responsável pela realização das métricas aplicadas aos ficheiros produzidos pelo *XML Transformer* nos diversos formatos de dados. O *Metrics Register* lê os ficheiros guardados no *File System*, simula as várias operações necessárias ao envio e à receção de cada ficheiro do Produtor para o Consumidor e calcula as métricas de tamanho e de *performance* referentes a cada um dos formatos de dados, registando esses valores na base dados *ScenarioDB* e no *File System* (para efeitos de depuração de erros).

5.2.3.2 Modelo de Dados da Base de Dados

O modelo de dados desenhado para servir de base de dados ao *Scenario Simulator* é apresentado na figura 5.2. Como se pode observar pelo modelo relacional apresentado na figura 5.2, são guardados vários tipos de dados na base de dados *ScenarioDB*:

- São guardadas as configurações de cada cenário (na tabela *scenario_config*), para futura consulta e fácil identificação dos cenários;
- São guardados todos os elementos (do tipo Força e do tipo Medidas de Coordenação) criados por cada cenário (na tabela *element*). O elementos guardam em si todas as características que são imutáveis durante as atualizações;

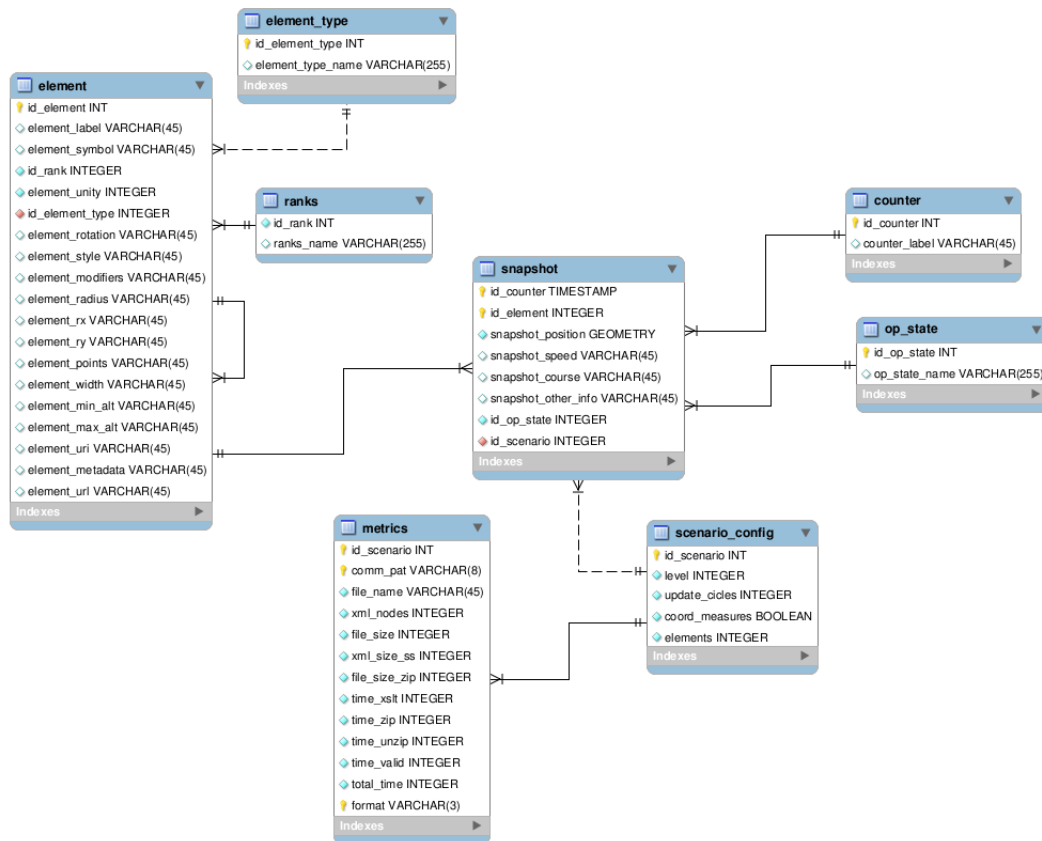


Figura 5.2: Modelo Relacional da Base de Dados ScenarioDB.

- São guardados ainda todos os diferentes estados dos elementos que compõem cada cenário, desde a sua criação até ao final das suas atualizações (na tabela *snapshot*). Cada estado é definido por uma estampagem temporal e um identificador do cenário a que diz respeito. Um estado de um cenário é definido pelo conjunto de todos os elementos que pertencem a esse cenário e que têm a mesma estampagem temporal;
- E por fim, são guardadas as métricas calculadas para cada situação testada pela aplicação *Scenário Simulator* (na tabela *metrics*). Estas métricas guardam todos os valores obtidos pelo elemento *Metrics Register* por forma a permitir uma consulta fácil dos resultados das métricas com os valores instanciados das diversas variáveis que serviram de valores iniciais para a respetiva situação a ser testada.

5.2.3.3 Detalhes de implementação

Durante a fase de implementação do elemento *XML Transformer* do *Scenario Simulator* algumas decisões de implementação tiveram de ser tomadas relativamente ao mapeamento da informação contida num dado cenário nos três formatos de dados. Dado que,

os formatos SVG e KML não possuem as mesmas primitivas para realizar todas as funções, previstas no formato de dados a contrastar, foi necessário assumir certos pressupostos e utilizar certos elementos para fins que não estavam previstos.

Foi também uma decisão de implementação, testar os diversos formatos de dados ao nível da utilização de meta-dados. Por isso todos os ficheiros produzidos contêm, para além da informação presente no cenário a mapear, meta-informação que identifica a entidade que produziu a informação e um título para identificar o conteúdo do ficheiro, igual em todos os formatos.

Pressupostos assumidos O formato de dados NVG, recorre a um Servidor de Símbolos para apresentar a imagem correspondente a cada símbolo. Na sua definição o formato NVG permite, através de atribuição de valores a atributos próprio, alterar um símbolo genérico para ser mais específico e menos ambíguo na informação que transmite. Os formatos SVG e KML, como já ficou bem explícito no capítulo 4, não foram desenhados para trabalhar com nenhum Servidor de Símbolos. No entanto, tanto o SVG como o KML permitem incorporar imagens como recursos externos ao ficheiro, através do atributo *xlink:href* no SVG, e do elemento *href* no KML, com recurso a URI. Logo, foi assumido que esse URI poderia designar um determinado símbolo num Servidor de Símbolos criado para o efeito.

Elementos utilizados para fins não previstos O formato de dados KML não tem as primitivas, *Circle* e *Ellipse*, existentes nos formatos de dados, NVG e SVG. Logo, sempre que foi necessário proceder ao mapeamento de circunferências ou elipses no formato KML, recorreremos ao uso da primitiva *Polygon*, para conseguirmos codificar a mesma informação, que mais não é que uma sequência de ponto interligados por um segmento de reta.

Tecnologia utilizada As tecnologias usadas para a realização das várias operações durante o processo de envio do ficheiro, do Produtor para o Consumidor estão listadas na primeira coluna da tabela 5.2, os pacotes respetivos da biblioteca Java usados, encontram-se na segunda coluna da mesma tabela.

Tabela 5.2: Tecnologia e pacotes Java utilizados.

Tecnologia	Pacote Java usado
Processador XSLT	javax.xml.transform.Transformer;
Compressão ZIP	java.util.zip
Validação XML Schema e DTD	javax.xml.validation e javax.xml.parsers

5.2.4 Tarefas a realizar no processo de partilha ficheiros

A partilha de um Cenário num dado momento (designado por *snapshot*) é realizada através da partilha de um ficheiro XML. A partilha desse ficheiro do Produtor (elemento responsável por gerar o ficheiro) para o Consumidor (elemento que recebe e processa a o ficheiro para obter a informação), é um processo que implica a realização várias tarefas. A figura 5.3, apresenta todo o processo de partilha de um ficheiro entre um Produtor e

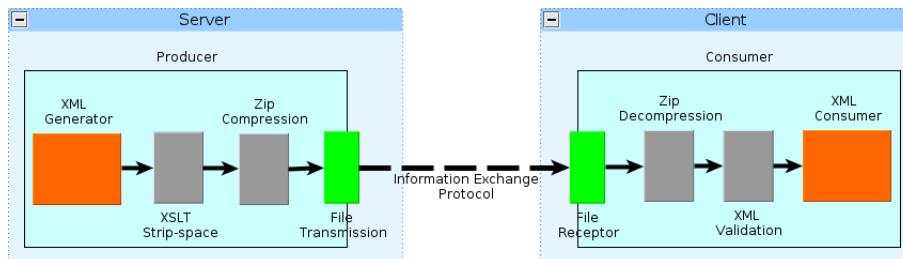


Figura 5.3: Modelo de partilha de um ficheiro XML do Produtor para o Consumidor.

um Consumidor. No lado esquerdo da figura 5.3 temos o Produtor e todos as tarefas realizados pelo Produtor:

1. *XML Generator* - geração do ficheiro XML;
2. *XSLT Strip-space* - aplicação da operação limpeza dos espaços em branco;
3. *Zip Compression* - a compressão do ficheiro para o formato *Zip*;
4. *File Transmission* - operação de transmissão do ficheiro.

No lado direito da figura 5.3 temos o Consumidor e tarefas realizados pelo Consumidor:

1. *File Receptor* - recebe a transmissão do ficheiro;
2. *Zip Decompression* - realiza a descompressão do ficheiro;
3. *Validation* - efetua a validação do documento XML;
4. *XML Consumer* - os dados são renderizados/visualizados.

Entre os dois blocos (Produtor e Consumidor), temos um protocolo de transmissão, que pode variar entre: *Publish-Subscribe* e o *Request-Response*. O código de cores utilizado na figura 5.3 tem a seguinte semântica:

- A laranja, operações que delimitam o início e o fim do processo de partilha mas que não estão no âmbito do nosso estudo;
- A cinza, operações que fazem parte do processo e que vão ser analisadas;

- A verde, operações que fazem parte do processo mas não vão ser analisadas pela falta interesse que representa para o estudo, o registo valores de transmissão de ficheiros numa rede em laboratório.

5.2.5 Hipóteses, parâmetros e Variáveis

Subquestões de investigação Todos os formatos de dados selecionados para esta experiência têm a capacidade de poder ser utilizados na troca de informação georreferenciada. Como o NVG é o formato dados desenvolvido pela NATO para o domínio específico da troca de informação georreferenciada entre C2, é nossa convicção que o NVG é mais eficiente (quer no tamanho de ficheiros produzidos, quer no tempo de envio desses ficheiros) que os formatos alternativos selecionados para esta comparação. Para analisar mais facilmente os objetivos **Obj1** e **Obj2** definidos em 5.2.1, decidimos formular cinco subquestões de investigação para servirem de base a esta comparação, e que se relacionam com os objetivos da seguinte forma:

Obj1: **q1** O formato de dados NVG produz ficheiros de dados mais pequenos que os formatos alternativos?

Obj2: **q2** O formato de dados NVG tem tempos de processamento de operações de transformação XSLT dos documentos XML mais pequenos que os formatos alternativos?

q3 O formato de dados NVG tem tempos de compressão dos ficheiros de dados mais pequenos que os formatos alternativos?

q4 O formato de dados NVG tem tempos de descompressão dos ficheiros de dados mais pequenos que os formatos alternativos?

q5 O formato de dados NVG tem tempos de validação dos documentos XML mais pequenos que os formatos alternativos?

Métricas aplicadas No seguimento da abordagem *Objetivo, Questões e Métricas* definida em [Sha03], apresentamos as métricas que se impõem no contexto das questões levantadas. Para respondermos à primeira subquestão, **q1**, avaliamos os formatos de dados em termos de métricas de tamanho [KSH02], sendo usadas as seguintes métricas:

- **FileSize**, número de *bytes* que o ficheiro ocupa após ter sido gerado;
- **FileSizeSS**, número de *bytes* que o ficheiro ocupa após ter sido realizada a operação de transformação XSL de *Strip-space*;
- **ZipFileSize**, número de *bytes* que o ficheiro ocupa após ter sido comprimido;
- **XMLSize**, número de *XML nodes* que o documento XML contém;

Medimos o *FileSize*, porque este representa o tamanho da informação após ter sido gerada nos seu estado natural. Para medirmos o tamanho do ficheiro após a operação de *Strip-space* utilizamos a métrica *FileSizeSS*. Dado que operação de compressão afecta de forma muito diferente cada formato de dados, usamos a métrica *ZipFileSize* para medir o tamanho final do ficheiro após ter sido comprimido. Finalmente, também medimos o número de *XML nodes* do documento XML, *XMLSize*. Esta métrica é importante porque o número de *XML nodes* tem influência nas operações a realizar sobre o documento XML, nomeadamente: transformação XSL e a validação. Com estas quatro diferentes métricas, esperamos mitigar ameaças à validade que resultaria de usar apenas uma métrica para medir o tamanho do formato de dados.

No que diz respeito às restantes (**q2**, **q3**, **q4** e **q5**), todas elas estão relacionadas com tempo gasto no processo de transferência do ficheiro, procedemos à utilização das seguintes métricas de *performance*:

1. **Strip-space Time**, o tempo, em milissegundos, que a operação de transformação XSLT *Strip-space* leva a processar o documento XML;
2. **Compression Time**[Sak09], o tempo, em milissegundos, que a operação de compressão leva a processar o ficheiro;
3. **Decompression Time**[Sak09], o tempo, em milissegundos, que a operação de descompressão leva a processar o ficheiro;
4. **Validation Time**, o tempo, milissegundos, que a operação de validação leva a processar o ficheiro;

A aplicação das métricas de *tamanho* e de *performance* foi conseguido mediante o desenvolvimento das operações *Strip-space*, *Compression*, *Decompression* e *Validation*, aplicadas ao ficheiro durante o seu processo de transferência do Produtor para o Consumidor. Vamos de seguida fazer uma breve análise sobre cada uma das operações e explicar o modo de como é realizada a aplicação das respetivas métricas:

- **Strip-space**, esta operação remove os espaços em branco existentes no documento XML. Os ficheiros XML quando são gerados, são sujeitos a regras de indentação (que permitem uma mais fácil leitura dos documentos XML, por seres humanos) que introduzem muitos "espaços em branco" nos ficheiros. Dado que, este "espaço em branco" não altera em nada a informação a transmitir e tem taxas de ocupação muito consideráveis, optamos por introduzir uma operação de transformação XSL, *Strip-space*[MK07], para a remoção dos "espaço em branco" dos respetivos documentos XML. Aplicação das métricas: o elemento *Metrics Register* carrega o ficheiro XML do *FileSystem*, carrega a folha de estilos XSL cuja a única instrução é aplicação da operação *Strip-space*, e aplicada a transformação XSL um novo ficheiro é guardado no *FileSystem* e as métricas do tamanho do novo ficheiro e do tempo gasto na operação de *Strip-space* são guardadas na base de dados *ScenarioDB* (ver figura 5.4).

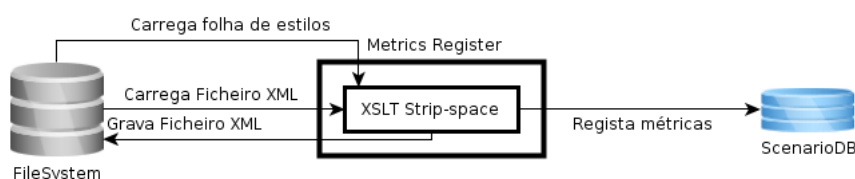


Figura 5.4: Diagrama de aplicação da transformação XSL *Strip-space*.

- Compression**, esta operação comprime o ficheiro XML. Para diminuir o tamanho do ficheiro a ser transmitido durante a troca de informação, resolvemos aplicar uma operação de compressão ao ficheiro antes de o transmitir. Existem vários compressores para ficheiros XML, que podem ser divididos em: *General Text compressor* ou *XML Conscious Compressors*[Sak09]. Utilizamos um do tipo *General Text compressor*, disponibilizado nas bibliotecas do *Java*, que utiliza o algoritmo de compressão *DEFLATE*³ e grava a informação binária no bem conhecido format de dados *ZIP*. Este algoritmo de compressão *DEFLATE* obteve excelentes resultados nos testes realizados em [Sak09]. Aplicação das métricas: o elemento *Metrics Register* carrega o ficheiro XML do *FileSystem*, e aplicada a operação de compressão *Zip* um novo ficheiro comprimido no formato *ZIP* é guardado no *FileSystem* e as métricas do tamanho do novo ficheiro e do tempo gasto na operação de *Zip Compression* são guardadas na base de dados *ScenarioDB* (ver figura 5.5).

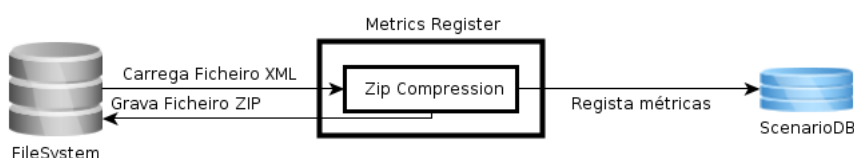


Figura 5.5: Diagrama da aplicação da operação de compressão *Zip Compression*.

- Decompression**, esta operação descomprime o ficheiro comprimido, *ZIP*, para o estado original do ficheiro. O cliente recebe o ficheiro enviado pelo servidor, em formato binário *ZIP*, e aplica a operação descompressão (inversa à compressão), para obter o ficheiro XML no seu estado original. Aplicação das métricas: o elemento *Metrics Register* carrega o ficheiro *ZIP* do *FileSystem*, e aplicada a operação de descompressão *UnZip Decompression* um novo ficheiro XML é guardado no *FileSystem* e as métricas do tamanho do novo ficheiro e do tempo gasto na operação de *UnZip* são guardadas na base de dados *ScenarioDB* (ver figura 5.6).
- Validation**, esta operação valida o documento XML recebido de acordo com o *schema*. Quando uma aplicação recebe um ficheiro XML, por norma, esta aplicação verifica se o documento XML que vem no ficheiro está de acordo com o *schema*

³<http://www.info-zip.org/doc/rfc1951.txt>

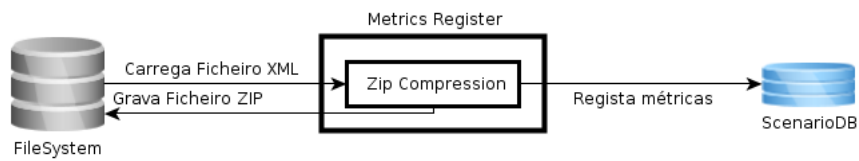


Figura 5.6: Diagrama da aplicação da operação de descompressão *UnZip Decompression*.

definido para esse documento. Esta operação permite antecipar problemas com erros no documento XML que prejudiquem o normal funcionamento da aplicação. Aplicação da métrica: o elemento *Metrics Register* carrega o ficheiro XML do *FileSystem*, e aplicada a operação *Validation* de validação do ficheiro XML carregado, depois de fazer a validação a métrica do tempo gasto na operação de *Validation* é guardada na base de dados *ScenarioDB* (ver figura 5.7).

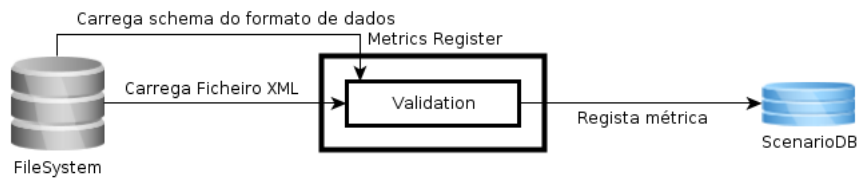


Figura 5.7: Diagrama da aplicação da operação de validação XML *Validation*.

Quadro Resumo A figura 5.8, apresenta um quadro com a aplicação das métricas de tamanho e de *performance* antes identificadas, ao longo das diversas operações realizadas na transferência de um ficheiro do Produtor para o Consumidor. No quadro observamos também qual a tecnologia em causa em cada operação.

Metrics Line							
	XML Generation	Strip-spaces	ZIP	File Transmission	UnZip	Validation	XML Consumer
Size metrics	XML Size			No measures		No measures	No measures
	File Size	File Size	File Size				
Performance metrics	No measures	Strip-spaces Time	Compressor Time	No measures	Decompressor Time	Validation Time	No measures
Technology	Java application Scenario Simulator	Java XSLTransformer	Java Zip Compressor	Theoretical approach	Java Zip Compressor	Java parsing: DOM API	F-Spot Photo Manager GoogleEarth

Figura 5.8: Quadro resumo da aplicação das métricas.

5.2.6 Desenho

Esta experiência foi planejada com a preocupação de isolar todas as variáveis existentes na configuração desta experiência. Para se perceber qual o impacto dessa variação no comportamento da simulação, em cada teste, variou-se apenas o valor de uma das variáveis mantendo todas as outras fixas. As variáveis e respectivos valores encontram-se na tabela 5.3.

Tabela 5.3: Variáveis da experiência e respectivos valores testados.

Variáveis	Valores testados
Ciclos de atualizações	1, 2, 4, 8, 16, 32, 64
Nível de detalhe	0, 1, 2, 3, 4, 5
Elementos sem Medidas de Coordenação	5, 9, 21, 54, 147, 333
Elementos com Medidas de Coordenação	14, 18, 30, 63, 156, 342
Formato de Dados	NVG, SVG, KML
Padrão de Comunicação	Publish-Subscribe, Resquest-Response

A primeira linha, *Ciclos de atualização*, representa o número de atualizações a que cada cenário foi sujeito. Inicialmente começamos com um cenário com apenas uma atualização e fomos duplicando o valor das atualizações do cenário. A segunda linha, *Nível de detalhe*, representa o *escalão da força* que foi selecionado para ser representado como elementos no cenário. Ou seja, como a dimensão da Força foi ancorada à dimensão das forças que participaram na Batalha de libertação da cidade de *Ra's al Khaffi*, a dimensão permaneceu fixa durante toda a experiência. O que variou na experiência, é o nível de detalhe com que se representam as suas unidades: nível-0 Esquadra, nível-1 Seção, nível-2 Pelotão, nível-3 Companhia, nível-4 Batalhão, nível-5 Brigada. A terceira e quarta linha representam o número de elementos representados no cenário. A terceira linha, diz respeito a cenários sem a utilização de elementos de Medidas de Coordenação. A quarta linha em oposição, diz respeito a cenários com a utilização de elementos de Medidas de Coordenação. Como se pode observar o número de elementos de Medidas de Coordenação foi fixo em nove elementos durante toda a experiência. A quinta linha, *Formato de Dados*, representa os três formatos de dados que os ficheiros foram gerados. A sexta linha, *Padrão de Comunicação*, representa os dois diferentes padrões de comunicação utilizados para a transferência dos ficheiros.

Na impossibilidade de controlar por completo todas as variáveis que podem interferir, quer na leitura dos valores das métricas (por exemplo: o tempo de espera dos processos na fila de Processos Prontos do Sistema Operativo, o tempo de espera nas tarefas de I/O ou a execução automática do serviço de Gestão de Memória da *Java Virtual Machine*⁴ "Garbage collection", entre outros) quer na geração e atualização dos cenários, visto que, os cenários são gerados de forma não determinística, optamos por repetir cada operação várias vezes para diluir possíveis erros introduzidos nesses valores.

⁴<http://www.oracle.com/technetwork/java/index.html>

Esta experiência foi repetida 3 vezes para cada configuração assumindo como valor de trabalho a média aritmética dos três valores. Para cada cenário o valor encontrado para cada métrica é o valor resultante da média aritmética do somatório dos valores de cada métrica calculada para cada ficheiro desde a geração até à última atualização sofrida pelo cenário. Todas as medições de tempos de execução das referidas operações: de *XSLT*, de Compressão, de Descompressão e de Validação; foram repetidas cinco vezes sem *cache*, sendo que o valor de trabalho foi o valor da média aritmética dessas cinco leituras.

Apresentamos de seguida alguns números considerados como interessantes, realizados nesta experiência:

Cenários gerados:

$$7 \times 6 \times 2 = 84$$

Testes efetuados:

$$7 \times 6 \times 2 \times 3 \times 2 = 504$$

Ficheiros analisados:

$$(2 \times 72 + 3 \times 72 + 5 \times 72 + 9 \times 72 + 17 \times 72 + 33 \times 72 + 65 \times 72) \times 3 = 28944$$

5.2.7 Ambiente de realização da experiência

A realização desta avaliação quantitativa foi efetuada numa máquina, cujo os recursos (*Hardware/Software*) existentes, estão detalhados na tabela 5.4.

Tabela 5.4: Detalhes de configuração dos recursos utilizados

Sistema de Operação	Ubuntu 10.04 (Kernel Linux 2.6.32-33 - generic)
CPU	2 - Intel Core 2 Duo T7500, 2.2 GHz
Hard Disk	Seagate, 7500 rpm, 250 GB
RAM	2 GB
Java Compiler	JavaSE-1.6

A configuração do cenário ao nível da estrutura da Força utilizado nesta experiência foi a que se apresenta na tabela 5.5.

A configuração do cenário ao nível das Medidas de Coordenação utilizada nesta experiência foi a que se apresenta na tabela 5.6.

A configuração dos quatro parâmetros responsáveis pela evolução pseudo-aleatória do cenário, foi que se pode observar na tabela 5.7

5.2.8 Processo

A obtenção de resultados nesta experiência é realizada através de um processo que assenta a sua execução em três fases. Cada fase é executada por um dos elementos da aplicação, desenvolvida para o efeito, *Scenario Simulator*.

Tabela 5.5: Configuração da estrutura dos elementos do Tipo Força

Tipo de Unidade	Nº Uni. por cada unidade de escalão superior
Brigada	1
Batalhão de Infantaria	2
Companhia de Infantaria	3
Pelotão de Infantaria	3
Secção de Infantaria	3
Esquadra Infantaria	2
Grupo Carros de Combate	1
Esquadrão de Carros de Combate	3
Pelotão de Carros de Combate	3
Secção de Carros de Combate	3
Esquadra de Carros de Combate	2
Grupo Artilharia de Campanha	1
Bateria de Artilharia de Campanha	3
Pelotão de Artilharia de Campanha	2
Secção de Artilharia de Campanha	2
Esquadra de Artilharia de Campanha	2
Esquadrão de Aeronaves de Asa-fixa	1
<i>Navy Task Group</i>	2
Navios Marinha mercante	1

Tabela 5.6: Configuração das Medidas de Coordenação

Tipo de Medida de Coordenação	Quantidade
Corredor Aéreo	2
Poli-linha	2
Círculo	1
Elipse	1
Polígono	1
Texto	2

Tabela 5.7: Configuração *Scenario Generator*

Parâmetro	Valor
P1	20
P2	30
P3	30
P4	70

1. A primeira fase é executada pelo elemento *Scenario Generator* e consiste na geração de cenários baseado no ficheiro de configuração existente no Disco. O ficheiro de configuração define as Medidas de Coordenação a utilizar, define a estrutura da Força e define a probabilidade de certos eventos acontecerem ou não durante a evolução dos cenários. Cada cenário é constituído por elementos do tipo Forças

- e elementos do tipo Medidas de Coordenação. Depois do cenário gerado dá-se início à evolução do cenário gerado. A evolução do cenário é um sub-processo de atualização periódica do mesmo, com o registo das respetivas atualizações (*snapshot*) na base de dados *ScenarioDB* (BD). Com o fim da evolução do cenário termina a primeira fase.
2. A segunda fase é executada pelo elemento, *XML Transformer*, e começa com a leitura do cenário (conjunto de *snapshots*) da BD, recorrendo a dois "filtros" diferentes. Estes "filtros" servem para preparar a informação lida da BD, para ser utilizada segundo o padrão de comunicação *Requeste-Responde* e o padrão de comunicação *Publish-Subscribe*. Dá-se então início a dois sub-processos paralelos de mapeamento da informação proveniente do cenário em causa. Cada sub-processo mapeia o cenário, segundo o respetivo protocolo de comunicação, nos três diferentes formatos NVG, SVG e KML, produzindo três ficheiros diferentes. A fase dois termina com a gravação no Disco (*FileSystem*) dos ficheiros produzidos.
 3. A terceira e última fase é realizada pelo elemento *Metrics Register*, e efetua simulação do processo de transferência dos ficheiros gerados na fase dois e realiza o registo das respetivas métricas. O processo começa com a leitura de cada ficheiro (f1) individualmente do Disco, e o registo na BD do respetivo valor do tamanho do mesmo. Depois aplica ao documento XML existente no ficheiro f1 a operação de *XSLT, Strip-space*, para limpar todos os espaços em branco criados pela indentação dos ficheiros XML, transformando o ficheiro f1 no ficheiro f2. Guarda na BD o valor do tamanho de f2 e o valor do tempo gasto na operação de *Strip-space*. De seguida, aplica a operação de compressão, *Zip Compression*, para reduzir ainda mais o tamanho do ficheiro f2 e obtém-se o ficheiro f3. Guarda na BD o valor do tamanho do ficheiro f3 e o valor do tempo gasto na operação de *Zip Compression*. Após a operação de compressão, é aplicada a operação de descompressão ao ficheiro f3, *UnZip Decompression*, e voltamos ao estado do ficheiro f2. Grava-se na BD o tempo gasto na operação *UnZip Decompression*. Por último, aplica-se ao ficheiro f2 a operação de validação, *XML Validation*, de acordo com o *schema* do formato respetivo. Grava-se na BD o tempo gasto na operação de *XML Validation*.

Na figura 5.9 apresenta-se o Diagrama de Atividades que de forma simplificada apresenta todo o processo necessário para a realização da avaliação quantitativa.

5.3 Resultados

Nesta secção, serão apresentados graficamente os resultados da avaliação quantitativa. No entanto, para efeitos de uma análise mais detalhada, a tabela com todos os valores poderá ser consultada no final deste documento, no anexo A.

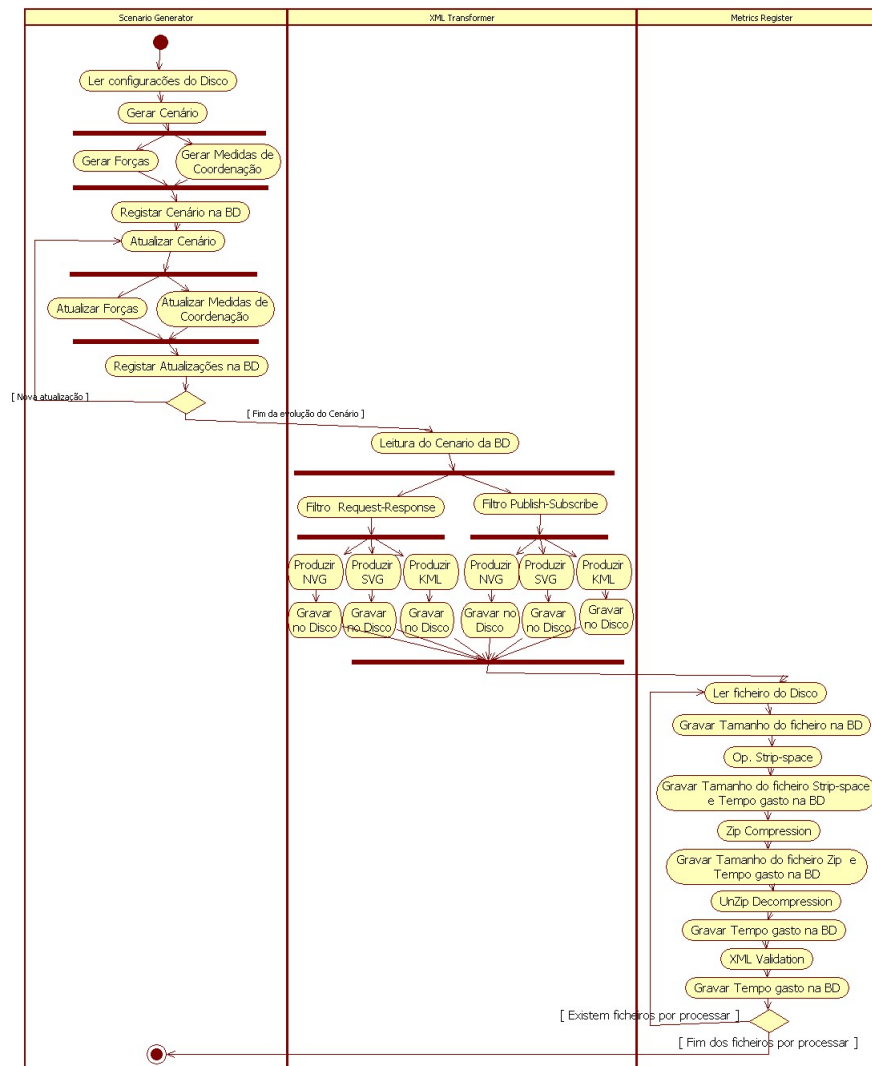


Figura 5.9: Diagrama de Atividades da avaliação quantitativa.

5.3.1 Resultados de métricas de dimensão

Como se pode visualizar, os gráficos presentes nas figuras 5.10, 5.11, 5.12 e 5.13, têm no seu eixo das ordenadas, os valores das diferentes actualizações que cada cenário pode ser sujeito (1, 2, 3, 4, 8, 16, 32, 64). Os gráficos presentes nas figuras 5.14, 5.15, 5.16 e 5.17 têm por seu lado no eixo das ordenadas, o número de elementos que podem ser utilizados para representar cada cenário. No que diz respeito ao eixo das abcissas das figuras 5.10 a 5.17, o gráfico sempre mais à esquerda, tem neste eixo de coordenadas os valores referentes ao número de nós XML existentes nos documentos XML. Nos restantes gráficos à direita deste, representam a dimensão (em *bytes*) dos ficheiros a serem transmitidos, nomeadamente: dimensão dos ficheiros XML no formato de dados original, a dimensão dos ficheiros XML sem espaços em branco e a dimensão do ficheiro comprimido no formato Zip. É de notar que em todas as situações a seguir apresentadas o valor do tamanho dos ficheiros diminui sempre, quer quando aplicada a operação de *Strip-space*, quer quando

aplicada a operação de compressão.

Evolução da dimensão dos ficheiros/documentos XML ao longo da variação do número de atualizações por cenário As figuras 5.10, 5.11, 5.12 e 5.13 sumarizam a aplicação das métricas de dimensão aos ficheiros produzidos a partir de cenários com 333 elementos (cenários sem Medidas de Coordenação) ou com 342 elementos (cenários com Medidas de coordenação). Nos gráficos apresentados na figura 5.10 observamos que ao longo dos

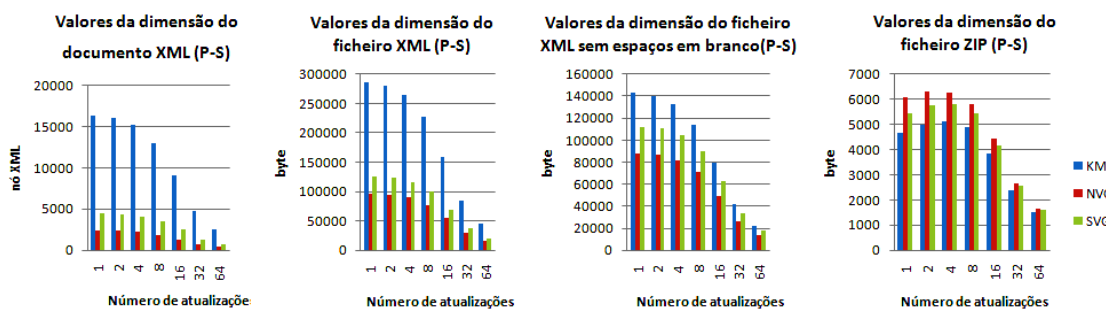


Figura 5.10: Métricas de dimensão aplicadas a cenários com 333 elementos e **sem** Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

cenários com mais atualizações o valor da dimensão dos ficheiros/documentos XML cresce, com a exceção da dimensão em formato *Zip* que tem uma fase crescente até aos cenários com 4 atualizações. O formato de dados KML sofre uma redução de tamanho que faz com que passe do formato com a maior dimensão, aquando a sua codificação, para o formato com menor dimensão no final da compressão. Observa-se que é na operação de compressão que o formato KML diminui o seu tamanho para valores inferiores ao do NVG. Paralelamente, o formato NVG percorre o caminho oposto ao do formato KML. No entanto, a diferença de tamanho entre os três formatos, no final, é muito pequena.

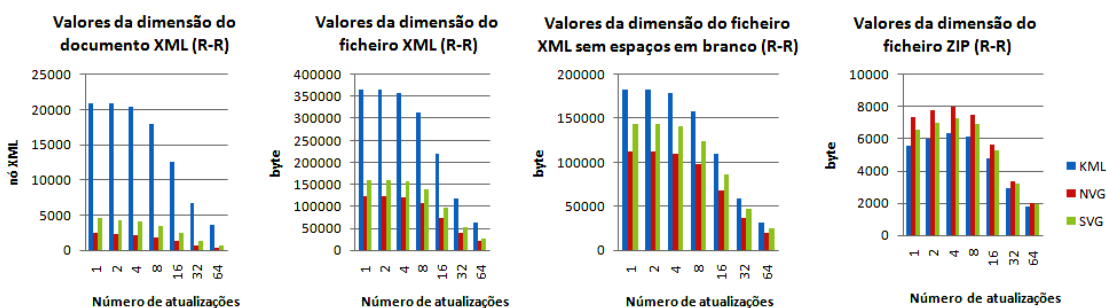


Figura 5.11: Métricas de dimensão aplicadas a cenários com 333 elementos e **sem** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Nos gráficos apresentados na figura 5.11 observamos que seguem um comportamento idêntico aos da figura 5.10, mas indicando valores ligeiramente mais elevados.

Nos gráficos apresentados na figura 5.12 observamos que o valor da dimensão dos

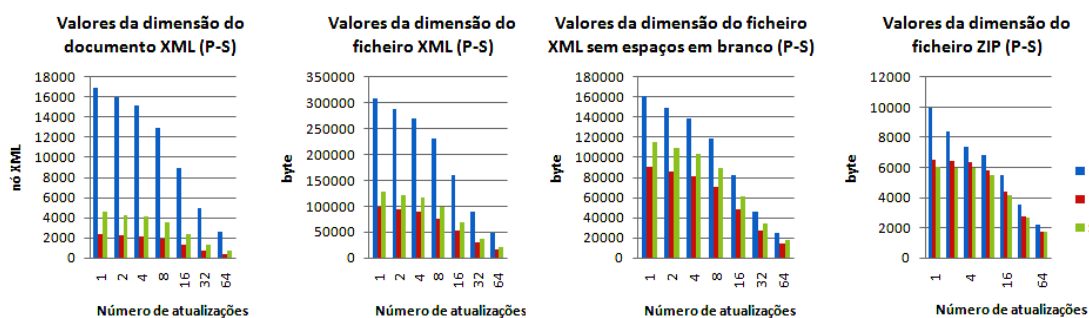


Figura 5.12: Métricas de dimensão aplicadas a cenários com 342 elementos e **com** Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

ficheiros/documentos XML decresce ao longo do aumento do número de atualizações. O formato de dados KML sofre uma notória redução de tamanho que o aproxima dos restantes dois formatos. O SVG ultrapassa o NVG na fase da compressão, ficando em primeiro lugar como o formato de dados com menor dimensão, embora a diferença de tamanho entre os dois formatos, também aqui, seja muito pequena e tende a diminuir ao longo do aumento do número de atualizações.

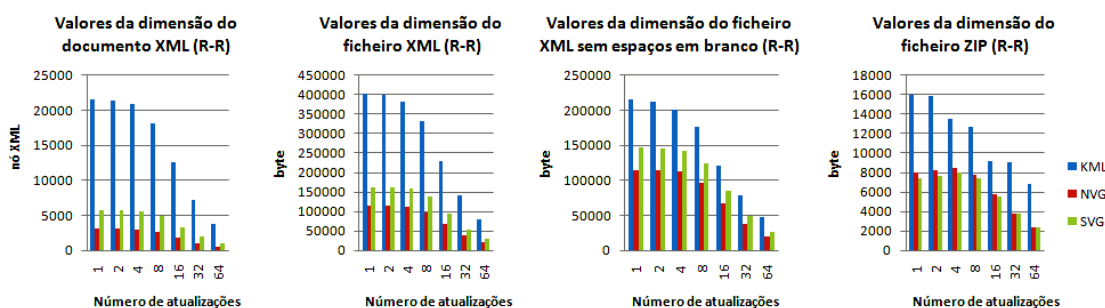


Figura 5.13: Métricas de dimensão aplicadas a cenários com 342 elementos e **com** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Nos gráficos apresentados na figura 5.13 observamos que o valor da dimensão dos ficheiros/documentos XML decresce ao longo do aumento do número de atualizações. O formato de dados KML sofre uma pequena redução de tamanho que o deixa distante dos restantes dois formatos. Também aqui, o SVG ultrapassa o NVG na fase da compressão, ficando em primeiro lugar como o formato de dados com menor dimensão, embora a diferença de tamanho entre os dois formatos, no final, seja muito pequena.

Observa-se ainda, ao longo das figuras 5.10, 5.11, 5.12 e 5.13 que o número de nós XML diminui à medida que aumenta o número de atualizações por cenário. Verifica-se também que o NVG é o formato com menor número de nós seguido de perto pelo SVG e por último, com uma distância considerável, o KML.

Evolução da dimensão dos ficheiros/documentos XML ao longo da variação do número de elementos por cenário As figuras 5.14, 5.15, 5.16 e 5.17 sumarizam a aplicação das

métricas XML de dimensão aos ficheiros produzidos a partir de cenários que foram sujeitos a 64 actualizações após a sua geração inicial.

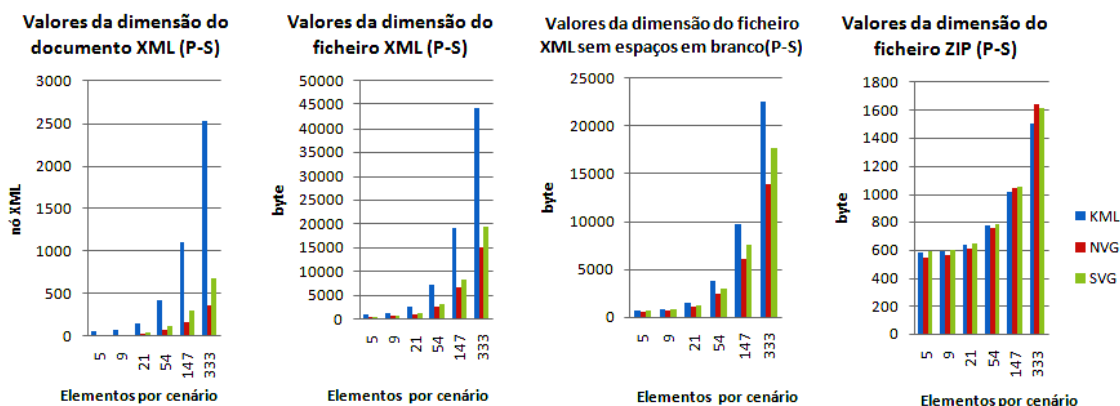


Figura 5.14: Métricas de dimensão aplicadas a cenários com 64 atualizações e **sem** Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

Nos gráficos apresentados na figura 5.14 observamos que o valor da dimensão dos ficheiros/documentos XML cresce ao longo do aumento do número de elementos por cenário. O formato de dados KML sofre uma excepcional redução de tamanho que faz com que passe do formato com a maior dimensão, aquando a sua codificação, se aproxime dos outros dois formatos. Observa-se também que o KML após a operação de compressão, demonstra uma tendência de diminuição relativa de tamanho ao longo do aumento do número de elementos por cenário, que lhe permite ultrapassar o SVG e NVG, obtendo os melhores resultados para os cenários com 147 e 333 elementos. Paralelamente, o formato NVG percorre o caminho oposto ao do formato KML. No entanto, a diferença de tamanho entre os três formatos, no final, é muito pequena.

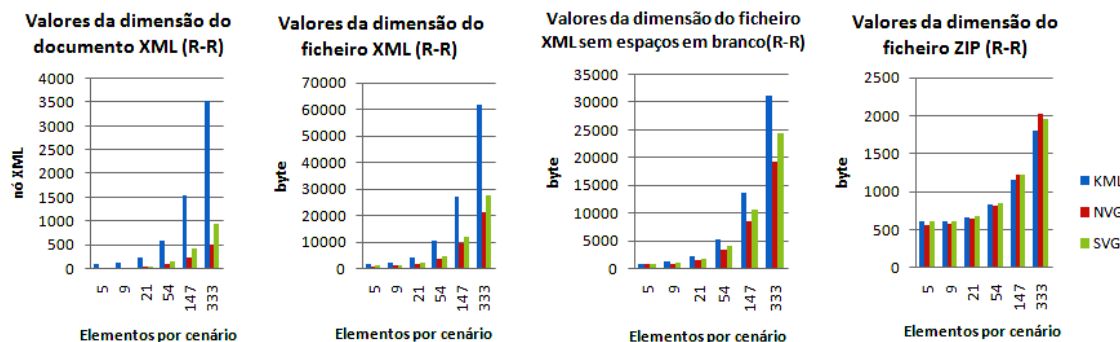


Figura 5.15: Métricas de dimensão aplicadas a cenários com 64 atualizações e **sem** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Nos gráficos apresentados na figura 5.15 observamos que seguem um comportamento idêntico aos da figura 5.14, apenas indicando valores ligeiramente mais elevados.

Nos gráficos apresentados na figura 5.16 observamos que o valor da dimensão dos

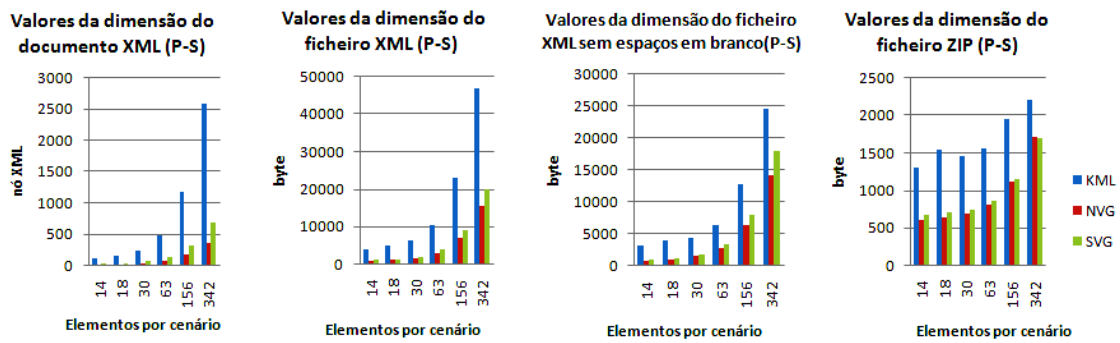


Figura 5.16: Métricas de dimensão aplicadas a cenários com 64 atualizações e **com** Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

ficheiros/ documentos XML cresce ao longo do aumento do número de elementos por cenário. O formato de dados KML sofre uma grande redução de tamanho que faz com que reduza a enorme distancia inicial para os outros dois formatos. Também aqui, é notória uma tendência de aproximação do KML aos outros dois formatos à medida que a dimensão do cenário aumenta. O NVG mantém ao longo de todas as operações os menores valores de tamanho de ficheiros gerados, com exceção para o valor após a compressão de cenários com a dimensão de 342 elementos. O SVG, mantém a sua relação de tamanho com o formato NVG ao longo do segundo e terceiro gráfico, a contar da esquerda para a direita respetivamente, da figura 5.16, mas no quarto gráfico (o mais à direita da figura 5.16) revela uma tendência de diminuição dessa relação, apresentando mesmo, para os cenários com 342 elementos, valores de tamanho ligeiramente mais pequenos que o NVG.

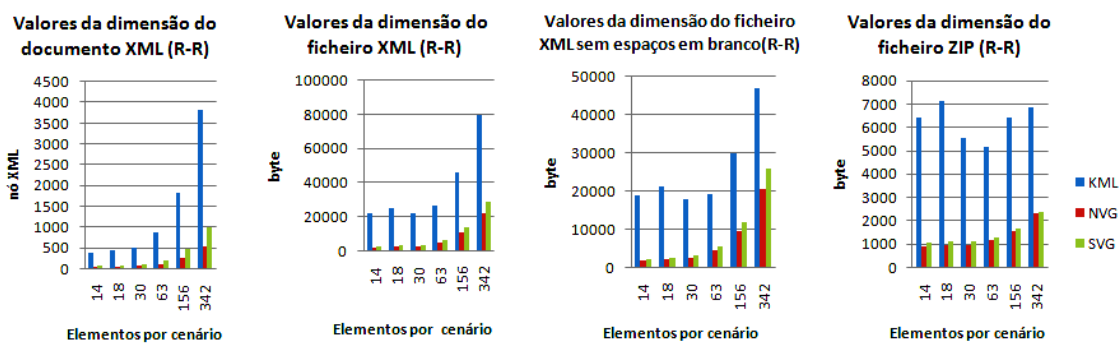


Figura 5.17: Métricas de dimensão aplicadas a cenários com 64 atualizações e **com** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Nos gráficos apresentados na figura 5.17 observamos que o valor da dimensão dos ficheiros/ documentos XML, nos formatos NVG e SVG mantêm um crescimento positivo ao longo do aumento do número de elementos por cenário, enquanto que, o formato KML apresenta ligeira irregularidade no tamanho dos ficheiros iniciais, uma mais acentuada irregularidade no terceiro gráfico e por fim uma completa irregularidade no quarto e último gráfico. O formato de dados KML sofre uma redução de tamanho que faz com

que reduza um pouco a enorme distancia inicial, para os restantes. O NVG mantém ao longo de todas as operações os menores valores de tamanho de ficheiros gerados. O SVG, mantém a sua relação de tamanho com o formato NVG ao longo do segundo e terceiro gráfico, mas no quarto revela uma tendência de diminuição dessa diferença, com o NVG.

Observa-se ainda, um facto constante ao longo das figuras 5.14, 5.15, 5.16 e 5.17 que o número de nós XML aumenta à medida que aumenta o número de elementos por cenário. Verifica-se também que o NVG é o formato com menor número de nós seguido de perto pelo SVG e por último, com uma distância considerável o KML.

Evolução da dimensão dos ficheiros/documentos XML ao longo das 64 atualizações por cada formato de dados As figuras 5.18, 5.19, 5.20 e 5.21 sumarizam a aplicação das métricas de dimensão aos ficheiros produzidos ao longo das 64 atualizações a partir de um cenário inicial. Cada figura contém uma matriz de 3×2 de gráficos.

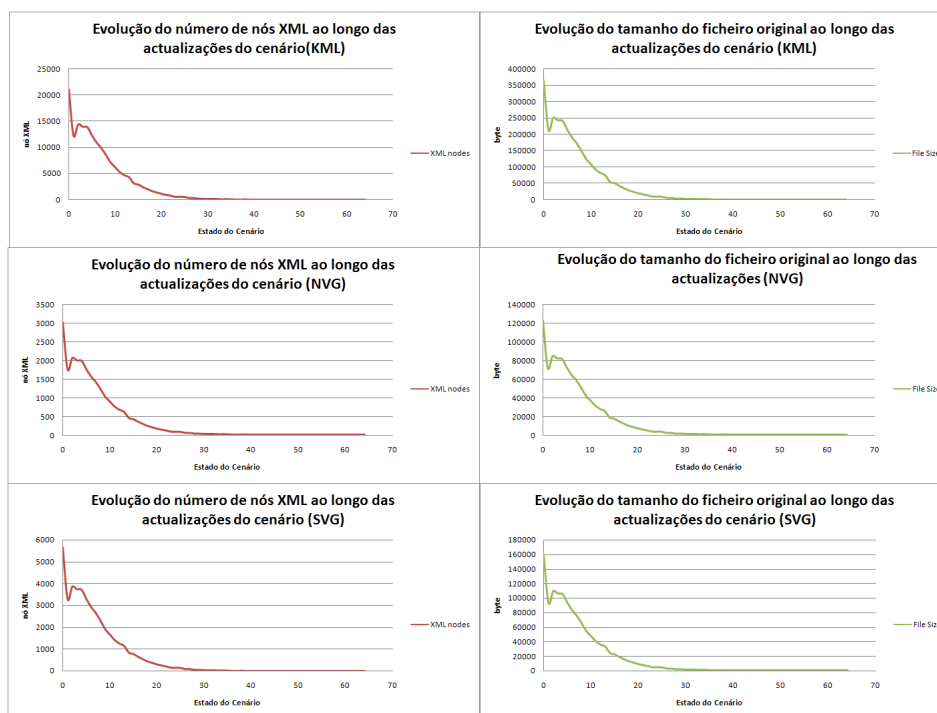


Figura 5.18: Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados (KML em cima, NVG ao centro, SVG em baixo), **sem** Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

Cada linha representa um formato de dados, KML na primeira linha, NVG na segunda linha e o SVG na terceira e última linha. A coluna da Esquerda representa a evolução ao nível dos nós existentes nos documentos XML, a coluna da direita representa a evolução ao nível da dimensão dos ficheiros.

Nos gráficos apresentados na figura 5.18 observamos que existe uma conformidade entre todos formatos quer no que diz respeito à dimensão dos ficheiros quer na dimensão dos documentos XML presentes nesses mesmos ficheiros. Como se pode observar existe

uma queda com um declive⁵ muito acentuado do estado inicial para a primeira atualização, invertendo de seguida o sentido de crescimento na segunda para depois não mais parar de descer chegando a valores residuais cerca da trigésima atualização.

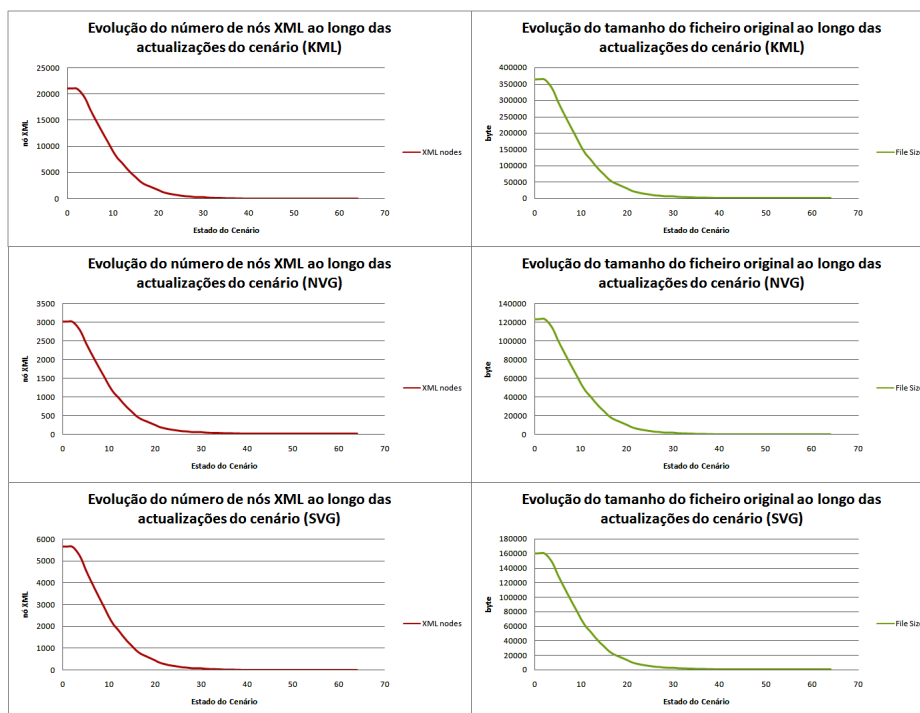


Figura 5.19: Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados (KML em cima, NVG ao centro, SVG em baixo), **sem** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Nos gráficos apresentados na figura 5.19 observamos que existe uma conformidade entre todos formatos quer no que diz respeito à dimensão dos ficheiros quer na dimensão dos documentos XML presentes nesses mesmos ficheiros. Como se pode observar existe um declive zero no início da linha, para depois começar a cair a partir da quarta atualização para depois não mais parar de descer chegando a valores residuais cerca da trigésima quinta atualização.

Nos gráficos apresentados na figura 5.20 observamos que existe uma conformidade com os gráficos apresentados na figura 5.18.

Nos gráficos apresentados na figura 5.21 observamos que existe uma conformidade com os gráficos apresentados na figura 5.19, com a ressalva do gráfico que diz respeito aos valores da dimensão dos ficheiros do formato KML (gráfico no topo à direita), que apresenta uma tendência diferente dos restantes apresentando, a partir da vigésima terceira atualização, uma irregularidade de valores.

⁵O declive mede a inclinação de uma reta face ao eixo das abcissas.

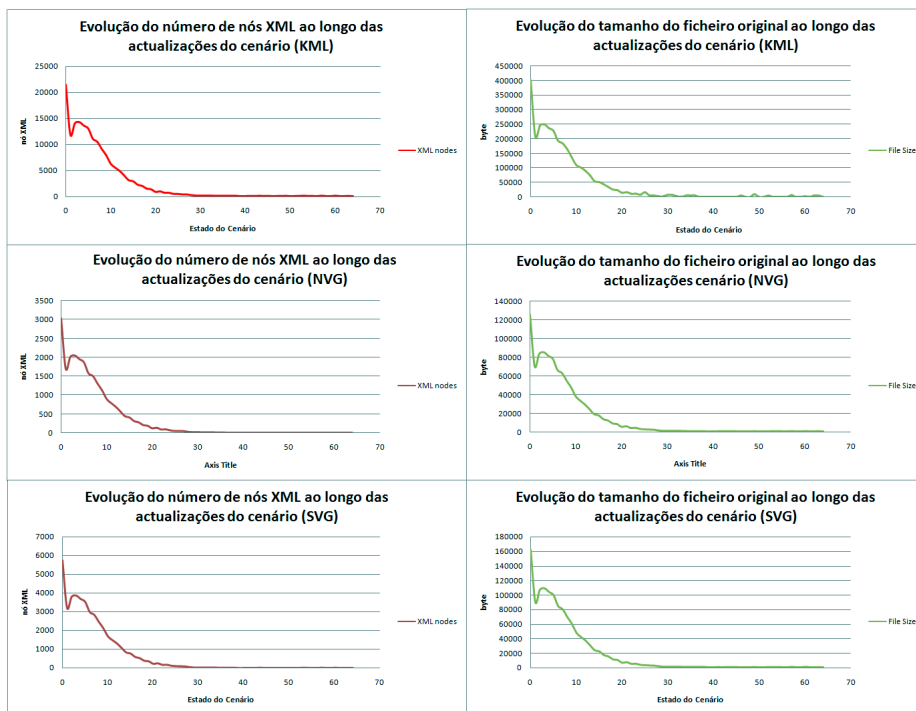


Figura 5.20: Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados, **com** Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

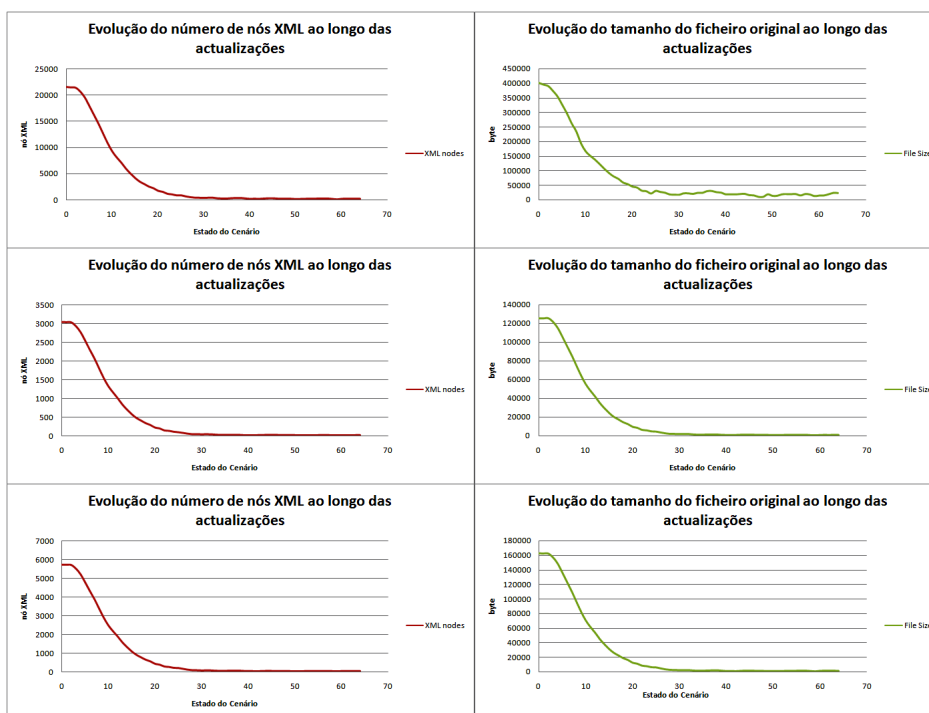


Figura 5.21: Evolução da dimensão dos ficheiros/documentos XML gerados ao longo das 64 atualizações por cada formato de dados (KML em cima, NVG ao centro, SVG em baixo), **com** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Distribuição dos valores acumulados de dimensão dos ficheiros/documentos Os gráficos a seguir apresentados são resultado dos valores acumulados por cada estado do ficheiro (estado original, estado sem espaços em branco e por último estado comprimido) e do documento XML, ao longo dos cenários com número crescente de elementos sujeitos a sessenta e quatro atualizações.

Os gráficos apresentados nas figuras 5.22 e 5.23, relativizam os valores obtidos pelas várias métricas de dimensão. No gráfico da esquerda apresentam-se os valores relativos às métricas de tamanho aplicadas aos ficheiros por formato de dados, no gráfico da direita apresentam-se os valores relativos ao tamanho do documentos XML, por formato de dados.

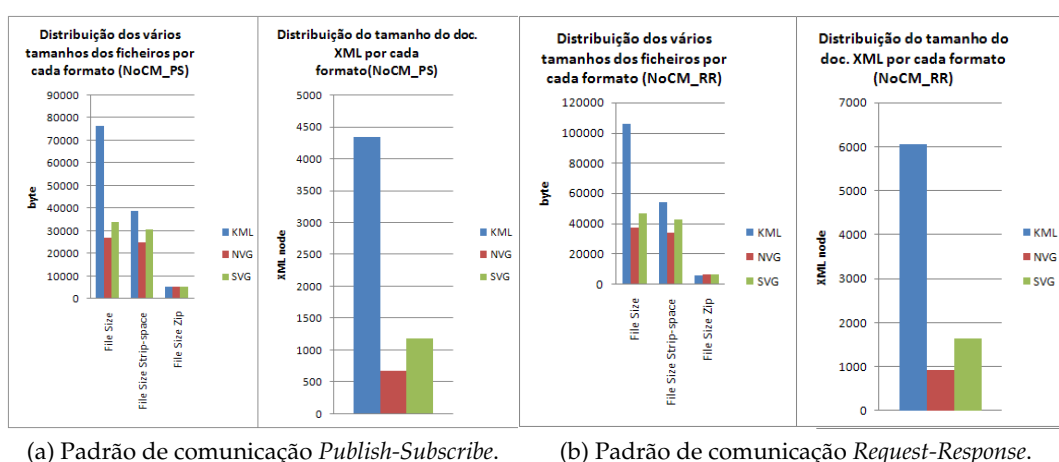


Figura 5.22: Peso de cada formato de dados nas várias métrica de dimensão aplicadas a cenários com 64 atualizações e **sem** Medidas de Coordenação

Nos gráficos apresentados na figura 5.22 observamos que o formato que tem uma maior redução de tamanho com as duas operações (de *Strip-space* e *Zip*) aplicadas ao ficheiro original é o KML. O formato dados KML apresenta uma taxa de redução (taxa de redução = valor inicial / valor final) de 19 vezes inferior ao tamanho original, após a aplicação da operação de compressão *Zip* (sequência de aplicação *Strip-space* e *Zip*) para o padrão de comunicação *Publish-Subscribe* e de 15 vezes inferior ao tamanho original, para o padrão de comunicação *Request-Response*. Os dois restantes formatos de dados apresentam taxas de redução muito inferiores. O NVG apresenta uma taxa de redução 6 vezes inferior ao tamanho original em ambos padrões de comunicação. O SVG apresenta taxas de redução no valor de 7 e 8 vezes inferior ao tamanho original, para os padrões de comunicação *Publish-Subscribe* e *Request-Response* respetivamente. No entanto, como resultado final de tamanho a considerar para a transmissão todos os formatos apresentam valores muito próximos. Ao nível do documento XML a relação de nós apresentados pelo KML (formato com maior valor) é de 6 vezes superior ao número de nós que o NVG (formato com menor valor).

Nos gráficos apresentados na figura 5.22b observamos que os formatos têm o mesmo

comportamento dos gráficos apresentados na figura 5.22a. Apenas se visualiza um aumento generalizado de valores na comparação dos valores de tamanho de ficheiros e tamanho de documentos XML, apresentados pelo protocolo *Publish-Subscribe* com os valores apresentados pelo protocolo *Request-Response*, nas percentagens de: a) Tamanho de ficheiros: 111% KML , 113% NVG e 112% SVG; b) Tamanho de documento XML: 138% KML , 136% NVG e 139% SVG.

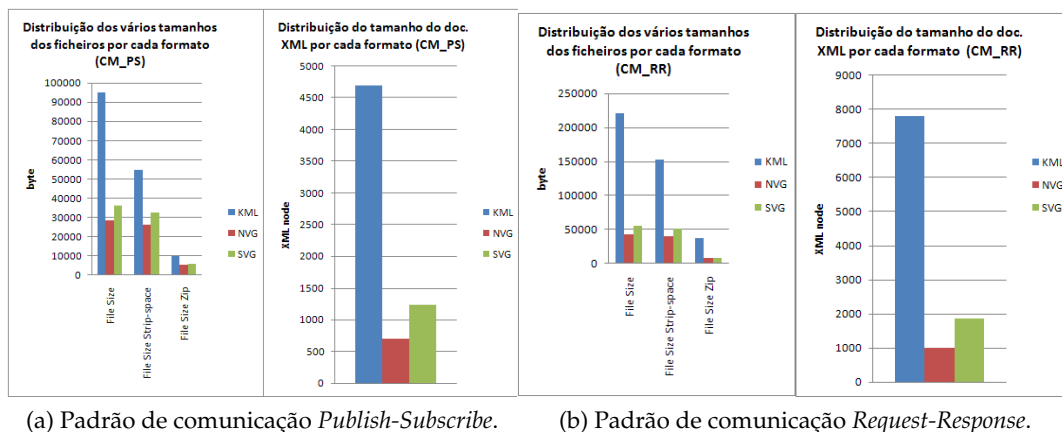


Figura 5.23: Peso de cada formato de dados nas várias métrica de dimensão aplicadas a cenários com 64 atualizações e com Medidas de Coordenação

Nos gráficos apresentados na figura 5.23 observamos que o formato KML no padrão de comunicação *Publish-Subscribe* (ver gráfico na figura 5.23a) apresenta maior redução do tamanho de ficheiro com a aplicação das operações de *Strip-space* e de *Zip*, apresentando uma taxa de redução de 9 vezes inferior ao tamanho original, após a aplicação da operação de compressão *Zip* (sequencia de aplicação *Strip-space* e *Zip*). No entanto, para o padrão de comunicação *Request-Response* (ver gráfico na figura 5.23b) a taxa de redução é de 6 vezes inferior ao tamanho original. Os dois restantes formatos de dados apresentam taxas de redução inferiores no padrão de comunicação *Publish-Subscribe* e equivalentes no padrão de comunicação *Request-Response*. O NVG apresenta uma taxa de redução de 5 e 6 vezes inferior ao tamanho original, para os padrões de comunicação *Publish-Subscribe* e *Request-Response* respetivamente. O SVG apresenta taxas de redução de 6 e 7 vezes inferior ao tamanho original, para os padrões de comunicação *Publish-Subscribe* e *Request-Response* respetivamente. Ao nível do documento XML a relação de nós apresentados pelo KML (formato com maior valor) é de 6 vezes superior ao número de nós que o NVG (formato com menor valor).

Nos gráficos apresentados na figura 5.23b observamos uma redução do KML, após a aplicação da operação de limpeza dos espaços em branco (*Strip-space*), em relação aos restantes. Mas, ao contrário das situações anteriores, verifica-se que, após a aplicação da operação de compressão (*Zip*) o KML perde para o NVG e o SVG. O SVG manifesta um comportamento de aproximação do NVG ao longo das duas referidas operações. No entanto, o NVG mantém em todos os estados o menor valor de tamanho final. Ao nível dos

documentos XML observa-se uma ligeira subida generalizada de valores do padrão de comunicação *Request-Response* (ver gráfico da figura 5.23b) em relação ao padrão de comunicação *Publish-Subscribe* (ver gráfico da figura 5.23a), com as seguintes percentagens em 166% KML, 143% NVG e 151% SVG.

5.3.2 Resultados de métricas de *performance*

Os valores utilizados nos gráficos desta subsecção 5.3.2, resultam da média aritmética dos valores apresentados pelos diversos cenários com diferentes número de atualizações (1, 2, 4, 8, 16, 32, 64), e com o mesmo número de elementos.

Tempo gasto nas diversas operações As figuras 5.24, 5.25, 5.26 e 5.27 sumarizam a aplicação das métricas de *performance* às operações realizadas sobre os ficheiros a serem enviados desde o Produtor até ao Consumidor.

Como se pode visualizar, os gráficos presentes nas figuras 5.24, 5.25, 5.26 e 5.27 têm no seu eixo das ordenadas, o número de elementos que podem ser utilizados para representar cada cenário. No que diz respeito ao eixo das abcissas dos gráficos atrás referidos, têm neste eixo de coordenadas os valores referentes ao tempo gasto, em milissegundos, na aplicação ao ficheiro da respetiva operação. Começando a descrever da esquerda para a direita os gráficos em cada uma das figura, o primeiro, representa o tempo consumido na operação de *Strip-space*, na limpeza dos espaços em branco. O segundo, representa o tempo necessário para comprimir o ficheiro saído da operação anterior, para o formato *Zip*. O terceiro, representa o tempo consumido na operação de descompressão, *UnZip*. A quarta, representa o tempo gasto na validação do documento *XML* de acordo com o *Schema* do respetivo formato de dados. O quinto e último gráfico, representa o tempo Total gasto nas referidas operações por cenário, ao longo das diversas atualizações de cada cenário.

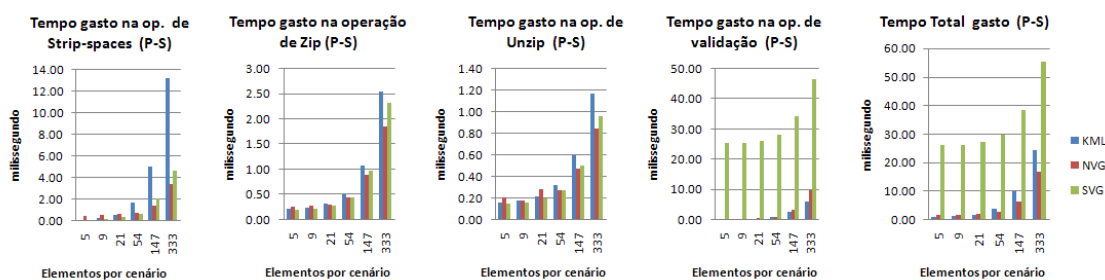


Figura 5.24: Métricas de *performance* aplicadas às operações de *Strip-spaces*, *Zip*, *UnZip* e *Validation* em cenários sem Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

Nos gráficos apresentados na figura 5.24 observamos que todas as operações tendem a gastar mais tempo com o aumento do número de elementos. A operação mais sensível a esse aumento é a operação de limpeza dos espaços em branco (operação de *Strip-space*)

para o formato KML. O SVG apresenta tempos de validação muito maiores que os seus adversários. O NVG apresenta os melhores valores de *Tempo Total* em todos os cenários.

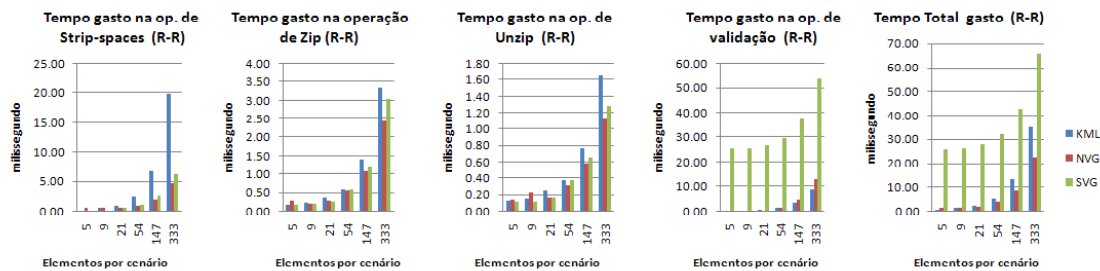


Figura 5.25: Métricas de *performance* aplicadas às operações de *Strip-spaces*, *Zip*, *UnZip* e *Validation* em cenários **sem** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Nos gráficos apresentados na figura 5.25 observamos que existe uma conformidade com os gráficos apresentados na figura 5.24, com a diferença de que os valores são ligeiramente mais altos de forma generalizada.

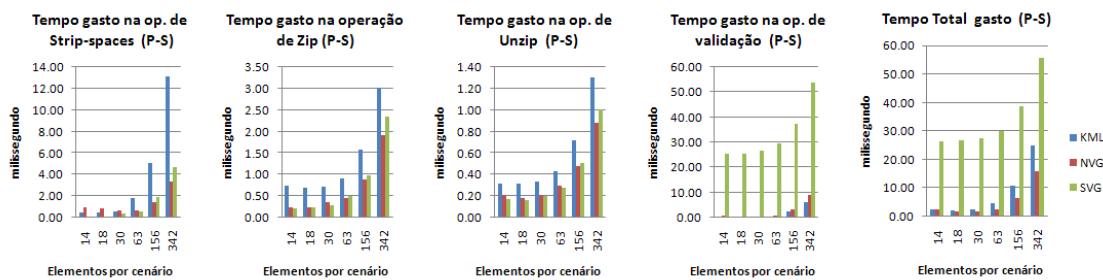


Figura 5.26: Métricas de *performance* aplicadas às operações de *Strip-spaces*, *Zip*, *UnZip* e *Validation* em cenários **com** Medidas de Coordenação para o padrão de comunicação *Publish-Subscribe*.

Nos gráficos apresentados na figura 5.26 observamos que todas as operações tendem a gastar mais tempo com o aumento do número de elementos, mas esta regra só se verifica para cenários com 21 ou mais elementos. A operação de *Strip-space* para o formato KML é, no âmbito geral, a operação em que a variação do tempo gasto ao aumento do número de elementos apresenta um maior valor. Nesta mesma operação de *Strip-space*, verifica-se que o NVG começa por ter os piores resultados, mas inverte a situação a partir dos cenários com 30 elementos, assumindo os melhores resultados a partir de cenários com 156 elementos. Na operação de *Zip*, o KML apresenta sempre os piores resultados. O SVG obtém os melhores resultados até cenários com 30 ou menos elementos, perdendo para o formato NVG nos cenários com maior número de elementos. Também se observa que o KML obtém os piores valores de *performance* referente às operação de compressão *Zip* e de descompressão *Unzip*. No que diz respeito ao *Tempo Total* verifica-se que o NVG é o formato que melhores valores apresenta.

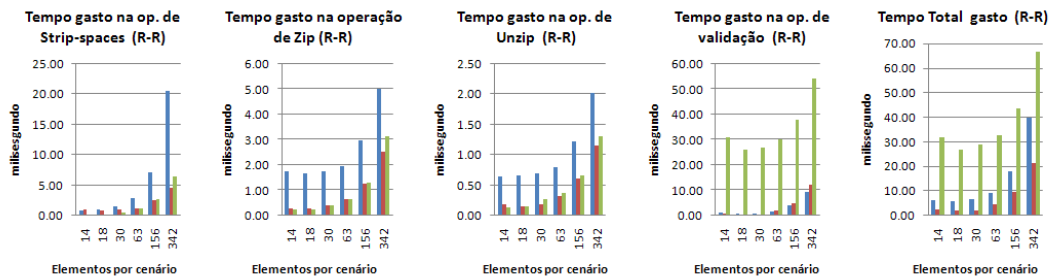


Figura 5.27: Métricas de *performance* aplicadas às operações de *Strip-spaces*, *Zip*, *UnZip* e *Validation* em cenários **com** Medidas de Coordenação para o padrão de comunicação *Request-Response*.

Nos gráficos apresentados na figura 5.27 observamos que existe uma conformidade com os gráficos apresentados na figura 5.26, com a diferença de que os valores são na generalidade mais elevados e o formato KML apresenta uma maior distancia para o NVG nas operações de *Strip-space*, de *Zip* e de *UnZip*, sendo que isso se traduz numa aumentos relativa de desempenho global por parte do NVG em relação ao KML.

Distribuição do tempo consumido A figura 5.28 sumaria a aplicação das métricas de *performance* por operações e por formato (ou seja, inclui análise de resultados de cenários **com** e **sem** Medidas de Coordenação e nos **dois** padrões de comunicação de um dado formato), realizadas sobre os ficheiros a serem enviados desde o Produtor até ao Consumidor.

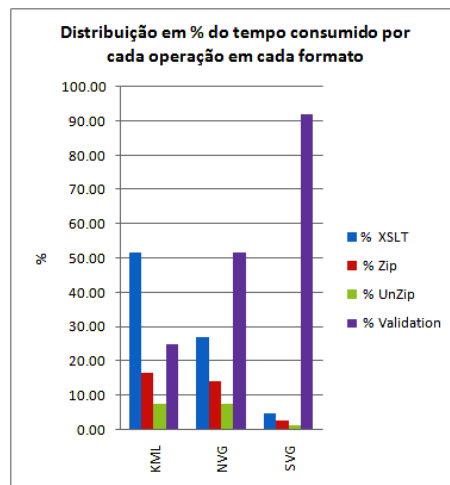


Figura 5.28: Distribuição do tempo consumido, em %, em cada operação por formato de dados.

No gráfico apresentado na figura 5.28 observamos que a operação com maior peso no KML é a operação de XSLT (*Strip-space*), por outro lado a operação com maior peso nos formatos NVG e SVG é a operação *Validation*. Observamos ainda que, a operação que tem menor peso é a operação *UnZip*. Esta operação é comum a todos os formatos de dados.

Pelo mesmo gráfico apresentado na figura 5.28 observamos que a distribuição do tempo consumido por cada operação não é uniforme para todos os formatos. E verificamos que a operação XSLT (*Strip-space*) tem um peso elevado para o formato KML, e a operação de *Validation* tem um peso tão elevado para o SVG que o coloca no formato com um valor de Tempo Total mais elevado.

Relação do tamanho do ficheiro com a *performance* nas diversas operações A relação do tamanho do ficheiro com a *performance* das diversas operações aplicadas durante o processo de partilha do mesmo, é apresentada nos gráficos presentes nas figuras 5.29, 5.30 e 5.31, onde se pode observar a variação de cada uma das referidas operações.

Nos gráficos presentes nas figuras 5.29, 5.30 e 5.31, é apresentada uma reta que é obtida através de uma *regressão linear* dos dados presentes em cada gráfico. Em estatística, uma regressão é considerada uma forma de modelar a relação entre uma variável Y escalar (variável dependente) e uma ou mais variáveis $X_k (k = 1..p)$ (variável independente). Na *regressão linear*, os dados são modelados usando funções lineares, e os parâmetros desconhecidos do modelo são estimados a partir dos dados. No geral, refere-se a *regressão linear* para um modelo no qual a média condicional de Y dado o valor de X_k é uma função afim de X_k . O objetivo da *regressão linear* pode ser explicativo (denotar uma relação de causa-efeito) ou preditivo (encontrar uma relação que possibilite, perante futuras observações das variáveis, independente X_k , prever o correspondente valor de Y , sem necessidade de o medir)[Mat95].

Uma reta é também caracterizada pelo seu declive⁶ que no contexto deste trabalho o declive da reta traduz-se na variação maior ou menor, conforme o valor do declive, do crescimento linearmente do tempo com a dimensão dos ficheiros em análise. A observação do declive da reta nos gráficos que medem o impacto da dimensão do ficheiro no tempo gasto pelas diferentes operações é muito importante para percebermos melhor as diferenças entre os diversos formatos.

Novamente nos gráficos, presentes nas figuras 5.29, 5.30 e 5.31, é apresentado o **coeficiente de determinação** (R^2). R^2 mede proporção da variação de Y em relação à média que é explicada em relação à *regressão linear*. R^2 é também, usado como medida de qualidade do ajuste que varia entre [0,1], a qualidade do ajuste será tanto maior quanto mais R^2 se aproximar da unidade [Mat95].

No gráfico do lado esquerdo da figura 5.29 observamos que para o formato de dados KML, em todas as operações o tempo cresce linearmente com a dimensão do ficheiro. Verifica-se que a operação cujo valor do declive da reta é maior é a operação de *Strip-spaces*, sendo seguida da operação de *Validation*. A operação de *UnZip* é a que apresenta menor declive. Também se observa que (R^2) na operação de *Strip-spaces* e de *Validation* apresenta valores muito próximo da unidade em contraste com as operações de *Zip* e *UnZip*, que apresentam valores de 0,84 e 0,93 respetivamente.

⁶Formula de calculo do valor do declive de uma reta: $d = (y1 - y0)/(x1 - x0)$, com $(x0, y0)$ e $(x1, y1)$ pontos da reta.

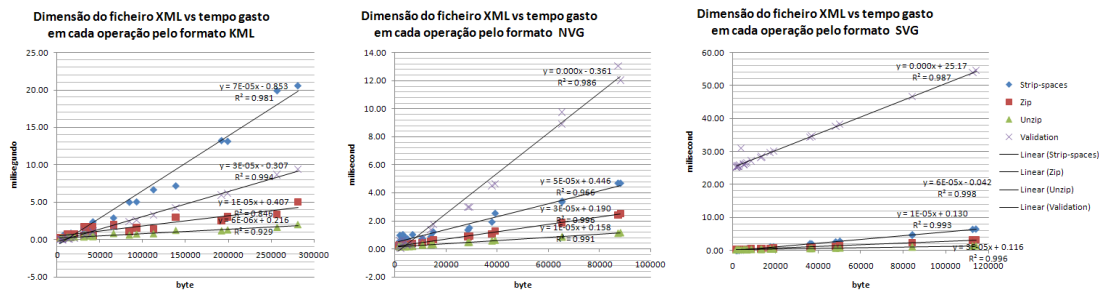


Figura 5.29: Impacto da dimensão do ficheiro, dos diferentes formatos, nas operações de: *Strip-space*, *Zip*, *UnZip* e *Validation*.

No gráfico ao centro da figura 5.29 observamos que para o formato de dados NVG, em todas as operações o tempo cresce linearmente com a dimensão do ficheiro. Verifica-se que a operação com maior declive é a operação de *Validation*, sendo seguida da operação de *Strip-space* com uma considerável diferença de declive. Sendo a operação de *UnZip* a que apresenta menor declive. Observa ainda que (R^2) apresenta todos os valores acima dos 0.967.

No gráfico do lado direito da figura 5.29 observamos que para o formato de dados SVG, em todas as operações o tempo cresce linearmente com a dimensão do ficheiro. Verifica-se que a operação com maior declive é a operação de *Validation*, sendo seguida da operação de *Strip-space* com uma razoável diferença de declive. Esta operação destaca-se das restantes por apresentar um custo residual muito elevado. Sendo a operação de *UnZip* a que apresenta menor valor de declive. Observa ainda que (R^2) apresenta todos os valores muito próximo da unidade, sendo o valor mais baixo de 0.987.

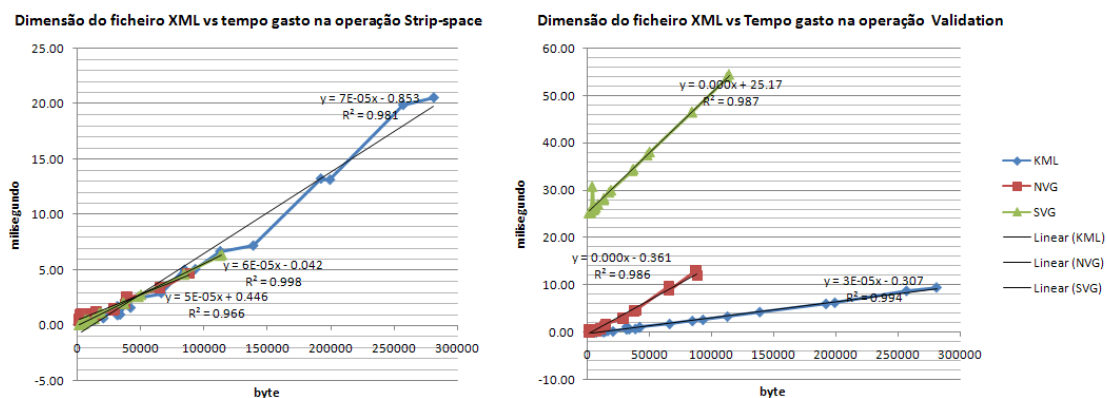


Figura 5.30: Variação das operações de *Strip-space* e *Validation* ao longo do aumento da dimensão dos ficheiros.

Nos gráfico gráficos presentes na figura 5.30 observamos que relativamente à operação *Strip-space* (gráfico do lado esquerdo) todos os formatos assumem um de comportamento idêntico. Relativamente à operação *Validation* (gráfico do lado direito) verifica-se uma completa heterogeneidade de variações, com o SVG a ter um declive acentuado no

sentido crescente, seguido de perto do NVG e por último com um declive muito menor, o KML. Observa-se ainda que para valores muito pequenos de tamanho de ficheiros o tempo gasto na operação de validação pelo formato SVG, assume o valor de **23** milissegundos, em contraste com o formato NVG e KML que assumem valores abaixo de **1** milissegundos.

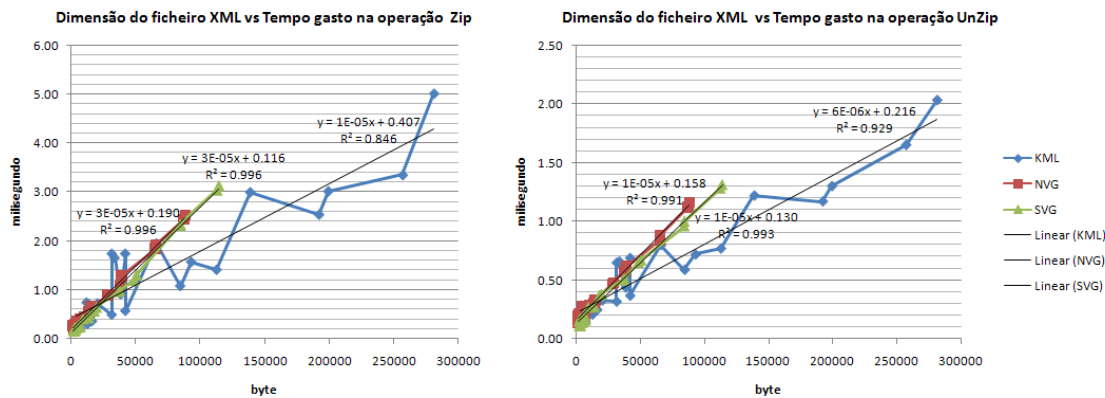


Figura 5.31: Variação das operações de *Zip* e *UnZip* ao longo do aumento da dimensão dos ficheiros.

Nos gráficos presentes na figura 5.31 observamos que relativamente à operação *Zip* (gráfico do lado esquerdo) os formatos NVG e SVG, assumem um comportamento idêntico, com declive ligeiramente inferior à unidade, seguidos do KML com um declive menor, que apresenta um comportamento extremamente irregular. Relativamente à operação *Unzip* (gráfico do lado direito) verifica-se que os comportamentos são em todos os formatos muito idênticos ao gráfico da esquerda, registando-se apenas um ligeira melhoria na irregularidade do formato KML.

5.3.3 Tempo de transmissão teórico

Na experiência levada a cabo para a avaliação quantitativa efetuamos diversas medições de tempos, no entanto, não medimos o tempo de transmissão efetivo dos ficheiros, entre o Servidor e o Cliente. A razão pela qual decidimos não medir este valor, foi a seguinte, o tempo de transmissão de um ficheiro depende de dois fatores: tamanho do ficheiro e Condições da Rede; e qualquer leitura realizada seria sempre dependente dessas duas variáveis, como as Condições da Rede variam de forma não determinística com o fenómeno designado em [KR06] por *Queueing delay*⁷, a leitura desse valor, nada acrescentaria ao nosso estudo.

A título de exemplo, apresentam-se os resultados **estimados** dos Tempo de Transmissão para as diferentes larguras de banda (ver tabela 5.9), tendo como referência o mapeamento de um cenário (c1) utilizado nos testes (escalão de Brigada Independente), com

⁷Tempo que os pacotes (dados) consomem à espera de serem transmitidos nos *routers* espalhados pela Rede de Computadores.

14 elementos (com Medidas de Coordenação) para o padrão de comunicação *Request-Response* (ver figura 5.32).

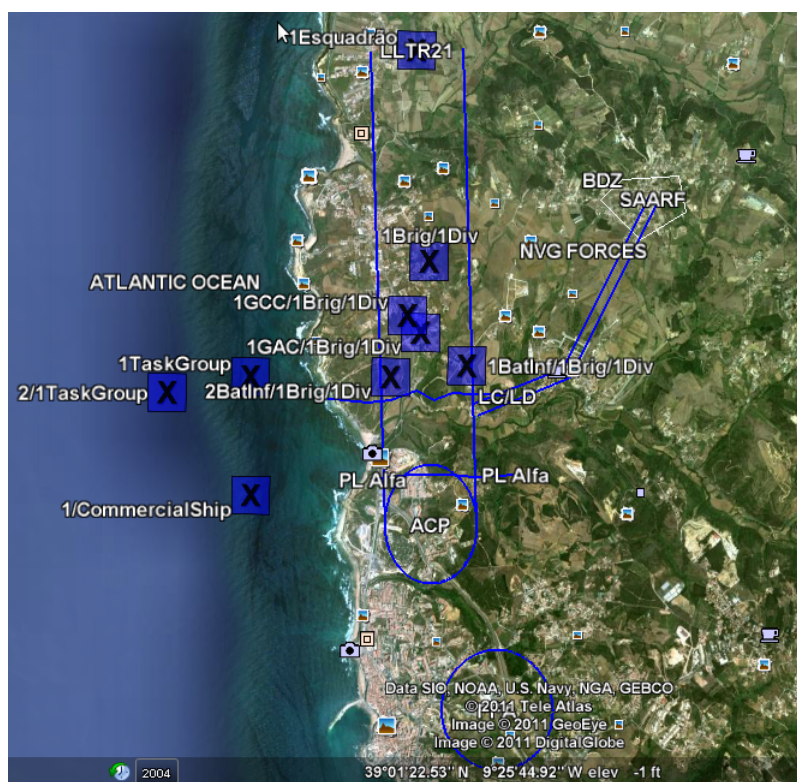


Figura 5.32: Imagem do cenário c1 visualizado no Google Earth.

Tabela 5.8: Tamanho dos ficheiros em cada formato, referente ao cenário c1.

Unidades	Ficheiro KML	Ficheiro NVG	Ficheiro SVG
<i>byte</i>	49055	5372	7188
<i>Kbit</i>	383.242188	41.96875	56.156225

A tabela 5.8 apresenta o tamanho dos ficheiros produzidos pelo formatos de dados NVG, SVG e KML, para partilhar a informação referente ao cenário c1.

Tabela 5.9: Tempo de transferência do cenário XPTO, para Largura de Banda diferentes.

Largura de Banda	Tt KML (s)	Tt NVG (s)	Tt SVG (s)
10 (Kbits)	38.3242188	4.196875	5.6156225
100 (Kbits)	3.83242188	0.4196875	0.56156225
1 (Mbits)	0.383242188	0.04196875	0.056156225
10 (Mbits)	0.0383242188	0.004196875	0.0056156225
100 (Mbits)	0.00383242188	0.0004196875	0.00056156225
1 (Gbits)	0.000383242188	0.00004196875	0.000056156225

Na tabela 5.9 apresentam-se os tempos estimados necessários para transmitir os respetivos ficheiros nos diferentes formato de dados, tendo em conta a *Largura de Banda* da rede existente. A *Largura de Banda* é um valor que mede a quantidade de bits por segundo que cada rede consegue transmitir, esta capacidade depende do meio físico que transmite o sinal.

5.4 Discussão

5.4.1 Resultados da experiência

Nesta experiência foi elaborado um processo de transferência de um ficheiro do Produtor para o Consumidor, por forma a tirar o máximo partido das ferramentas existentes para diminuição do tamanho do ficheiro XML a transmitir, nomeadamente as operações de *Strip-space* e *Zip*. Importa no entanto referir que a operação *Strip-space* é uma operação inócua no que toca a danos colaterais para tecnologias que trabalham com dados em XML, mas o mesmo não se pode dizer em respeito à operação de compressão *Zip* que transforma um *ficheiro XML* num *ficheiro binário*. A utilização de *ficheiros binários* limita o potencial de algumas tecnologias como é o caso dos *Web Services*.

Nesta secção, recorrendo a uma análise crítica, procuramos discutir todos pontos que pensamos serem alvo de maior interesse de debater e procuramos realçar os resultados aparentemente menos óbvios, por forma a oferecer uma explicação clara dos resultados apresentados na secção 5.1 acerca dos formatos de dados NVG, SVG e KML. Nesta experiência estamos interessados em avaliar essencialmente duas vertentes do desempenho, o tempo e o espaço, nesse intuito subdividimos a discussão dos resultados em duas secções: Desempenho dos formatos em relação ao espaço e Desempenho dos formatos em relação ao tempo.

5.4.2 Desempenho dos formatos em relação ao espaço

Apresentamos de seguida a discussão dos resultados à luz do desempenho em ordem ao espaço, dos formatos de dados NVG, SVG e KML .

- A relação do tamanho dos ficheiros entre os diferentes tipos de formatos pode mudar depois de aplicarmos as operações: *Strip-space* e *Compression*. Isto pode ser observado, quer nos gráficos presentes nas figuras 5.10, 5.11, 5.12 e 5.13, quer nos gráficos presentes nas figuras 5.14, 5.15, 5.16 e 5.17. Por exemplo a relação de tamanho entre o KML e o NVG inverte-se ao longo do processo de envio, para cenários sem Medidas de Coordenação. O NVG inicialmente apresenta valores significativamente inferiores ao KML, no entanto no final da operação de *Zip*, o KML apresenta valores de tamanho menores que o NVG para cenários a partir de 147 elementos. Existem pelo menos duas razões para que isto se suceda: a) KML recorre ao *aninhamento* de elementos, enquanto que o NVG usa atributos; isto torna o ficheiro KML

maior por causa da indentação - a operação de *Strip-space* compensa este efeito; b) normalmente, cada elemento KML tem duas *tags*, e a operação de *Zip* é particularmente eficiente na compressão deste tipo de padrão.

- Nos testes **sem** Medidas de Coordenação, todos os formatos apresentam uma diferença de valores, para menos, em relação ao teste realizado **com** Medidas de Coordenação (ver figuras 5.22 e 5.23). Isto acontece porque as medidas de coordenação devido à sua natureza, têm uma estrutura maior relativamente aos elementos do tipo força. O formato de dados KML é o que mais faz notar esta diferença, esta situação explica-se com o facto de que o formato de dados KML, em contraste com o NVG e SVG, não possui as primitivas geométricas *Circle* e *Ellipse*, que são frequentemente usadas na definição das Medidas de Coordenação. Isto implica a utilização de primitivas geométricas não otimizadas para o efeito, como a *Polygon*, para construir círculos ou elipses. Por outro lado, a primitiva *Polygon* é definido à custa de uma sequência de pares/triplos de coordenadas, que ocupam muito espaço no ficheiro original e apresentam uma baixa taxa de compressão pelo algoritmo de compressão usado na transformação para o formato *Zip*. O formato de dados NVG é de todos o que menor alteração apresenta nos testes **com** e **sem** Medidas de Coordenação, porque de todos os formatos é aquele em que a estrutura utilizada na codificação de um elementos do tipo de Medida de Coordenação é a mais idêntica à de um elemento do tipo Força.
- Os padrões de comunicação têm impacto na dimensão dos ficheiros produzidos (ver figuras 5.22 e 5.23). Padrão de comunicação *Publish-Subscribe* produz ficheiros com menores dimensões que o protocolo de comunicação *Request-Response*. Esta situação explica-se por duas razões:a) pela diferença de funcionamento intrínseca de cada protocolo (explicada em 2.2). Uma vez que no padrão de comunicação *Publish-Subscribe*, o Produtor após enviar o cenário inicial para o Consumidor, nas atualizações subsequentes do cenário, apenas envia os elementos que sofreram alteração com essa atualização. O que tem implicação direta na diminuição da dimensão dos ficheiros produzidos em relação ao padrão de comunicação *Request-Response*, que a cada atualização do cenário o Produtor envia para o consumidor todos os elementos que fazem parte do cenário; b) pelo facto de que entre duas atualizações consecutivas do cenário apenas parte dos elementos (Força e Medidas de Coordenação), salvo casos muito excecionais, sofrem alterações.
- Nos cenários **com** Medidas de Coordenação, a utilização do Padrão de Comunicação *Publish-Subscribe* em detrimento do padrão de comunicação *Request-Response* permite ao formato KML, com o aumento do número de atualizações por cenário aproximar-se dos valores, de tamanho final de ficheiro obtidos pelo formato com menor tamanho, o NVG (ver figuras 5.12 e 5.13). Situação que acontece por três

razões: a) pelo modo de funcionamento diferenciado dos dois padrões de comunicação em que o *Publish-Subscribe* ao contrário do *Request-Response*, após o envio do ficheiro inicial com o cenário todo, nas atualizações seguintes envia apenas os elementos alterados (ver explicação 2.2); b) pelo diferencial enorme que existe na taxa de alteração dos elementos do tipo Medidas de Coordenação, com os elementos do tipo Força. As Medidas de Coordenação, por norma, tem uma duração substancial no tempo (de horas ou até mesmo de dias), enquanto que, um elementos do tipo Força se encontrar em deslocamento está constantemente a ser alterado; c) pelo facto que no formato KML os elementos do tipo Medidas de Coordenação e elementos do tipo Força (como se pode ver na figura 5.21) têm ainda maior assimetria no espaço que ocupam, que nos restantes formatos, devido ao facto, já anteriormente explicado, de o KML não possuir as primitivas geométricas: *Circle* e *Ellipse*.

- Um maior documento XML pode não significar, obrigatoriamente, uma maior ficheiro do disco. A estrutura de definição dos elementos é um fator importante a ter em conta na análise de um documento XML, porque como podemos observar nos gráficos apresentados nas figuras 5.22 e 5.23 a razão entre o tamanho em *bytes* e *XML nodes* varia para os vários formatos. Sendo que, no contexto dos cenários **sem** Medidas de Coordenação a razão, em ambos padrões de comunicação, é de **18, 40 e 29** para o KML, NVG e SVG respetivamente. No contexto de cenários **com** Medidas de Coordenação a razão é a seguinte: a) *Publish-Subscribe*: **20, 40 e 29** para o KML, NVG e SVG respetivamente; *Request-Response*: **28, 43 e 30** para o KML, NVG e SVG respetivamente. Este situação explica-se pelo facto de que das metodologias usadas para descrever os diversos tipos de elementos nos respetivos modelo de dados dos formatos de dados serem diferentes. O formato de dados NVG apresenta um modelo de dados mais compacto que os restantes pelo uso intensivo de atributos em detrimento de sub-elementos. O formato de dados KML apresenta um modelo de dados, menos compacto, com uso intensivo de sub-elementos. O formato SVG apresenta-se como um modelo de dados intermédio dos anteriores no que toca a utilização de atributos e sub-elementos na definição dos tipos de dados.
- O linha desenhada pelos gráficos que apresentam a variação da dimensão dos ficheiros ao longo das atualizações não tem a mesma forma para os padrões de comunicação *Request-Response* e *Publish-Subscribe*, porque para o padrão *Request-Response*, os elementos enviados nos ficheiros de atualização do cenário são sempre todos os elementos que existem naquele momento, mas como os elementos vão sofrendo diminuição da sua capacidade de combate até que ficam inoperacionais e são eliminados do cenário, ao longo do tempo os elementos do tipo força vão diminuindo gradualmente dando origem a uma curva que se observa nos gráficos das figuras 5.19 e 5.21. No que diz respeito ao padrão *Publish-Subscribe*, o primeiro ficheiro leva todos o elementos existentes no cenário, mas os seguintes levam apenas o elementos alterados. Logo, a queda acentuada da dimensão dos ficheiros para depois

voltar a subir é natural, pois a dimensão volta a subir dado ao facto da alteração de alguma das Medidas de Coordenação e de elementos do tipo Força, no entanto a diminuição vai voltar a descer pelo facto, já explicado, da eliminação de elementos que ao longo dos combates vão ficando inoperacionais.

- Ficheiros com tamanhos mais pequenos demoram menos tempo a ser partilhados. Os elevados valores de R^2 observados nos gráficos presentes nas figuras 5.30 e 5.31 mostram que, o tamanho do ficheiro tem um impacto linear em todas as operações realizadas ao longo do processo de partilha do ficheiro (limpeza dos espaços em branco, compressão do formato de dados, descompressão do formato de dados e validação do documento XML), por consequência tem impacto no resultado do desempenho global do respetivo formato de dados.

5.4.3 Desempenho dos formatos em relação ao tempo

Apresentamos de seguida a discussão dos resultados à luz do desempenho em ordem ao tempo dos formatos de dados NVG, SVG e KML.

- O tamanho não é o único aspeto com impacto na *performance* do formato de dados. Como podemos observar nos gráficos presentes nas figuras 5.24, 5.25, 5.26 e 5.27, o resultado final da *performance* dos formatos de dados, não tem uma correspondência direta com o tamanho do ficheiro produzido. Por exemplo, podemos observar pelo gráfico presente na figura 5.28 que o formato de dados SVG tem um bom tamanho, e é relativamente rápido em todas as operações com exceção da operação de validação, que consome cerca de 92 por cento, do tempo total consumido. Esta situação prejudica o resultado final da sua *performance*, colocando o SVG sempre, em último lugar. Existem duas razões principais para que esta situação aconteça, são elas: a) como indica o elevado valor R^2 dos gráficos presentes nas figuras 5.30 e 5.31, estamos na presença de variações lineares do tempo com a dimensão, mas, como se percebe do declive das respetivas retas estamos na presença de variações lineares diferenciadas; b) existem outros fatores, para além da dimensão que influenciam o tempo total gasto na transferência do ficheiro, por exemplo, a complexidade do *schema* que define os dados.
- Embora a operação de *Strip-space* tenha um impacto positivo na diminuição do tamanho dos ficheiros de todos os formatos, como se pode observar nos gráficos presentes nas figuras 5.14, 5.15, 5.16 e 5.17, esta produz um maior impacto no formato de dados KML (ver figura 5.28). Naturalmente todas as operações aplicadas durante o processo de transferência do ficheiros envolvem consumos de tempo para qualquer formato de dados, no entanto, o custo em tempo da operação de *Strip-space* para o formato KML chega a representar 51 por cento do valor total do tempo

consumido (ver figura 5.28). Deste modo, esta operação deverá ser equacionada relativamente ao tempo que leva a ser processada, com as condições da Rede de Computadores - condições que podem ser medidas aplicando as métricas apresentadas em [Pra03]: Capacidade⁸, Largura de banda disponível⁹ e *Bulk Transfer Capacity*¹⁰ (BTC) -, que está a servir de meio de transmissão. Pois, se a Rede de Computadores onde estes ficheiros vão ser transmitidos apresentar grande Capacidade, alta disponibilidade de Largura de Banda e alto valor BTC, a diminuição no tamanho do ficheiro que se obtém com a aplicação da operação *Strip-space*, pode não compensar o tempo que se gasta na realização da mesma e, por conseguinte, significar uma diminuição da *performance* no resultado final.

- O tempo gasto na validação de um documento XML não depende exclusivamente da dimensão do ficheiro. Como se pode observar no gráfico da direita presente na figura 5.30, a operação de validação apresenta, tal como nos restantes formatos, um comportamento linear ao longo do crescimento do tamanho do ficheiro, mas contrariamente ao NVG e KML começa com um valor inicial de 25 milissegundos. Este valor inicial de 25, é um valor que se deve não à dimensão do ficheiro mas a fatores externos intrínsecos ao formato de dados que estão presentes sempre que ocorre uma nova operação de validação de um documento XML contido num ficheiro no formato dados SVG. O estudo destes fatores foge ao âmbito deste trabalho e por isso não foram cabalmente investigados, mas a título de exemplo fica a complexidade do *schema* como um fator a ter em conta.
- O tempo gasto na operação de compressão de ficheiros no formato de dados KML de dados não depende exclusivamente da dimensão do ficheiro. Como se pode observar no gráfico do lado esquerdo presente na figura 5.31, o valor de R^2 é de 0.846, embora alto, está muito abaixo dos valores apresentados pelo formato NVG e SVG. Este valor de R^2 significa que a variação do tempo gasto na operação em relação à dimensão do ficheiro, é menos linear que as restantes. Isto acontece por motivos que fogem ao âmbito deste trabalho e por isso não foram cabalmente investigados, mas que têm a ver com o tipo de dados que o compressor está a processar e não só com a quantidade dos mesmos. Por exemplo, ainda no mesmo gráfico da figura 5.31, os saltos que se vêm correspondem a situação que acontece quando se passa da compressão de um cenário sem Medidas de Coordenação e se passa para um cenário com Medidas de Coordenação. Porque a introdução de Medidas de Coordenação no cenário provoca um aumento diferenciado na dimensão do ficheiro e no tempo de compressão do mesmo ficheiro. Isto deve-se fundamentalmente à

⁸Taxa de transferência de dados constante, medida ao nível da *layer 2*, também designada por Taxa de transmissão do Segmento.

⁹Representa a capacidade de uma ligação entre dois computadores (end-to-end) que não está a ser utilizada num dado momento.

¹⁰Representa o máximo débito (transferência de dados por unidade de tempo) obtido por uma conexão TCP (*Transmission Control Protocol*).

carência por parte do formato KML das primitivas *Circle* e *Ellipse* importantes na definição das Medidas de Coordenação com recurso a primitivas geométricas que é o caso do KML que como já foi explicado, são executadas utilizando a primitiva geométrica *Polygon*. Mas para complicar ainda mais a nossa análise, nem todas as Medidas de Coordenação utilizadas necessitam de recorrer à primitiva geométrica *Polygon*, o que faz com que a diferença do impacto diferenciado na dimensão e no tempo de compressão do ficheiro não seja constante.

- O formato de dados que apresenta melhor desempenho global é o NVG. Este facto é observável pelos gráficos presentes nas figuras 5.26 e 5.27, mesmo se o formato de dados KML em situações muito específicas - da não utilização de Medidas de Coordenação-, como mostram as figuras 5.10 e 5.11, apresenta valores de tamanho de ficheiros relativamente menores que o formato de dados NVG. Esta situação acontece porque estamos a considerar o valor final de tamanho, ou seja após a operação de compressão *Zip*, mas inicialmente isso não se verifica, como se observa nos gráficos das figuras 5.10 e 5.11, o KML é cerca de 3 vezes maior que o NVG. O tempo que o KML gasta na operação de *Strip-space* varia linearmente de forma igual para os dois formatos, como se pode observar no gráfico do lado esquerdo da figura 5.30, logo, o tempo gasto pelo KML é muito superior ao NVG. Situação idêntica acontece com o a operação de *Zip*, mas com resultados de ganhos de tempo menores para o NVG por duas razões: a) embora que a NVG continue a apresentar nesta fase tamanho de ficheiros mais pequenos que o KML, o diferencial inicial entre os dois formatos é menor; b) a operação de compressão *Zip* tem um desempenho relativamente menor para o NVG, como se pode ver no gráfico do lado esquerdo presente na figura 5.31. No entanto a diferença de tamanho entre o KML e NVG consegue, mesmo assim dar vantagem ao NVG, no tempo consumido na operação de *Zip*.

5.4.4 Discussão sobre a tempo de transmissão

A estimativa do *Tempo de Transmissão*¹¹ revela-se de especial interesse, para melhor percebermos qual o impacto do tamanho do ficheiro na transmissão do mesmo, visto que, de todos os valores que condicionam o tempo total de envio de um ficheiro na Rede de Computadores, o Tempo de Transmissão é único valor que está relacionado com a dimensão do próprio ficheiro.

Como se pode observar pelo gráfico da figura 5.33, para valores elevados de *Largura de Banda* da Rede de Computadores, os valores gastos com o *Tempo de Transmissão* tendem para valores insignificantes, no entanto, para valores de *Largura de Banda* pequenos, a diferença relativa entre os formatos pode ser considerável. É com esta sensibilidade, que o valor do *Tempo de Transmissão* deve ser sempre analisado à luz da *capacidade* disponível na Rede de Computadores.

¹¹Tempo de transmissão = dimensão do pacote (bit) / velocidade de transmissão (bit por segundo (bps))

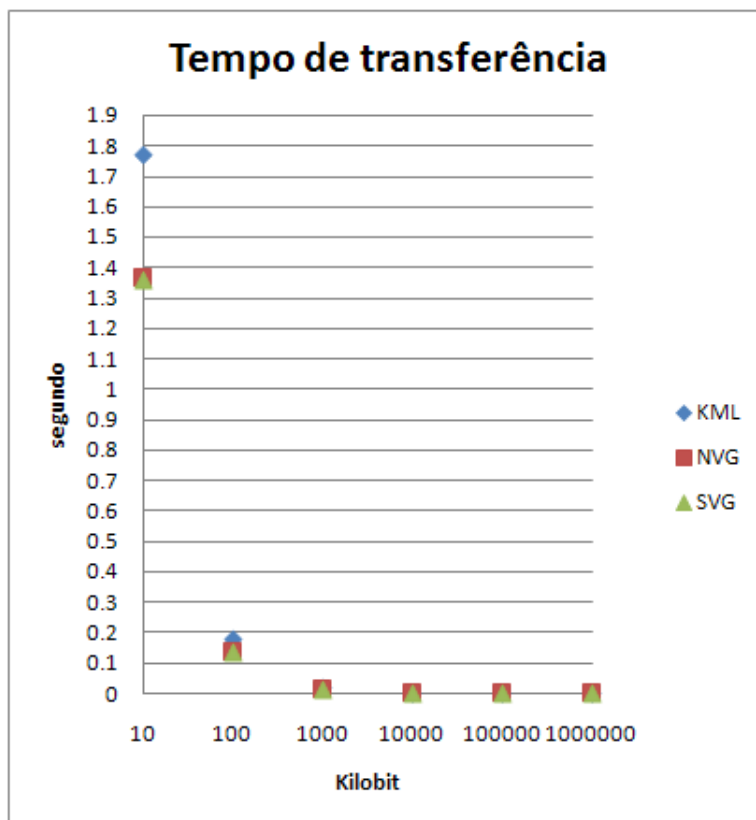


Figura 5.33: Tempo de transferência do cenário C1, para Largura de Banda diferentes.

5.4.5 Ameaças à validação

Na realização desta experiência foram identificadas algumas ameaças à validação da mesma que importa considerar:

- Complexidade e dimensão do Cenário;
- O tamanho da Amostra;
- A linguagem de programação utilizada para realizar a experiência;
- O tipo de *parser* utilizado na validação dos documentos XML;
- O formato de compressão utilizado na operação de compressão;
- O processador *XSLT* utilizado na operação de *Strip-space*;

A dimensão do Cenário No que se refere à dimensão dos cenários gerados, esta pode ser considerada como uma possível ameaça para uma possível extrapolação para cenários de dimensão diferentes. Para mitigar essa ameaça, foram gerados cenários com diferentes dimensões, alterando o nível de escalão. É também de realçar o facto que, na geração dos cenários utilizados para teste foi utilizado como âncora de referência a dimensão das forças que participaram na operação de libertação da cidade de *Ra's al-Khaffi* que aconteceu na primeira Guerra do Golfo, em Janeiro de 1991.

O tamanho da Amostra O tamanho da Amostra é, potencialmente, uma ameaça à validade de experiências deste tipo. Para mitigar essa ameaça, realizamos um total de 84 Cenários diferentes. Cada Cenário foi codificado em 3 formatos diferentes e segundo dois padrões de comunicação o que perfaz um total de 504 situações diferentes para analisar, número que nos permite ter confiança na validade dos resultados obtidos.

A linguagem de programação utilizada para implementar o protótipo usado na realização da experiência A linguagem de programação utilizada para desenvolver as ferramentas quer de simulação quer de medição foi a linguagem Java¹². Sabemos que a linguagem Java tem algumas limitações em termos de desempenho decorrentes de o seu código, *bytecode*, ser interpretado pela *Java Virtual Machine*. Dito isto, e por ter sido usada em todos os testes, os efeitos no desempenho serão, à partida, consistentes nos vários cenários experimentados, ou seja, não deverão contaminar os valores relativos entre vários cenários.

O tipo de XML parsing utilizado na validação dos documentos XML A validação dos documentos contra os *schemas* que definem os diversos formatos de dados, foi realizada recorrendo à API do modelo de XML parsing do tipo DOM¹³ (*Document Object Model*) existente nas bibliotecas da linguagem Java [AG96]. Como se pode observar em [NJ03], existem outros modelos de parsing como é o caso do SAX¹⁴ (*Simple API for XML*) e do JAXP¹⁵ (*Java API for XML Parsing*), analisar as suas vantagens e desvantagens fogem ao âmbito deste trabalho e por isso esse assunto não será aqui abordado. O importante para o este estudo é a fixação desta variável, durante toda a experiência, para os vários formatos de dados. Das alternativas possíveis, escolhemos aquela com que estamos mais familiarizados. A exemplo do que já discutimos relativamente à utilização da linguagem Java, o facto de usarmos sempre a mesma abordagem para o parsing na validação do documento facilita a comparabilidade dos resultados. Dito isto, seria interessante repetir a experiência usando um modelo de XML parsing alternativo e verificar se os resultados são consistentes, ou se esta alteração tem impacto na avaliação relativa dos formatos de transmissão de dados georreferenciados. Tal replicação diferenciada deverá ser realizada em trabalhos futuros.

O formato de compressão utilizado na operação de compressão O formato de compressão para o qual é efetuada a operação de compressão de dados a enviar foi o formato ZIP, fazendo uso do pacote *java.util.zip* como ferramenta para essa compressão. O formato ZIP é um formato de compressão sem perda de dados, condição fundamental para que os dados a transmitir entre sistemas de C2 não se deterioreem no processo. Existem

¹²A linguagem de programação Java foi originalmente desenvolvida por James Gosling para a Sun Microsystems

¹³<http://www.w3.org/DOM/>

¹⁴<http://www.saxproject.org/>

¹⁵<http://jaxp.java.net/>

vários compressores (Gzip, PPM, BWT, MPM e WBXML, entre outros) que utilizam outros algoritmos e técnicas de compressão, que foram alvo de um aprofundado estudo em [Sak09], no qual se pode observar as diferenças entre eles e a *performance* relativa de cada um, no entanto a análise de formatos de compressão está fora do âmbito desta dissertação pelo que essa temática não será aqui abordada. Na escolha do tipo de formato de compressão a utilizar nesta experiência a opção recaiu sobre o ZIP, por ser um dos formatos de compressão mais disseminados, no entanto, o importante para o nosso estudo é o facto que todos os formatos de dados foram alvo de compressão com o mesmo compressor. A fixação de um compressor poderá ser uma ameaça à validação dos resultados obtidos. Uma forma de mitigar essa ameaça é com a realização de réplicas desta experiência variando o formato de compressão, em futuras evoluções deste trabalho.

O processador XSLT utilizado na operação de *Strip-space* A operação de transformação XSL, *Strip-space*, aos documentos XML foi executada recorrendo ao pacote da linguagem Java `javax.xml.transform.sax`. Também nesta situação, admitimos que outros processadores de XSLT poderiam ser utilizados para realizar a referida operação e que poderiam chegar, eventualmente, a resultados distintos. Em todo caso, tal como com outras opções tomadas na montagem da experiência, o facto de termos usado consistentemente o mesmo processador de XSLT garante, no contexto desta experiência, a comparabilidade dos resultados. Uma forma de mitigar esta ameaça seria replicar esta experiência substituindo este processador de XSLT por outro alternativo.



Conclusões e Trabalho Futuro

6.1 Conclusões

Este trabalho centrou-se na avaliação comparativa de formatos de dados no contexto da troca de informação georreferenciada entre os sistemas de C2 no contexto militar. Chegados ao fim desta dissertação oferece-nos dizer com satisfação que todas as principais contribuições propostas na secção 1.4 foram completamente atingidas.

Nesta dissertação procedemos à realização de dois estudos complementares, uma avaliação qualitativa em resposta à primeira questão de investigação **QI1** e uma avaliação quantitativa em resposta à segunda questão de investigação **QI2**. Para fazer face às necessidades da avaliação quantitativa foi desenvolvida uma aplicação, *Scenario simulator*, para servir como ferramenta de apoio, e tem como principais sub-elementos: o *Scenario Simulator*, que gera cenários militares; e o *Metrics Register*, que simula a transferência de um ficheiro e mede e guarda as métricas de tamanho e *performance*.

Comparação qualitativa dos formatos em avaliação, resultante de um estudo sistemático da especificação de cada um deles, segundo os mesmos critérios Este contributo foi alcançado por intermédio de uma avaliação qualitativa realizada no Capítulo 4, os critérios utilizados nesta avaliação qualitativa foram objeto de estudo aprofundado com a produção de uma lista de características refinadas com base de um *feedback* proveniente dos elementos da NATO, onde as principais conclusões estão explanadas na subsecção 6.1.1.

Desenho e implementação de um simulador de cenários militares Este contributo foi alcançado com o desenho e implementação do *Scenario Generator* que se constitui como

uma ferramenta essencial à produção de simulações de cenários para servirem de dados para experiência realizada em secção 5.2.

Seleção de métricas para a avaliação dos vários formatos e implementação de um sistema para a recolha automática das referidas métricas Este contributo foi alcançado pela escolha das métricas de tamanho e de *performance* para transferência de um ficheiro de um Produtor para um Consumidor. Também contribuiu para o alcance deste atributo, o desenho e implementação de uma sistema de recolha automático de métricas, o *Metrics Register*. O *Metrics Register* é sub-elemento do *Scenario Simulator*, que lê e regista numa base de dados, as métricas de tamanho, e as métricas de *performance* nas operações aplicadas aos ficheiros, ao longo do processo de troca de informação entre o Produtor e o Consumidor. Tal como já foi referido no Gerador de Cenários Militares, este conjunto de métricas foi instrumental para a desejada avaliação quantitativa.

Comparação quantitativa entre os diversos formatos de dados georreferenciados, num cenário realista Este contributo foi alcançado com a avaliação quantitativa levada a cabo no capítulo 5, com base numa experiência onde são avaliados três formatos de dados. As principais conclusões retiradas dessa avaliação quantitativa são apresentadas na subsecção 6.1.2.

6.1.1 A avaliação qualitativa

Na avaliação qualitativa o nosso principal desafio foi conseguirmos perceber quais as características dos formatos de dados importantes para o domínio da troca de informação georreferenciada entre sistemas C2. Inicialmente, produzimos uma lista preliminar [Sal11] das características consideradas importantes para um formato de dados possuir neste domínio. Submetemos essa lista ao escrutínio por parte de elementos da NATO e incorporamos o *feedback* na nossa lista final de características apresentada na secção 1.3.

A avaliação qualitativa incidiu no contraste sistemático dos diversos formatos com as características constadas na lista. Esta comparação permitiu-nos conhecer quais as diferenças mais importantes entre os diversos formatos de dados, perceber quais os pontos fortes e fracos de cada um dos formatos de dados, e descobrir quais as vantagens e desvantagens que isso lhes confere no contexto da troca de informação georreferenciada entre sistemas C2.

O GML apresenta-se como um formato de dados com grande potencial na a representação da informação georreferenciada mas, em contra-partida, não possui nenhuma capacidade na apresentação da mesma. O KML apresenta-se como um formato de dados mais vocacionado para a apresentação de informação georreferenciada, mas sem esquecer a valência da representação. O SVG, apresenta-se como um formato de dados vocacionado para a apresentação de informação gráfica, no entanto, adapta-se muito bem à apresentação da informação georreferenciada, tendo mais dificuldades no campo da representação da informação georreferenciada. O NVG apresenta-se como o formato de

dados do domínio vocacionado para as duas vertentes representação e apresentação da informação georreferenciada, de modo balanceado.

Segundo a nossa análise qualitativa o formato NVG é o único formato de dados que possui mecanismos específicos para endereçar simbologia militar o que lhe confere enorme vantagem no domínio da troca de informação georreferenciada entre C2. No entanto, apresenta uma lacuna ao nível da utilização da terceira dimensão. Os formatos de dados GML e KML são os formatos mais completos ao nível da características, no entanto, têm uma lacuna considerada grave, dado que nenhum deles endereça a simbologia militar. O KML apresenta ainda uma reduzida expressividade ao nível das primitivas geométricas, pois não possui primitivas como *Circle* e *Ellipse*. Devido à falta destas primitivas o KML tem de recorrer a uma primitiva menos adequada, *Polygon*, para poder representar círculos e elipses. O formato de dados SVG é o formato de dados menos completo e simultaneamente com mais lacunas em características consideradas importantes, nomeadamente, o seu modelo de dados não é definido pela norma da W3C *XML Schema*, não possibilita a utilização da terceira dimensão, e não endereçar simbologia militar.

6.1.2 A avaliação quantitativa

Como podemos constatar pelos resultados da experiência realizada no capítulo 5, o NVG é o formato de dados que apresentou melhores resultados de *performance* no tempo de transferência de ficheiros nos dois padrões de comunicação para cenários com e sem Medidas de Coordenação. Também no que se refere ao *tamanho* dos ficheiros produzidos o NVG esteve sempre no lugar da frente, com a exceção dos cenários sem Medidas de Coordenação. O formato de dados KML surpreende pelas elevadas taxas de redução de tamanho ao longo do processo partilha do ficheiro XML. No entanto, para cenários com Medidas de Coordenação, a sua prestação fica bastante aquém do NVG, encurtando essa distancia em cenários sem Medidas de Coordenação. O SVG, embora com *tamanhos* de ficheiros muito próximo do NVG, perde para o NVG e KML nos resultados de *performance* em relação ao tempo total gasto no processo de transferência dos ficheiros em ambos os padrões de comunicação e em cenários com e sem Medidas de Coordenação, muito devido ao elevado tempo gasto na operação de *validação*.

Os padrões de comunicação têm implicação quer no tamanho dos ficheiros produzidos, quer na *performance* final do formato, assumindo-se o padrão *Publish-Subscribe* como o padrão que oferece melhor resultados para todos os formatos de dados, sendo KML o formato que mais ganha com a utilização deste padrão. Constata-se também, que a utilização ou não de Medidas de Coordenação pelos cenários influencia os resultados dos vários formatos de dados, sendo certo que, o formato NVG é o formato que menos alteração sofre na sua *performance* global com a introdução de Medidas de Coordenação.

6.1.3 Avaliação Global

Os resultados obtidos nas duas avaliações (qualitativa e quantitativa) realizadas nesta dissertação mostram que o formato de dados NVG é de facto, o formato mais adequado para a troca de informação georreferenciada entre sistemas de C2 no contexto militar, por apresentar uma grande vantagem no manuseamento da Simbologia Militar e ter obtido os melhores resultados ao nível do desempenho quer no dimensão dos ficheiros produzidos, quer em tempo gasto no processo da transferência dos ficheiros do Produtor para o Consumidor.

A reflexão que resulta desta dissertação em relação aos restantes formato de dados, é a constatação que a grande limitação que se opõe às linguagens (que definem os formatos de dados) fora do domínio é a dificuldade em construir algo mais que uma simples visualização da informação neste domínio específico.

6.2 Trabalho futuro

Avaliação Qualitativa

Neste estudo existem diversos tópicos que poderão ser objeto de estudo em trabalhos futuros, por forma a aprofundar a informação acerca do mesmo ou então por forma a alargar o conhecimento acerca do mesmo. Os tópicos em questão foram alvo de uma abordagem pormenorizada na secção 4.4.

Em síntese, os desafios que lançamos para futuros trabalhos, são os seguintes: estudar a precisão na conversão de coordenadas entre os diversos SRC, estudar as primitivas que cada formato disponibiliza e os mecanismos que são utilizados no processo de envio da meta-informação, estudar a capacidade de receber um URI (Uniform Resource Identifiers) e descobrir a localização do recurso de forma autónoma por parte dos formatos de dados, e por fim, estudar de que lado deve estar a decisão do controlo da visualização dos dados.

Avaliação Quantitativa

Ao longo da realização da avaliação quantitativa, deparámos com matérias interessantíssimas quer para ponto de partida de futuros novos estudos, quer para ponto de chegada para trabalhos réplicas desta avaliação quantitativa. No entanto, chamamos à atenção para o facto dos desafios para o futuro relacionados com esta avaliação terem sido alvo de uma abordagem mais pormenorizado na secção 5.4.5. Dito isto, os desafios que deixamos em aberto para futuros trabalhos são nomeadamente, a replicação deste estudo com recurso à utilização de diversas tecnologias: vários compressores de dados, vários modelos de *parsing*, vários processadores *XSLT* e várias linguagens de programação na implementação do *Scenario Simulator*; o estudo da complexidade dos formato de dados

ao nível dos modelos de dados e ao nível dos dados, e o estudo do impacto da complexidade do modelo de dados dos formatos na operação de validação dos documentos XML.

É também um excelente desafio para trabalhos futuros nesta área, a replicação da experiência com recurso a sistemas de C2 reais. Esta alteração seria muito importante para confirmar ou negar as conclusões apresentadas na secção 6.1.2 que são proveniente de um simulador com todas as implicações que isso possa ter.

Bibliografia

- [AC02] Carl Reed Bill Lalonde Greg Buehler Adrian Cuthbert, Craig Bruce. Styled layer descriptor implementation specification. Relatório técnico, Open Geospatial Consortium, Inc, 2002.
- [AG96] K. Arnold e J. Gosling. *The Java programming language*. The Java series. Addison-Wesley Pub. Co., 1996.
- [BLHL01] Tim Berners-Lee, James Hendler, e Ora Lassila. The semantic web. *Scientific American*, 284(5):34–43, Maio 2001.
- [BM07] Dilek Basci e Sanjay Misra. Complexity metric for xml schema documents. 2007.
- [BM09] Dilek Basci e Sanjay Misra. Measuring and evaluating a design complexity metric for xml schema documents. *Journal of Information Science and Engineering*, 25:1405–1425, 2009.
- [BMMC06] Sr. VP Brenda M. Michelson e Sr. Consultant. Event-driven architecture overview. Relatório técnico, Patricia Seybold Group / Business-Driven ArchitectureSM, February 2006.
- [Cag01] K. Cagle. *Professional XML schemas*. Programmer to programmer. Wrox Press, 2001.
- [CF09] Leonardo Azevedo Fernanda Baião e Flávia Santoro Camille Furtado, Vinícios Pereira. Arquitetura orientada a serviço - conceituação. Relatórios técnicos do dia/unirio, no. 00012/2009, UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO, CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA, Julho 2009.
- [DGB06] Robert I. Dunfey, Bruce M. Gittings, e James K. Batcheller. Towards an open architecture for vector gis. *COMPUTERS & GEOSCIENCES*, 32(10):1720–1732, DEC 2006.

- [DLR11] A. M. Mocanu (Virgolici) I. Surugiu D. Litan, M. Velicanu e O. Raduta. Information technology standards - a way to bring quality and performance in the software products field. *INTERNATIONAL JOURNAL OF COMPUTERS AND COMMUNICATIONS*, Issue 3, Volume 5:119 – 133, 2011.
- [doc08] Allied procedural publication 6b. Relatório técnico, 2008.
- [dOLJ02] Paulo de Oliveira Lima Junior. Suporte à edição cooperativa de informação geográfica em ambiente web, 2002.
- [eJAR07] Fábio Brandão e João Araújo Ribeiro. Estudo do xml, gml, svg e webservices (wms e wfs) para formatação e divulgação de informações geográficas., 2007.
- [Far07] Nuno André de Sampaio Faria. Suporte à edição cooperativa de informação geográfica em ambiente web, 2007.
- [FS07] Rúben Fonseca e Alberto Simões. Alternativas ao XML: YAML e JSON. In José C. Ramalho, João C. Lopes, e Luís Carríço, editores, *XATA 2007 — 5ª Conferência Nacional em XML, Aplicações e Tecnologias Aplicadas*, pág. 33–46, Fevereiro 2007.
- [fXS06] Structure Metrics for XML Schema. Structure metrics for xml schema. In *Proceedings of XATA 2006*, 2006.
- [GA04] Harumi Kuno e Vijay Machiraju Gustavo Alonso, Fabio Casati. *Web Service*. Springer, 2004.
- [Gro07] GML Working Group. Opengis® geography markup language (gml) encoding standard. Opengis® standard, Open Geospatial Consortium Inc, August 2007.
- [Gro09] SVG Working Group. Scalable vector graphics (SVG) 1.1 specification. W3C working draft, World Wide Web Consortium, 2009. <http://www.w3.org/TR/2003/REC-SVG11-20030114/>.
- [GT11] Bruno Gil e Paulo Trezentos. Impacts of data interchange formats on energy consumption and performance in smartphones. In *Proceedings of the 2011 Workshop on Open Source and Design of Communication, OSDOC '11*, pág. 1–6, New York, NY, USA, 2011. ACM.
- [HW03] Gregor Hohpe e Bobby Woolf. *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*. Addison-Wesley Professional, 2003.
- [KKLL09] Joon-Seok Kim, Hye-Young Kang, Tae-Hoon Lee, e Ki-Joune Li. Topology of the prism model for 3d indoor spatial objects. *Mobile Data Management, IEEE International Conference on*, 0:698–703, 2009.

- [KR06] J.F. Kurose e K.W. Ross. *Computer Networking: A Top-Down Approach Featuring the Internet*. Addison-Wesley Longman, Incorporated, 2006.
- [KSH02] Meike Klettke, Lars Schneider, e Andreas Heuer. Metrics for xml document collections. In *Proceedings of the Workshops XMLDM, MDDE, and YRWS on XML-Based Data Management and Multimedia Engineering-Revised Papers*, EDBT 02, pág. 15–28, London, UK, UK, 2002. Springer-Verlag.
- [Lak08a] Ron Lake. Exploring schema constructs in gml. August 2008.
- [Lak08b] Ron Lake. Exploring schema constructs in kml. May 2008.
- [LHC06] Zi Lin, Bingsheng He, e Byron Choi. A quantitative summary of xml structures. *Conceptual Modeling - ER 2006*, pág. 228–240, 2006.
- [Mat95] Manuel António Matos. *Manual Operacional para a Regressão Linear*. Faculdade de Engenharia da Universidade do Porto, 1995.
- [Mem09] Multilateral Interoperability Programme Members. The joint c3 information exchange data model. Relatório técnico, North Atlantic Treaty Organisation, 2009.
- [MK07] Saxonica Michael Kay. Xsl transformations (xslt) version 2.0. Relatório técnico, World Wide Web Consortium, January 2007.
- [MS04] Andrew Mcdowell e Chris Schmidt. Analysis and metrics of xml schema. *Work*, pág. 538–544, 2004.
- [MSbY04] Andrew McDowell, Chris Schmidt, e Kwok bun Yue. Analysis and Metrics of XML Schema. In Hamid R. Arabnia e Hassan Reza, editores, *Software Engineering Research and Practice*, pág. 538–544. CSREA Press, 2004.
- [NC308a] NC3A. Nato vector graphics app-6a bindings. Relatório técnico, North Atlantic Treaty Organization, 2008.
- [NC308b] NC3A. Nato vector graphics app-6b bindings. Relatório técnico, North Atlantic Treaty Organization, 2008.
- [NC309] NC3A. Nato vector graphics mil-std-2525b bindings. Relatório técnico, North Atlantic Treaty Organization, 2009.
- [NC310a] NC3A. Establishment of a c4isr integration fund. Relatório técnico, North Atlantic Treaty Organization, 2010.
- [NC310b] NC3A. Nato vector graphics protocol. Relatório técnico, North Atlantic Treaty Organization, 2010.

- [NJ03] Matthias Nicola e Jasmi John. Xml parsing: a threat to database performance. In *Proceedings of the twelfth international conference on Information and knowledge management, CIKM '03*, pág. 175–178, New York, NY, USA, 2003. ACM.
- [NN09] Randall Reynolds Clemente Izurieta Nurzhan Nurseitov, Michael Paulson. Comparison of JSON and XML Data Interchange Formats: A Case Study. In *CAINE 2009*, 157-162, 2009.
- [(NS07] NATO STANDARDIZATION AGENCY (NSA). Allied joint publication (ajp) 3.2.2, command and control of allied land forces. Allied joint publication (ajp) 3.2.2, command and control of allied land forces, NORTH ATLANTIC TREATY ORGANIZATION, May 2007.
- [oD97] Department of Defense. C4isr architecture framework. Relatório técnico, Departmente of Defense of United States of America, 1997.
- [OGC08] OGC. Ogc® kml. Relatório técnico, Open Geospatial Consortium, Inc, 2008.
- [Ols09] Erik Olsson. Exporting 3d geoinformation from baggis database to citygml, 2009.
- [Por05] Exército Português. Regulamento de campanha - operações. Relatório técnico, Ministério da Defesa Nacional, Rua do Museu de Artilharia, 1149-065 LISBOA, PORTUGAL, Setembro 2005.
- [Pra03] C.; Murray M.; Claffy K.; Georgia Inst. of Technol. Atlanta GA USA Prasad, R.; Dovrolis. Bandwidth estimation: metrics, measurement techniques, and tools. *Network, IEEE*, 17 Issue:6:27 – 35, 2003.
- [PSH10] Christian Pichler, Michael Strommer, e Christian Huemer. Size matters!? measuring the complexity of xml schema mapping models. *Services, IEEE Congress on*, 0:497–502, 2010.
- [QS05] M H Qureshi e M H Samadzadeh. Determining the complexity of xml documents. *ITCC 2*, pág. 416–421, 2005.
- [RGCM04] R. Ramachandran, S. Graves, H. Conover, e K. Moe. Earth Science Markup Language (ESML): A solution for scientific data-application interoperability problem. *Computers and Geosciences*, 30(1):117–124+, 2004.
- [RL05] Dave Remy Ralf Lammel, Stan Kitsis. Analysis of xml schema usage. In *XML 2005 Conference proceeding by RenderX*, nov 2005.
- [Roc05] Jorge Gustavo Pereira Bastos Rocha. Informação geográfica: Meta-informação, codificação e visualização, 2005.
- [Sak09] Sherif Sakr. Xml compression techniques: A survey and comparison. *J. Comput. Syst. Sci.*, 75:303–322, August 2009.

- [Sal11] Francisco Salvador. Avaliação comparativa de formatos de dados georreferenciados em contexto militar. Relatório técnico, Faculdade de Ciências e Tecnologia/Universidade de Lisboa, Monte da Caparica, Almada, Fevereiro 2011.
- [San08] B. Sandvik. Using KML for Thematic Mapping. *AGU Fall Meeting Abstracts*, pág. B2+, Dezembro 2008.
- [Sha03] Mary Shaw. Writing good software engineering research papers: minitutorial. In *Proceedings of the 25th International Conference on Software Engineering, ICSE '03*, pág. 726–736, Washington, DC, USA, 2003. IEEE Computer Society.
- [Spa10] Nicholas Spanos. 100 IT Performance Metrics. article, Computer Aid, Inc., 2010.
- [SY08] Shaotao Yuan Ruan RenZong Sheng Ye, Feng XueZhi. Gml — an open-standard geospatial data format. In *THE INTERNATIONAL ARCHIVES OF THE PHOTOGRAMMETRY, REMOTE SENSING AND SPATIAL INFORMATION SCIENCES*, Volume XXXVII, Part B4, Commission IV, pág. 1773–1778, Washington, DC, USA, 2008. International Society for Photogrammetry and Remote Sensing.
- [Trn05] Milan Trnini. Oo, xml, and gml: Are angle brackets a flexible modeling material? <http://www.dotmlpfi.com/>, 2005. Último acesso: 25Jan2011.
- [Vis06] Joost Visser. Structure metrics for xml schema. *Proceedings of XATA*, pág. 1–10, 2006.
- [W3C] W3C. Extensible Markup Language (XML) 1.1.
- [weba] Dotmlpfi. <http://www.dotmlpfi.com/>. Último acesso: 25Jan2011.
- [webb] Gml complexity. <http://www.galdosinc.com/archives/186>. Último acesso: 20Jan2011.
- [webc] KML tutorial - KML - google code. http://code.google.com/intl/pt-PT/apis/kml/documentation/kml_tut.html. Último acesso: 06Jan2011".
- [webd] NATO - what is NATO? http://www.nato.int/cps/en/SID-F7AE6C8A-01A8B373/natolive/what_is_nato.htm. Último acesso: 04Jan2011.
- [webe] NATO network enabled capability (NNEC). http://www.nato.int/cps/en/natolive/topics_54644.htm. Último acesso: 04Jan2011.
- [webf] tutorials.jenkov.com. <http://tutorials.jenkov.com/svg/index.html>. Último acesso: 18Jan2011".

- [web08] SOA consortium core document. http://www.soa-consortium.org/SOAs_Core-Document.pdf, 2008. Último acesso: 01Set2011.
- [Wes08] Paul W. Westermeyer. *The Battle of al-Khaffi*. History Division, United States Marine Corps, Washington, DC, 2008.
- [Wil08] Tim Wilson. Ogc® kml. Relatório técnico, Open Geospatial Consortium, Inc, Abril 2008.
- [Zha08] Yanyin Zhang. 60-510 literature review and survey xml schema metrics winter 2008. 2008.
- [ZZ03] Rong Li 1 Shunliang Cao e Yangyong Zhu Zhongping Zhang. Similarity metric for xml documents. Relatório técnico, Fudan University, ShangHai China 200433, Yanshan University, HeBei China 066004 NingBo University, ZheJiang China 315211, 2003.



Anexo A - Tabela de resultados

Para efeitos de completude desta dissertação resolvemos juntar neste anexo a tabela de valores resultantes da experiência realizada no capítulo 5, com os quais foram produzidos os gráficos apresentados na secção 5.3.

Coord Measures	Comm Pattern	Elements	Update Cicles	Format	XML Nodes	File Size	File Size Strip-space	File Size Zip	Time XSLT (ms)	Time Zip (ms)	Time UnZip (ms)	Time Valid(ms)
FALSE	EDA	5	1	KML	318	5544	2856	889	0.67	0.75	0.67	0.00
FALSE	EDA	5	1	NVG	54	1986	1811	807	0.33	0.31	0.34	0.00
FALSE	EDA	5	1	SVG	87	2402	2166	833	0.67	0.72	0.66	25.67
FALSE	EDA	5	2	KML	315	5492	2835	902	0.33	0.60	0.61	0.00
FALSE	EDA	5	2	NVG	54	1974	1801	814	0.67	0.74	0.66	0.00
FALSE	EDA	5	2	SVG	86	2376	2142	836	0.33	0.51	0.52	26.00
FALSE	EDA	5	4	KML	307	5364	2768	897	1.00	0.71	0.64	0.00
FALSE	EDA	5	4	NVG	53	1927	1758	810	0.67	0.71	0.65	0.33
FALSE	EDA	5	4	SVG	84	2329	2102	835	0.67	0.69	0.65	25.67
FALSE	EDA	5	8	KML	257	4497	2346	866	0.33	0.66	0.58	0.00
FALSE	EDA	5	8	NVG	46	1665	1522	784	0.33	0.63	0.59	0.00
FALSE	EDA	5	8	SVG	70	1989	1800	810	0.33	0.63	0.58	25.33
FALSE	EDA	5	16	KML	207	3681	1953	841	0.67	0.71	0.65	0.00
FALSE	EDA	5	16	NVG	38	1410	1292	754	0.67	0.68	0.62	0.00
FALSE	EDA	5	16	SVG	57	1663	1511	785	0.33	0.62	0.57	24.67
FALSE	EDA	5	32	KML	115	2099	1171	664	0.67	0.69	0.66	0.00
FALSE	EDA	5	32	NVG	25	913	841	618	0.33	0.66	0.65	0.00
FALSE	EDA	5	32	SVG	32	1049	968	655	0.33	0.60	0.58	24.67
FALSE	EDA	5	64	KML	69	1296	776	590	0.67	0.68	0.65	0.00
FALSE	EDA	5	64	NVG	19	659	612	558	0.33	0.66	0.63	0.00
FALSE	EDA	5	64	SVG	19	738	692	598	0.33	0.61	0.60	24.67
FALSE	EDA	9	1	KML	539	9268	4666	979	1.33	0.79	0.70	0.00
FALSE	EDA	9	1	NVG	86	3168	2882	918	0.67	0.74	0.67	0.67
FALSE	EDA	9	1	SVG	146	3900	3496	928	0.67	0.74	0.67	25.67
FALSE	EDA	9	2	KML	504	8679	4381	971	0.67	0.69	0.66	0.33
FALSE	EDA	9	2	NVG	81	2983	2714	904	1.00	0.49	0.14	0.33
FALSE	EDA	9	2	SVG	137	3662	3284	917	0.67	0.74	0.67	26.33
FALSE	EDA	9	4	KML	509	8771	4423	978	1.33	0.78	0.69	0.00

Figura A.1: Tabela com os resultados.

FALSE	EDA	9	4	NVG	81	3017	2746	914	0.67	0.73	0.66	1.00
FALSE	EDA	9	4	SVG	138	3701	3320	923	0.33	0.65	0.62	26.00
FALSE	EDA	9	8	KML	492	8500	4303	983	1.00	0.76	0.68	0.00
FALSE	EDA	9	8	NVG	79	2947	2684	913	0.67	0.74	0.67	0.33
FALSE	EDA	9	8	SVG	134	3591	3222	924	0.33	0.66	0.60	26.00
FALSE	EDA	9	16	KML	287	5016	2602	853	0.67	0.74	0.67	0.00
FALSE	EDA	9	16	NVG	50	1838	1680	780	0.00	0.17	0.14	0.00
FALSE	EDA	9	16	SVG	79	2212	2000	810	0.33	0.62	0.58	25.00
FALSE	EDA	9	32	KML	179	3179	1698	723	0.67	0.72	0.67	0.00
FALSE	EDA	9	32	NVG	34	1258	1154	672	0.67	0.70	0.66	0.00
FALSE	EDA	9	32	SVG	49	1486	1356	706	0.33	0.64	0.59	24.67
FALSE	EDA	9	64	KML	117	2123	1181	630	0.67	0.71	0.66	0.00
FALSE	EDA	9	64	NVG	25	922	849	594	0.33	0.67	0.64	0.00
FALSE	EDA	9	64	SVG	33	1067	984	631	0.33	0.64	0.63	24.67
FALSE	EDA	21	1	KML	1169	20004	9932	1160	1.00	0.33	0.20	0.67
FALSE	EDA	21	1	NVG	176	6647	6040	1163	0.67	0.80	0.70	3.00
FALSE	EDA	21	1	SVG	316	8338	7454	1157	1.00	0.75	0.70	28.00
FALSE	EDA	21	2	KML	1169	20020	9947	1185	1.67	0.88	0.75	0.00
FALSE	EDA	21	2	NVG	176	6661	6054	1180	0.67	0.81	0.71	2.67
FALSE	EDA	21	2	SVG	316	8343	7459	1168	1.33	0.83	0.72	27.33
FALSE	EDA	21	4	KML	1148	19677	9787	1193	1.67	0.88	0.74	0.00
FALSE	EDA	21	4	NVG	173	6573	5977	1185	0.67	0.82	0.71	2.00
FALSE	EDA	21	4	SVG	311	8200	7332	1180	1.33	0.75	0.66	27.00
FALSE	EDA	21	8	KML	1041	17861	8903	1179	1.33	0.87	0.74	0.00
FALSE	EDA	21	8	NVG	158	5996	5454	1159	0.67	0.73	0.64	1.00
FALSE	EDA	21	8	SVG	282	7453	6666	1151	0.33	0.75	0.65	26.67
FALSE	EDA	21	16	KML	739	12748	6408	1075	1.00	0.83	0.72	0.00
FALSE	EDA	21	16	NVG	114	4355	3967	1026	0.67	0.76	0.68	1.33
FALSE	EDA	21	16	SVG	201	5374	4817	1030	0.33	0.71	0.64	26.00
FALSE	EDA	21	32	KML	392	6821	3495	809	0.67	0.74	0.68	0.00
FALSE	EDA	21	32	NVG	65	2445	2233	772	0.67	0.72	0.68	0.33

Figura A.2: Tabela com os resultados (continuação).

FALSE	EDA	21	32	SVG	107	2986	2694	799	0.33	0.65	0.61	25.00
FALSE	EDA	21	64	KML	197	3494	1856	638	0.67	0.71	0.67	0.00
FALSE	EDA	21	64	NVG	37	1372	1259	616	0.33	0.69	0.68	0.00
FALSE	EDA	21	64	SVG	54	1638	1495	650	0.33	0.63	0.61	24.67
FALSE	EDA	54	1	KML	2891	49640	24617	1654	3.33	1.20	0.91	1.33
FALSE	EDA	54	1	NVG	422	16436	14955	1800	1.67	1.03	0.82	4.33
FALSE	EDA	54	1	SVG	781	20861	18665	1739	1.33	1.01	0.82	31.00
FALSE	EDA	54	2	KML	2800	48141	23907	1655	3.00	1.19	0.89	1.00
FALSE	EDA	54	2	NVG	409	15994	14559	1783	1.33	1.03	0.81	7.00
FALSE	EDA	54	2	SVG	756	20261	18135	1746	1.67	0.98	0.79	31.00
FALSE	EDA	54	4	KML	3013	51821	25740	1776	3.00	1.23	0.91	1.33
FALSE	EDA	54	4	NVG	439	17216	15673	1911	1.67	1.06	0.83	3.33
FALSE	EDA	54	4	SVG	814	21771	19483	1846	1.33	1.09	0.85	31.33
FALSE	EDA	54	8	KML	2616	45066	22433	1693	3.00	1.11	0.86	1.00
FALSE	EDA	54	8	NVG	383	15050	13708	1789	1.33	1.02	0.81	5.00
FALSE	EDA	54	8	SVG	706	18977	16991	1746	1.33	1.01	0.80	30.33
FALSE	EDA	54	16	KML	1984	34235	17084	1507	2.33	1.06	0.83	0.67
FALSE	EDA	54	16	NVG	292	11506	10485	1553	0.67	0.91	0.74	5.33
FALSE	EDA	54	16	SVG	537	14464	12959	1527	1.00	0.91	0.76	29.00
FALSE	EDA	54	32	KML	1118	19339	9703	1122	1.33	0.89	0.75	0.00
FALSE	EDA	54	32	NVG	169	6600	6019	1130	0.67	0.82	0.70	1.67
FALSE	EDA	54	32	SVG	303	8249	7403	1135	0.67	0.78	0.68	26.67
FALSE	EDA	54	64	KML	556	9697	4940	799	0.67	0.82	0.76	0.00
FALSE	EDA	54	64	NVG	88	3436	3140	794	0.67	0.76	0.67	0.00
FALSE	EDA	54	64	SVG	151	4262	3844	819	0.33	0.69	0.65	25.67
FALSE	EDA	147	1	KML	7437	128600	64101	2744	8.00	2.16	1.31	4.33
FALSE	EDA	147	1	NVG	1071	43007	39217	3339	2.33	1.26	0.72	9.67
FALSE	EDA	147	1	SVG	2008	55156	49496	3091	3.67	1.87	1.17	39.67
FALSE	EDA	147	2	KML	7343	126984	63306	2854	7.67	2.04	1.30	4.00
FALSE	EDA	147	2	NVG	1058	42433	38690	3400	2.00	1.16	0.56	7.33
FALSE	EDA	147	2	SVG	1982	54506	48918	3201	3.33	1.87	1.18	39.67

Figura A.3: Tabela com os resultados (continuação).

FALSE	EDA	147	4	KML	7384	127916	63873	3029	7.33	2.16	1.33	4.00
FALSE	EDA	147	4	NVG	1063	42940	39176	3547	2.67	1.69	1.10	13.00
FALSE	EDA	147	4	SVG	1994	54893	49273	3328	3.67	1.89	1.18	40.33
FALSE	EDA	147	8	KML	7156	124130	62072	3115	7.33	2.17	1.32	4.00
FALSE	EDA	147	8	NVG	1031	41788	38140	3584	2.33	1.69	1.09	9.67
FALSE	EDA	147	8	SVG	1932	53228	47782	3371	3.33	1.83	1.14	39.00
FALSE	EDA	147	16	KML	5337	92628	46357	2617	6.00	1.72	1.14	3.00
FALSE	EDA	147	16	NVG	771	31240	28515	2924	1.67	1.42	0.97	8.00
FALSE	EDA	147	16	SVG	1442	39757	35697	2782	2.33	1.50	1.02	35.67
FALSE	EDA	147	32	KML	3059	53151	26663	1837	3.33	1.32	0.99	1.00
FALSE	EDA	147	32	NVG	446	18029	16462	1976	1.33	1.12	0.85	4.33
FALSE	EDA	147	32	SVG	826	22908	20584	1912	1.67	1.12	0.84	30.33
FALSE	EDA	147	64	KML	1507	26237	13225	1140	1.67	0.99	0.79	0.00
FALSE	EDA	147	64	NVG	224	9017	8239	1205	0.67	0.87	0.72	2.00
FALSE	EDA	147	64	SVG	407	11425	10283	1197	0.67	0.86	0.71	27.67
FALSE	EDA	333	1	KML	16635	289589	145228	4769	25.33	4.42	2.14	10.00
FALSE	EDA	333	1	NVG	2385	97935	89473	6220	5.33	3.04	1.49	24.33
FALSE	EDA	333	1	SVG	4490	126682	114014	5588	7.67	3.93	1.86	59.67
FALSE	EDA	333	2	KML	16163	281579	141320	5022	25.00	4.56	2.12	10.00
FALSE	EDA	333	2	NVG	2318	95386	87163	6336	5.00	3.16	1.62	30.33
FALSE	EDA	333	2	SVG	4362	123202	110894	5797	7.33	3.80	1.82	58.00
FALSE	EDA	333	4	KML	18212	317260	159206	5873	34.33	5.26	2.61	12.67
FALSE	EDA	333	4	NVG	2610	107494	98231	7292	5.67	3.23	12.39	12.67
FALSE	EDA	333	4	SVG	4915	138941	125071	6715	8.00	4.24	2.11	62.00
FALSE	EDA	333	8	KML	16289	283988	142636	5665	25.33	4.49	2.14	9.67
FALSE	EDA	333	8	NVG	2336	96401	88114	6865	5.00	3.21	1.66	26.00
FALSE	EDA	333	8	SVG	4396	124351	111947	6394	7.00	3.88	1.97	57.67
FALSE	EDA	333	16	KML	11330	197847	99547	4477	16.00	3.58	1.97	7.00
FALSE	EDA	333	16	NVG	1627	67406	61638	5255	4.00	2.00	1.23	7.67
FALSE	EDA	333	16	SVG	3059	86624	77998	4921	5.00	2.83	1.48	47.67
FALSE	EDA	333	32	KML	6410	112005	56427	2840	9.33	2.17	1.42	3.67

Figura A.4: Tabela com os resultados (continuação).

FALSE	EDA	333	32	NVG	924	38281	35012	3240	2.33	1.65	1.03	12.67
FALSE	EDA	333	32	SVG	1731	49115	44238	3069	2.67	1.87	1.10	37.33
FALSE	EDA	333	64	KML	3379	59107	29842	1747	5.00	1.47	0.97	1.67
FALSE	EDA	333	64	NVG	491	20311	18582	1948	1.33	1.19	0.85	4.33
FALSE	EDA	333	64	SVG	913	26017	23449	1881	1.67	1.26	0.85	31.00
FALSE	SOA	5	1	KML	329	5721	2942	895	1.33	1.22	1.20	0.00
FALSE	SOA	5	1	NVG	56	2042	1862	814	1.00	1.14	1.08	0.00
FALSE	SOA	5	1	SVG	90	2471	2227	837	0.33	0.76	0.76	25.33
FALSE	SOA	5	2	KML	329	5730	2951	910	1.67	1.16	1.09	0.00
FALSE	SOA	5	2	NVG	56	2051	1871	825	1.33	1.12	1.06	0.00
FALSE	SOA	5	2	SVG	90	2470	2226	843	1.00	1.13	1.07	25.33
FALSE	SOA	5	4	KML	325	5646	2904	903	1.33	1.29	1.22	0.00
FALSE	SOA	5	4	NVG	55	2015	1837	818	1.00	1.13	1.06	0.33
FALSE	SOA	5	4	SVG	89	2440	2200	841	1.00	1.03	0.97	25.33
FALSE	SOA	5	8	KML	296	5181	2686	901	1.33	1.27	1.20	0.00
FALSE	SOA	5	8	NVG	51	1879	1715	813	1.00	1.11	1.06	0.33
FALSE	SOA	5	8	SVG	81	2253	2034	832	0.67	1.02	1.00	25.33
FALSE	SOA	5	16	KML	230	4062	2142	859	1.00	1.24	1.20	0.00
FALSE	SOA	5	16	NVG	42	1532	1402	770	0.67	0.95	0.91	0.00
FALSE	SOA	5	16	SVG	63	1810	1642	797	0.67	1.03	0.99	24.67
FALSE	SOA	5	32	KML	120	2186	1214	670	1.33	1.24	1.20	0.00
FALSE	SOA	5	32	NVG	26	941	867	623	1.00	1.10	1.06	0.00
FALSE	SOA	5	32	SVG	33	1083	998	659	0.67	1.04	1.03	24.67
FALSE	SOA	5	64	KML	70	1319	787	591	0.33	0.54	0.51	0.00
FALSE	SOA	5	64	NVG	19	666	618	559	0.67	0.94	0.90	0.00
FALSE	SOA	5	64	SVG	20	746	700	599	0.67	1.03	1.03	24.33
FALSE	SOA	9	1	KML	581	9977	5010	991	2.00	1.20	1.12	0.00
FALSE	SOA	9	1	NVG	92	3394	3086	935	1.00	1.16	1.08	1.00
FALSE	SOA	9	1	SVG	158	4184	3748	941	1.00	0.84	0.81	25.67
FALSE	SOA	9	2	KML	581	9981	5014	998	1.67	1.36	1.25	0.33
FALSE	SOA	9	2	NVG	92	3398	3090	938	2.67	0.75	3.92	1.33

Figura A.5: Tabela com os resultados (continuação).

FALSE	SOA	9	2	SVG	158	4184	3748	947	1.00	0.89	0.80	35.67
FALSE	SOA	9	4	KML	564	9701	4879	1002	1.33	1.22	1.11	0.00
FALSE	SOA	9	4	NVG	89	3312	3013	940	1.00	1.15	1.08	1.33
FALSE	SOA	9	4	SVG	153	4072	3649	946	1.00	1.03	0.86	26.00
FALSE	SOA	9	8	KML	527	9101	4600	1007	1.67	1.19	1.11	0.00
FALSE	SOA	9	8	NVG	84	3137	2856	934	1.33	1.14	1.08	0.67
FALSE	SOA	9	8	SVG	143	3826	3431	941	0.67	1.09	1.04	25.33
FALSE	SOA	9	16	KML	298	5208	2697	868	0.67	0.74	0.67	0.00
FALSE	SOA	9	16	NVG	51	1898	1734	791	0.67	0.87	0.81	0.00
FALSE	SOA	9	16	SVG	82	2287	2067	822	0.67	1.07	1.02	25.00
FALSE	SOA	9	32	KML	183	3255	1735	725	1.00	1.12	1.08	0.00
FALSE	SOA	9	32	NVG	35	1282	1176	674	1.00	1.12	1.06	0.00
FALSE	SOA	9	32	SVG	51	1516	1382	707	0.67	1.05	1.03	24.67
FALSE	SOA	9	64	KML	118	2145	1192	631	1.00	1.18	1.14	0.00
FALSE	SOA	9	64	NVG	26	929	856	595	0.67	1.11	1.08	0.00
FALSE	SOA	9	64	SVG	33	1076	992	632	0.67	1.05	1.03	24.67
FALSE	SOA	21	1	KML	1337	22854	11323	1201	2.67	1.45	1.29	1.00
FALSE	SOA	21	1	NVG	200	7559	6867	1223	1.67	1.24	1.12	3.67
FALSE	SOA	21	1	SVG	362	9495	8483	1207	1.67	1.27	1.15	27.33
FALSE	SOA	21	2	KML	1337	22865	11334	1225	2.67	1.26	1.17	0.67
FALSE	SOA	21	2	NVG	200	7571	6879	1241	1.00	1.03	0.93	4.00
FALSE	SOA	21	2	SVG	362	9498	8486	1222	1.67	1.28	1.14	27.33
FALSE	SOA	21	4	KML	1307	22399	11123	1250	2.67	1.44	1.31	0.67
FALSE	SOA	21	4	NVG	195	7443	6766	1251	1.33	1.25	1.13	3.00
FALSE	SOA	21	4	SVG	354	9296	8307	1234	1.33	1.17	1.09	27.67
FALSE	SOA	21	8	KML	1178	20207	10054	1221	2.33	1.40	1.26	0.00
FALSE	SOA	21	8	NVG	177	6755	6143	1215	1.33	1.23	1.12	1.00
FALSE	SOA	21	8	SVG	319	8410	7519	1201	1.33	1.20	1.09	26.67
FALSE	SOA	21	16	KML	781	13467	6763	1093	1.33	0.83	0.72	0.00
FALSE	SOA	21	16	NVG	120	4589	4180	1045	0.67	1.15	1.04	1.33
FALSE	SOA	21	16	SVG	212	5664	5076	1048	1.00	1.15	1.08	25.67

Figura A.6: Tabela com os resultados (continuação).

FALSE	SOA	21	32	KML	407	7082	3623	815	1.33	1.27	1.26	0.00
FALSE	SOA	21	32	NVG	67	2530	2310	779	1.00	1.14	1.08	0.33
FALSE	SOA	21	32	SVG	111	3092	2788	805	0.67	1.10	1.05	25.00
FALSE	SOA	21	64	KML	200	3554	1886	640	1.00	1.00	0.98	0.00
FALSE	SOA	21	64	NVG	37	1392	1276	617	0.67	1.10	1.06	0.00
FALSE	SOA	21	64	SVG	55	1663	1516	652	0.67	1.06	1.05	24.67
FALSE	SOA	54	1	KML	3416	58631	29049	1771	4.33	1.74	1.45	2.00
FALSE	SOA	54	1	NVG	497	19379	17631	1973	2.00	1.55	1.42	5.33
FALSE	SOA	54	1	SVG	923	24627	22031	1888	2.67	1.51	1.27	31.67
FALSE	SOA	54	2	KML	3416	58688	29106	1806	4.00	1.29	0.95	2.00
FALSE	SOA	54	2	NVG	497	19433	17685	1993	2.00	1.51	1.25	8.67
FALSE	SOA	54	2	SVG	923	24634	22038	1918	2.33	1.57	1.29	32.00
FALSE	SOA	54	4	KML	3353	57659	28624	1865	4.67	1.83	1.38	2.00
FALSE	SOA	54	4	NVG	487	19130	17414	2031	1.67	1.51	1.26	3.67
FALSE	SOA	54	4	SVG	905	24198	21650	1951	1.33	0.61	0.33	31.33
FALSE	SOA	54	8	KML	2856	49188	24469	1764	3.67	1.70	1.43	1.00
FALSE	SOA	54	8	NVG	417	16402	14938	1884	1.67	1.46	1.23	5.67
FALSE	SOA	54	8	SVG	771	20700	18530	1830	1.67	1.49	1.25	30.33
FALSE	SOA	54	16	KML	2025	34938	17433	1520	2.67	1.59	1.37	1.00
FALSE	SOA	54	16	NVG	298	11738	10697	1569	1.67	1.35	1.18	5.00
FALSE	SOA	54	16	SVG	548	14760	13224	1543	1.67	1.35	1.16	28.33
FALSE	SOA	54	32	KML	1135	19635	9850	1129	2.00	1.29	1.20	0.00
FALSE	SOA	54	32	NVG	171	6696	6106	1138	1.33	1.29	1.20	1.67
FALSE	SOA	54	32	SVG	307	8371	7512	1142	1.00	1.22	1.12	26.67
FALSE	SOA	54	64	KML	558	9725	4954	799	1.00	1.19	1.12	0.00
FALSE	SOA	54	64	NVG	88	3445	3148	795	1.00	1.16	1.14	0.00
FALSE	SOA	54	64	SVG	152	4273	3854	819	1.00	1.12	1.07	25.33
FALSE	SOA	147	1	KML	9275	160351	79898	3147	10.67	2.83	1.80	5.67
FALSE	SOA	147	1	NVG	1334	53580	48856	3906	3.67	1.84	1.23	12.33
FALSE	SOA	147	1	SVG	2504	68735	61675	3584	4.67	2.43	1.66	43.67
FALSE	SOA	147	2	KML	9275	160359	79906	3293	10.33	3.07	2.01	5.33

Figura A.7: Tabela com os resultados (continuação).

FALSE	SOA	147	2	NVG	1334	53517	48793	4012	3.67	2.03	1.29	9.00
FALSE	SOA	147	2	SVG	2504	68773	61713	3731	4.67	2.57	1.71	43.67
FALSE	SOA	147	4	KML	9065	156943	78314	3436	9.33	2.58	1.47	5.00
FALSE	SOA	147	4	NVG	1304	52598	47981	4101	4.00	2.33	1.58	15.33
FALSE	SOA	147	4	SVG	2447	67297	60397	3820	4.33	2.48	1.66	43.00
FALSE	SOA	147	8	KML	7817	135562	67771	3294	9.00	2.86	1.91	4.33
FALSE	SOA	147	8	NVG	1126	45609	41626	3827	3.00	2.15	1.53	10.67
FALSE	SOA	147	8	SVG	2110	58083	52134	3583	4.00	2.39	1.62	40.33
FALSE	SOA	147	16	KML	5607	97292	48683	2698	6.33	2.32	1.68	3.00
FALSE	SOA	147	16	NVG	810	32798	29937	3030	2.33	1.85	1.40	8.00
FALSE	SOA	147	16	SVG	1514	41747	37481	2873	3.00	1.96	1.46	35.67
FALSE	SOA	147	32	KML	3129	54363	27267	1861	4.33	1.85	1.47	1.00
FALSE	SOA	147	32	NVG	456	18435	16833	2006	1.67	1.56	1.26	4.00
FALSE	SOA	147	32	SVG	845	23426	21048	1938	1.67	1.55	1.25	30.33
FALSE	SOA	147	64	KML	1517	26406	13310	1144	2.33	1.51	1.32	0.00
FALSE	SOA	147	64	NVG	225	9074	8290	1209	1.00	1.30	1.15	2.00
FALSE	SOA	147	64	SVG	410	11498	10349	1201	1.00	1.29	1.13	27.33
FALSE	SOA	333	1	KML	20993	365421	183226	5626	37.00	5.72	2.88	13.00
FALSE	SOA	333	1	NVG	3008	123542	112866	7470	7.33	4.40	2.32	30.67
FALSE	SOA	333	1	SVG	5666	159834	143846	6653	10.00	5.03	2.50	66.67
FALSE	SOA	333	2	KML	20993	365625	183430	6030	35.00	5.64	3.07	13.00
FALSE	SOA	333	2	NVG	3008	123747	113071	7763	7.33	4.27	2.32	38.33
FALSE	SOA	333	2	SVG	5666	159921	143933	7032	9.67	5.10	2.55	67.00
FALSE	SOA	333	4	KML	20367	354729	177967	6357	31.33	4.71	1.95	13.67
FALSE	SOA	333	4	NVG	2918	120125	109767	7963	13.33	5.26	2.61	19.33
FALSE	SOA	333	4	SVG	5497	155298	139787	7289	9.00	5.16	2.53	66.67
FALSE	SOA	333	8	KML	17857	311299	156332	6053	23.33	5.07	3.11	10.33
FALSE	SOA	333	8	NVG	2560	105639	96556	7387	6.00	3.85	2.13	27.67
FALSE	SOA	333	8	SVG	4819	136282	122684	6842	8.00	4.61	2.36	60.67
FALSE	SOA	333	16	KML	12383	216179	108741	4727	15.33	4.11	2.30	7.00
FALSE	SOA	333	16	NVG	1777	73606	67303	5588	7.00	3.19	1.83	10.33

Figura A.8: Tabela com os resultados (continuação).

FALSE	SOA	333	16	SVG	3343	94634	85206	5216	6.00	3.49	1.97	51.33
FALSE	SOA	333	32	KML	6624	115743	58303	2898	8.67	2.74	1.84	3.67
FALSE	SOA	333	32	NVG	955	39546	36168	3320	2.67	2.14	1.53	12.00
FALSE	SOA	333	32	SVG	1788	50748	45707	3137	3.33	2.32	1.51	37.67
FALSE	SOA	333	64	KML	3427	59934	30257	1758	4.33	1.26	0.85	1.33
FALSE	SOA	333	64	NVG	498	20591	18837	1966	2.00	1.62	1.38	4.00
FALSE	SOA	333	64	SVG	925	26379	23774	1896	2.00	1.69	1.28	31.00
TRUE	EDA	14	1	KML	602	24039	18836	5894	1.00	1.74	0.98	1.00
TRUE	EDA	14	1	NVG	67	2939	2753	1059	0.67	0.75	0.67	0.67
TRUE	EDA	14	1	SVG	123	3874	3534	1202	0.67	0.76	0.68	25.67
TRUE	EDA	14	2	KML	504	17829	13491	4232	0.67	1.38	0.86	0.00
TRUE	EDA	14	2	NVG	61	2591	2414	989	0.67	0.89	0.67	0.33
TRUE	EDA	14	2	SVG	109	3361	3060	1097	0.33	0.51	0.47	26.00
TRUE	EDA	14	4	KML	408	12569	9075	2904	0.67	1.09	0.76	0.00
TRUE	EDA	14	4	NVG	55	2218	2056	913	0.67	0.72	0.65	0.33
TRUE	EDA	14	4	SVG	93	2820	2566	993	0.33	0.63	0.62	25.67
TRUE	EDA	14	8	KML	371	11502	8338	2806	0.67	1.07	0.78	0.00
TRUE	EDA	14	8	NVG	52	2082	1928	891	0.67	0.72	0.66	0.00
TRUE	EDA	14	8	SVG	86	2594	2361	958	0.33	0.67	0.61	25.67
TRUE	EDA	14	16	KML	292	11985	9495	3327	0.67	1.19	0.81	0.00
TRUE	EDA	14	16	NVG	40	1644	1532	831	0.67	0.79	0.66	0.00
TRUE	EDA	14	16	SVG	63	2023	1853	897	0.33	0.64	0.59	24.67
TRUE	EDA	14	32	KML	226	9514	7598	2735	0.00	0.54	0.23	0.00
TRUE	EDA	14	32	NVG	30	1271	1198	769	0.67	0.71	0.65	0.00
TRUE	EDA	14	32	SVG	44	1608	1490	862	0.33	0.63	0.59	24.67
TRUE	EDA	14	64	KML	201	9873	8176	2995	0.67	1.13	0.80	0.00
TRUE	EDA	14	64	NVG	24	1095	1044	732	0.33	0.69	0.64	0.00
TRUE	EDA	14	64	SVG	35	1417	1324	837	0.33	0.63	0.60	24.67
TRUE	EDA	18	1	KML	780	27055	20302	5957	1.67	1.86	1.00	0.33
TRUE	EDA	18	1	NVG	93	3899	3621	1135	0.67	0.76	0.68	2.00
TRUE	EDA	18	1	SVG	171	5092	4616	1270	0.67	0.79	0.70	26.33

Figura A.9: Tabela com os resultados (continuação).

TRUE	EDA	18	2	KML	737	21799	15441	4349	1.33	1.48	0.90	0.00
TRUE	EDA	18	2	NVG	93	3795	3510	1099	0.67	0.76	0.67	0.33
TRUE	EDA	18	2	SVG	168	4924	4456	1217	0.67	0.78	0.69	26.00
TRUE	EDA	18	4	KML	638	18227	12739	3630	1.33	1.32	0.86	0.00
TRUE	EDA	18	4	NVG	87	3449	3174	1047	0.67	0.75	0.67	0.33
TRUE	EDA	18	4	SVG	154	4365	3937	1111	0.33	0.66	0.62	26.00
TRUE	EDA	18	8	KML	638	18073	12585	3682	1.33	1.26	0.85	0.00
TRUE	EDA	18	8	NVG	86	3431	3161	1065	0.67	0.76	0.68	1.00
TRUE	EDA	18	8	SVG	153	4359	3935	1140	0.67	0.70	0.65	26.00
TRUE	EDA	18	16	KML	376	11634	8421	2734	0.67	1.06	0.78	0.00
TRUE	EDA	18	16	NVG	52	2102	1949	876	0.67	0.72	0.66	0.00
TRUE	EDA	18	16	SVG	87	2660	2423	956	0.33	0.65	0.59	25.33
TRUE	EDA	18	32	KML	315	10783	8099	2746	0.67	1.08	0.80	0.00
TRUE	EDA	18	32	NVG	41	1735	1621	821	0.67	0.70	0.64	0.00
TRUE	EDA	18	32	SVG	67	2216	2032	912	0.33	0.65	0.61	24.67
TRUE	EDA	18	64	KML	241	9861	7821	2768	0.00	0.52	0.24	0.00
TRUE	EDA	18	64	NVG	30	1315	1242	759	0.67	0.69	0.76	0.00
TRUE	EDA	18	64	SVG	47	1706	1582	862	0.33	0.65	0.61	24.67
TRUE	EDA	30	1	KML	1484	39025	26164	6203	3.33	3.22	1.97	2.33
TRUE	EDA	30	1	NVG	193	7761	7127	1440	0.67	0.33	0.19	1.67
TRUE	EDA	30	1	SVG	361	10014	9002	1550	1.33	0.89	0.74	28.00
TRUE	EDA	30	2	KML	1330	31873	20367	4543	1.67	1.44	0.93	0.67
TRUE	EDA	30	2	NVG	180	7142	6541	1356	0.67	0.86	0.72	1.67
TRUE	EDA	30	2	SVG	333	9116	8184	1424	1.33	0.71	0.64	28.00
TRUE	EDA	30	4	KML	1286	27543	16426	3267	1.67	1.24	0.88	0.00
TRUE	EDA	30	4	NVG	180	7055	6450	1336	1.00	0.84	0.73	1.67
TRUE	EDA	30	4	SVG	330	8951	8028	1384	1.00	0.77	0.67	27.00
TRUE	EDA	30	8	KML	1154	25401	15427	3303	1.67	1.22	0.81	0.00
TRUE	EDA	30	8	NVG	162	6360	5818	1284	0.67	0.83	0.71	1.67
TRUE	EDA	30	8	SVG	295	8031	7207	1328	0.67	0.79	0.69	27.00
TRUE	EDA	30	16	KML	840	20135	12887	3256	1.33	1.21	0.91	0.00

Figura A.10: Tabela com os resultados (continuação).

TRUE	EDA	30	16	NVG	116	4638	4261	1140	0.67	0.79	0.69	0.67
TRUE	EDA	30	16	SVG	208	5873	5292	1209	0.33	0.74	0.65	26.67
TRUE	EDA	30	32	KML	529	14835	10291	2954	0.67	1.16	0.81	0.00
TRUE	EDA	30	32	NVG	71	2904	2685	942	0.67	0.79	0.72	0.67
TRUE	EDA	30	32	SVG	124	3698	3355	1029	0.33	0.71	0.66	25.67
TRUE	EDA	30	64	KML	363	12099	8993	2907	0.67	1.10	0.79	0.00
TRUE	EDA	30	64	NVG	48	2015	1878	834	0.67	0.73	0.67	0.33
TRUE	EDA	30	64	SVG	80	2564	2346	925	0.33	0.66	0.62	25.33
TRUE	EDA	63	1	KML	3069	66289	39661	6644	3.67	2.26	1.20	2.00
TRUE	EDA	63	1	NVG	420	16752	15312	2013	1.67	1.07	0.86	5.00
TRUE	EDA	63	1	SVG	789	21572	19352	2077	1.67	1.13	0.86	31.00
TRUE	EDA	63	2	KML	2972	61671	35905	5698	3.33	1.97	1.15	2.00
TRUE	EDA	63	2	NVG	411	16398	14980	1984	1.67	1.07	0.82	7.00
TRUE	EDA	63	2	SVG	772	21057	18885	2039	1.67	1.13	0.86	30.67
TRUE	EDA	63	4	KML	2893	55265	30193	3781	3.33	1.65	1.03	1.67
TRUE	EDA	63	4	NVG	409	16246	14826	1970	1.33	0.92	0.73	8.00
TRUE	EDA	63	4	SVG	762	20663	18520	1966	1.33	1.04	0.83	31.33
TRUE	EDA	63	8	KML	2752	52468	28627	3624	3.33	1.51	1.00	1.33
TRUE	EDA	63	8	NVG	391	15578	14217	1940	1.33	1.04	0.81	5.33
TRUE	EDA	63	8	SVG	728	19721	17674	1923	1.33	0.92	0.72	30.33
TRUE	EDA	63	16	KML	2041	41551	23887	3748	2.67	1.47	0.96	1.00
TRUE	EDA	63	16	NVG	290	11552	10553	1653	0.67	0.96	0.77	4.33
TRUE	EDA	63	16	SVG	536	14623	13116	1670	1.33	0.92	0.75	28.67
TRUE	EDA	63	32	KML	1134	24310	14513	2739	1.67	1.19	0.92	0.00
TRUE	EDA	63	32	NVG	159	6420	5884	1210	0.67	0.86	0.73	1.67
TRUE	EDA	63	32	SVG	290	8149	7339	1268	0.67	0.83	0.69	26.67
TRUE	EDA	63	64	KML	704	18445	12386	3229	0.67	1.21	0.84	0.00
TRUE	EDA	63	64	NVG	97	3956	3646	980	0.67	0.77	0.77	0.00
TRUE	EDA	63	64	SVG	172	5034	4557	1063	0.67	0.72	0.65	25.67
TRUE	EDA	156	1	KML	7868	149532	81242	7884	8.33	3.16	1.59	5.33
TRUE	EDA	156	1	NVG	1105	44700	40822	3663	2.33	1.61	1.06	6.00

Figura A.11: Tabela com os resultados (continuação).

TRUE	EDA	30	16	NVG	116	4638	4261	1140	0.67	0.79	0.69	0.67
TRUE	EDA	30	16	SVG	208	5873	5292	1209	0.33	0.74	0.65	26.67
TRUE	EDA	30	32	KML	529	14835	10291	2954	0.67	1.16	0.81	0.00
TRUE	EDA	30	32	NVG	71	2904	2685	942	0.67	0.79	0.72	0.67
TRUE	EDA	30	32	SVG	124	3698	3355	1029	0.33	0.71	0.66	25.67
TRUE	EDA	30	64	KML	363	12099	8993	2907	0.67	1.10	0.79	0.00
TRUE	EDA	30	64	NVG	48	2015	1878	834	0.67	0.73	0.67	0.33
TRUE	EDA	30	64	SVG	80	2564	2346	925	0.33	0.66	0.62	25.33
TRUE	EDA	63	1	KML	3069	66289	39661	6644	3.67	2.26	1.20	2.00
TRUE	EDA	63	1	NVG	420	16752	15312	2013	1.67	1.07	0.86	5.00
TRUE	EDA	63	1	SVG	789	21572	19352	2077	1.67	1.13	0.86	31.00
TRUE	EDA	63	2	KML	2972	61671	35905	5698	3.33	1.97	1.15	2.00
TRUE	EDA	63	2	NVG	411	16398	14980	1984	1.67	1.07	0.82	7.00
TRUE	EDA	63	2	SVG	772	21057	18885	2039	1.67	1.13	0.86	30.67
TRUE	EDA	63	4	KML	2893	55265	30193	3781	3.33	1.65	1.03	1.67
TRUE	EDA	63	4	NVG	409	16246	14826	1970	1.33	0.92	0.73	8.00
TRUE	EDA	63	4	SVG	762	20663	18520	1966	1.33	1.04	0.83	31.33
TRUE	EDA	63	8	KML	2752	52468	28627	3624	3.33	1.51	1.00	1.33
TRUE	EDA	63	8	NVG	391	15578	14217	1940	1.33	1.04	0.81	5.33
TRUE	EDA	63	8	SVG	728	19721	17674	1923	1.33	0.92	0.72	30.33
TRUE	EDA	63	16	KML	2041	41551	23887	3748	2.67	1.47	0.96	1.00
TRUE	EDA	63	16	NVG	290	11552	10553	1653	0.67	0.96	0.77	4.33
TRUE	EDA	63	16	SVG	536	14623	13116	1670	1.33	0.92	0.75	28.67
TRUE	EDA	63	32	KML	1134	24310	14513	2739	1.67	1.19	0.92	0.00
TRUE	EDA	63	32	NVG	159	6420	5884	1210	0.67	0.86	0.73	1.67
TRUE	EDA	63	32	SVG	290	8149	7339	1268	0.67	0.83	0.69	26.67
TRUE	EDA	63	64	KML	704	18445	12386	3229	0.67	1.21	0.84	0.00
TRUE	EDA	63	64	NVG	97	3956	3646	980	0.67	0.77	0.77	0.00
TRUE	EDA	63	64	SVG	172	5034	4557	1063	0.67	0.72	0.65	25.67
TRUE	EDA	156	1	KML	7868	149532	81242	7884	8.33	3.16	1.59	5.33
TRUE	EDA	156	1	NVG	1105	44700	40822	3663	2.33	1.61	1.06	6.00

Figura A.12: Tabela com os resultados (continuação).

TRUE	EDA	342	16	KML	11617	207448	106631	6219	17.33	3.61	1.81	7.00
TRUE	EDA	342	16	NVG	1657	68613	62752	5369	3.33	2.50	1.54	24.67
TRUE	EDA	342	16	SVG	3119	88497	79698	5107	5.33	2.90	1.52	48.67
TRUE	EDA	342	32	KML	6789	125558	66660	5478	10.00	2.71	1.44	4.00
TRUE	EDA	342	32	NVG	967	40200	36796	3515	2.33	1.71	1.30	9.33
TRUE	EDA	342	32	SVG	1815	51741	46625	3401	2.67	1.93	1.13	38.00
TRUE	EDA	342	64	KML	3563	68588	37700	4120	5.67	1.95	1.13	2.00
TRUE	EDA	342	64	NVG	504	21060	19301	2149	1.00	0.68	0.45	6.33
TRUE	EDA	342	64	SVG	943	27094	24439	2148	1.67	1.30	0.87	31.33
TRUE	SOA	14	1	KML	869	37853	30279	9710	2.33	2.95	1.60	1.00
TRUE	SOA	14	1	NVG	80	3860	3658	1286	1.33	1.01	1.06	1.00
TRUE	SOA	14	1	SVG	159	5314	4871	1545	1.00	1.23	1.12	26.33
TRUE	SOA	14	2	KML	814	36775	29690	9137	2.33	2.99	1.71	0.67
TRUE	SOA	14	2	NVG	78	3667	3467	1248	1.00	1.19	1.07	0.67
TRUE	SOA	14	2	SVG	151	4976	4555	1471	1.33	1.21	1.11	26.00
TRUE	SOA	14	4	KML	809	36745	29704	9628	2.33	3.11	1.74	1.00
TRUE	SOA	14	4	NVG	76	3623	3428	1239	1.33	1.19	1.08	1.00
TRUE	SOA	14	4	SVG	150	4983	4565	1486	1.00	1.18	1.10	26.33
TRUE	SOA	14	8	KML	691	33551	27554	8385	1.33	2.22	1.11	0.33
TRUE	SOA	14	8	NVG	69	3199	3020	1171	1.00	1.19	1.14	0.33
TRUE	SOA	14	8	SVG	131	4306	3941	1348	1.00	1.18	1.10	25.33
TRUE	SOA	14	16	KML	538	24380	19724	6362	1.67	2.34	1.50	0.33
TRUE	SOA	14	16	NVG	51	2448	2327	1045	0.67	0.98	0.90	0.00
TRUE	SOA	14	16	SVG	96	3414	3148	1272	0.67	1.11	1.05	25.33
TRUE	SOA	14	32	KML	434	26694	22938	7549	1.67	2.61	1.61	0.00
TRUE	SOA	14	32	NVG	40	1970	1887	951	1.00	1.13	1.06	0.00
TRUE	SOA	14	32	SVG	72	2658	2462	1111	0.67	1.09	1.06	25.33
TRUE	SOA	14	64	KML	377	22630	19370	6388	1.00	1.90	1.11	0.00
TRUE	SOA	14	64	NVG	32	1677	1619	898	1.00	1.14	1.19	0.00
TRUE	SOA	14	64	SVG	59	2336	2176	1054	0.67	1.09	1.06	25.33
TRUE	SOA	18	1	KML	1086	43615	34170	10321	2.67	3.33	1.80	1.33

Figura A.13: Tabela com os resultados (continuação).

TRUE	SOA	18	1	NVG	115	5091	4761	1366	1.33	1.22	1.11	3.00
TRUE	SOA	18	1	SVG	221	6799	6179	1598	1.00	1.05	1.04	26.67
TRUE	SOA	18	2	KML	1034	38973	29990	8804	2.33	2.89	1.71	1.00
TRUE	SOA	18	2	NVG	113	4933	4605	1345	1.67	1.21	1.10	1.00
TRUE	SOA	18	2	SVG	215	6557	5955	1547	1.00	1.09	1.10	26.67
TRUE	SOA	18	4	KML	1006	38167	29423	9052	2.00	3.00	1.75	1.00
TRUE	SOA	18	4	NVG	111	4834	4508	1333	1.33	1.21	1.11	1.00
TRUE	SOA	18	4	SVG	210	6325	5737	1498	1.33	1.20	1.11	26.33
TRUE	SOA	18	8	KML	887	33892	26195	7879	2.33	2.73	1.66	0.67
TRUE	SOA	18	8	NVG	101	4345	4050	1274	1.00	1.20	1.10	1.33
TRUE	SOA	18	8	SVG	186	5629	5110	1421	0.67	1.16	1.08	26.33
TRUE	SOA	18	16	KML	658	29276	23565	7055	1.67	2.42	1.58	0.67
TRUE	SOA	18	16	NVG	67	3086	2912	1120	1.00	1.17	1.08	0.33
TRUE	SOA	18	16	SVG	124	4150	3805	1318	0.67	1.12	1.05	25.67
TRUE	SOA	18	32	KML	512	24057	19623	6174	1.67	2.29	1.54	0.00
TRUE	SOA	18	32	NVG	51	2401	2276	1001	1.00	1.19	1.09	0.33
TRUE	SOA	18	32	SVG	94	3249	2990	1174	1.00	1.13	1.07	25.33
TRUE	SOA	18	64	KML	417	23938	20333	6967	1.00	2.22	1.36	0.00
TRUE	SOA	18	64	NVG	38	1891	1813	920	1.00	1.14	1.08	0.00
TRUE	SOA	18	64	SVG	70	2621	2429	1089	0.67	1.11	1.06	25.33
TRUE	SOA	30	1	KML	1854	56725	40606	10576	4.00	4.59	2.76	4.00
TRUE	SOA	30	1	NVG	223	9289	8577	1686	2.00	0.96	0.68	3.33
TRUE	SOA	30	1	SVG	428	12180	10976	1874	1.67	1.35	1.18	28.33
TRUE	SOA	30	2	KML	1835	54152	38198	9497	3.00	3.09	1.72	1.33
TRUE	SOA	30	2	NVG	224	9281	8565	1692	1.33	1.48	1.12	2.33
TRUE	SOA	30	2	SVG	424	12060	10870	1870	1.67	1.37	1.19	28.00
TRUE	SOA	30	4	KML	1775	48724	33291	7795	3.00	2.77	1.71	1.00
TRUE	SOA	30	4	NVG	217	9029	8329	1687	1.67	1.31	1.17	2.00
TRUE	SOA	30	4	SVG	414	11757	10596	1856	1.67	1.33	1.18	28.00
TRUE	SOA	30	8	KML	1577	47694	34004	8755	2.67	2.91	1.64	1.00
TRUE	SOA	30	8	NVG	194	8049	7425	1588	1.33	1.28	1.14	2.33

Figura A.14: Tabela com os resultados (continuação).

TRUE	SOA	30	8	SVG	368	10443	9410	1731	1.67	1.27	1.14	27.33
TRUE	SOA	30	16	KML	1074	36862	27545	7537	1.67	2.11	1.10	0.67
TRUE	SOA	30	16	NVG	128	5451	5056	1337	1.00	1.19	1.08	0.67
TRUE	SOA	30	16	SVG	241	7072	6396	1472	1.33	1.21	1.12	26.33
TRUE	SOA	30	32	KML	722	27507	21258	6640	1.67	2.44	1.62	0.00
TRUE	SOA	30	32	NVG	80	3566	3334	1112	1.00	1.19	1.10	0.67
TRUE	SOA	30	32	SVG	152	4742	4317	1267	0.67	1.14	1.07	25.33
TRUE	SOA	30	64	KML	535	23843	19215	6139	1.67	2.27	1.53	0.33
TRUE	SOA	30	64	NVG	55	2569	2426	986	0.67	1.08	1.00	0.33
TRUE	SOA	30	64	SVG	102	3453	3170	1153	0.67	1.12	1.07	25.33
TRUE	SOA	63	1	KML	3934	94639	60457	11720	5.00	3.94	2.09	3.33
TRUE	SOA	63	1	NVG	522	21099	19327	2418	2.33	1.60	1.27	7.00
TRUE	SOA	63	1	SVG	988	27261	24478	2541	2.67	1.62	1.30	32.33
TRUE	SOA	63	2	KML	3919	91459	57418	10721	5.67	3.81	2.35	2.67
TRUE	SOA	63	2	NVG	521	21091	19319	2456	1.67	1.55	1.27	9.33
TRUE	SOA	63	2	SVG	987	27236	24457	2558	2.67	1.64	1.32	32.33
TRUE	SOA	63	4	KML	3783	89521	56664	11090	5.00	4.10	2.00	3.00
TRUE	SOA	63	4	NVG	502	20449	18735	2453	2.00	1.58	1.28	10.33
TRUE	SOA	63	4	SVG	956	26347	23651	2558	2.00	1.20	0.89	31.67
TRUE	SOA	63	8	KML	3325	77866	49002	9440	4.67	3.34	1.94	2.00
TRUE	SOA	63	8	NVG	448	18206	16680	2310	1.67	1.54	1.28	6.67
TRUE	SOA	63	8	SVG	844	23298	20922	2391	1.67	1.53	1.26	31.00
TRUE	SOA	63	16	KML	2432	58687	37577	7849	3.00	2.38	1.34	1.67
TRUE	SOA	63	16	NVG	325	13256	12161	1920	1.67	1.43	1.20	4.67
TRUE	SOA	63	16	SVG	607	16928	15221	2022	1.67	1.42	1.21	29.00
TRUE	SOA	63	32	KML	1352	34982	23257	5608	2.00	2.17	1.43	1.00
TRUE	SOA	63	32	NVG	174	7260	6695	1392	3.00	1.70	1.38	3.33
TRUE	SOA	63	32	SVG	323	9342	8438	1507	1.33	1.25	1.13	28.00
TRUE	SOA	63	64	KML	884	30463	22805	6262	2.00	2.34	1.56	0.33
TRUE	SOA	63	64	NVG	105	4578	4257	1146	1.00	1.21	1.11	0.33
TRUE	SOA	63	64	SVG	197	6007	5456	1289	1.00	1.16	1.09	26.33

Figura A.15: Tabela com os resultados (continuação).

TRUE	SOA	156	1	KML	9842	197203	111720	13292	11.33	5.42	2.49	7.00
TRUE	SOA	156	1	NVG	1360	55341	50593	4430	4.00	2.41	1.68	7.67
TRUE	SOA	156	1	SVG	2576	71688	64420	4379	5.00	2.73	1.80	44.67
TRUE	SOA	156	2	KML	9790	196495	111469	13375	11.00	4.97	2.07	7.00
TRUE	SOA	156	2	NVG	1358	55418	50670	4540	3.67	2.25	1.52	26.00
TRUE	SOA	156	2	SVG	2569	71425	64178	4435	4.67	2.65	1.74	44.33
TRUE	SOA	156	4	KML	9476	189218	106928	12807	10.33	5.28	2.55	6.33
TRUE	SOA	156	4	NVG	1319	53818	49199	4532	4.00	2.36	1.57	17.33
TRUE	SOA	156	4	SVG	2493	69232	62198	4426	4.67	2.63	1.72	43.33
TRUE	SOA	156	8	KML	8434	163840	90617	9971	9.67	4.21	2.30	5.00
TRUE	SOA	156	8	NVG	1177	48228	44104	4275	3.00	2.24	1.53	12.00
TRUE	SOA	156	8	SVG	2225	61860	55583	4184	4.00	2.46	1.63	41.33
TRUE	SOA	156	16	KML	5786	113724	63497	8123	6.67	3.24	1.85	3.33
TRUE	SOA	156	16	NVG	804	33019	30218	3281	2.33	1.92	1.42	3.33
TRUE	SOA	156	16	SVG	1516	42398	38125	3253	3.00	2.08	1.51	36.33
TRUE	SOA	156	32	KML	3430	70915	41148	6011	4.33	2.67	1.72	2.00
TRUE	SOA	156	32	NVG	468	19407	17799	2297	1.67	1.57	1.26	5.33
TRUE	SOA	156	32	SVG	880	24960	22481	2363	1.67	1.39	1.04	31.00
TRUE	SOA	156	64	KML	1857	47675	31565	6948	2.67	2.66	1.69	1.00
TRUE	SOA	156	64	NVG	246	10391	9567	1559	1.33	1.36	1.17	2.33
TRUE	SOA	156	64	SVG	460	13384	12089	1662	1.67	1.34	1.16	28.33
TRUE	SOA	342	1	KML	21482	399204	212670	15310	35.67	7.90	3.62	14.33
TRUE	SOA	342	1	NVG	3031	125652	114954	7951	7.00	4.33	2.37	34.67
TRUE	SOA	342	1	SVG	5727	162369	146203	7335	9.67	5.21	2.58	67.67
TRUE	SOA	342	2	KML	21485	399391	212836	15844	37.00	7.41	3.07	14.33
TRUE	SOA	342	2	NVG	3031	125075	114377	8291	7.33	4.38	2.35	40.00
TRUE	SOA	342	2	SVG	5728	162457	146287	7722	9.67	5.23	2.57	68.00
TRUE	SOA	342	4	KML	20812	381541	200832	14139	30.67	7.96	3.75	14.00
TRUE	SOA	342	4	NVG	2941	121707	111318	8389	6.33	3.84	1.77	44.67
TRUE	SOA	342	4	SVG	5556	157647	141963	7886	9.00	5.27	2.56	66.33
TRUE	SOA	342	8	KML	18131	333889	176478	13120	28.00	7.38	3.82	12.00

Figura A.16: Tabela com os resultados (continuação).

TRUE	SOA	342	8	NVG	2563	106538	97485	7789	6.00	3.83	1.93	36.67
TRUE	SOA	342	8	SVG	4841	137593	123929	7389	8.33	4.79	2.40	61.00
TRUE	SOA	342	16	KML	12770	237732	126860	11034	16.33	5.03	3.76	8.00
TRUE	SOA	342	16	NVG	1797	74793	68463	5942	4.33	3.09	1.87	25.67
TRUE	SOA	342	16	SVG	3395	96696	87116	5747	5.67	3.61	2.00	50.00
TRUE	SOA	342	32	KML	7296	145662	82314	10003	10.00	4.09	2.16	4.33
TRUE	SOA	342	32	NVG	1015	42591	39043	3841	3.00	2.28	1.64	9.67
TRUE	SOA	342	32	SVG	1919	55033	49621	3792	3.67	2.50	1.58	39.00
TRUE	SOA	342	64	KML	3798	84125	51165	8461	5.33	3.33	1.94	2.00
TRUE	SOA	342	64	NVG	522	22042	20239	2334	2.00	1.67	1.28	6.00
TRUE	SOA	342	64	SVG	983	28468	25699	2385	2.33	1.79	1.32	32.33

Figura A.17: Tabela com os resultados (continuação).