

Faculdade de Engenharia da Universidade do Porto  
Mestrado em Engenharia Informática

# **Definição da Arquitectura Empresarial**

## **O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa**



Universidade do Porto

Faculdade de Engenharia

**FEUP**

António José Tavares Gaspar

Setembro, 2009

Faculdade de Engenharia da Universidade do Porto  
Mestrado em Engenharia Informática

# **Definição da Arquitectura Empresarial**

## **O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa**



Universidade do Porto

Faculdade de Engenharia

**FEUP**

Dissertação apresentada para a obtenção do grau de  
Mestre em Engenharia Informática

Dissertação apresentada sob a orientação científica de  
Doutor João Carlos Pascoal de Faria  
Professor Auxiliar do Departamento de Engenharia Informática  
da Faculdade de Engenharia da Universidade do Porto

e co-orientação científica de  
Doutor Pedro Manuel Moreira Vaz Antunes de Sousa  
Professor Associado do Departamento de Engenharia Informática  
do Instituto Superior Técnico

António José Tavares Gaspar

Setembro, 2009

## Resumo

O trabalho desenvolvido nesta dissertação foi enquadrado no estudo de um Observatório de Mobilidade na Região Metropolitana de Lisboa. Este trabalho foi desenvolvido no âmbito de um projecto co-financiado pelo Programa Operacional Regional Lisboa e Vale do Tejo 2000-2006, o qual foi dinamizado no contexto da parceria europeia estabelecida entre Lisboa, Valência e Génova no âmbito da Operação Quadro Regional MARE, liderada pela CCDR-LVT (Comissão de Coordenação e Desenvolvimento Regional de Lisboa e Vale do Tejo). No âmbito deste estudo foi proposto um método para a criação de Observatórios de Mobilidade, o qual inclui a construção da Arquitectura Empresarial da organização. Esta dissertação teve como objectivo a elaboração desta Arquitectura Empresarial, com o objectivo de garantir o alinhamento entre o negócio e os sistemas de informação e tecnologias de suporte, de forma a minimizar os custos de investimento e exploração, garantindo a sustentabilidade do Observatório.

Para a criação da Arquitectura Empresarial foi usado o *Framework* de Zachman, o qual foi preenchido usando um método híbrido, englobando várias propostas. Uma parte da Arquitectura Empresarial, a Arquitectura Informacional, inclui a definição de uma Ontologia, a qual foca os indicadores de mobilidade, os elementos chave a serem processados pelo Observatório e que poderão ser trocados com outros Observatórios. A construção da Arquitectura Empresarial foi feita recorrendo às ferramentas Enterprise Architect (Arquitectura Empresarial) e Protégé (ontologia). A primeira para permitir a criação de um repositório documentando a Arquitectura e expondo-a à crítica e à pressão de actualização. Esta ferramenta tem também a vantagem adicional de criar um modelo integrado, baseado em UML 2.1, com algumas extensões, e de suportar MDA (Model Driven Architecture) permitindo gerar alguns artefactos das células do *framework* a partir de artefactos de outras células através de transformações, podendo no limite gerar código com base em modelos. A segunda para permitir a re-utilização de ontologias existentes e permitir a publicação da nova ontologia num repositório de ontologias.

Ao longo deste documento é feito um enquadramento teórico, é definido o contexto do problema, é apresentado o método para a criação da Arquitectura Empresarial, é desenvolvido o caso de estudo recorrendo ao método proposto e, finalmente, são extraídas conclusões e identificadas linhas de trabalho futuro.

A solução adoptada, quer em termos ferramentais, quer em termos metodológicos, verificou-se adequada aos objectivos, permitindo modelizar de forma integrada, uma Arquitectura Empresarial, com base no *Framework* de Zachman, garantindo o alinhamento entre negócio e sistemas de informação e tecnologias de suporte associadas. Foram sentidas apenas limitações na produção de matrizes de CRUD e na interoperabilidade entre as ferramentas de suporte às ontologias e à arquitectura empresarial. A integração com as operações da organização verificou-se adequada ao permitir a exportação para documentação via RTF, para intranet via HTML e por suportar MDA, permitindo a geração de código a partir dos artefactos e assegurando uma reconfiguração ágil do mesmo, caso haja alteração do negócio.

Palavras-chave: Arquitectura Empresarial, Framework de Zachman, Ontologia, UML, MDA, Observatório de Mobilidade

## Abstract

This work was developed under a study for a Mobility Observatory in the Lisbon Metropolitan Region. This study was carried out in a project co-funded by the Lisboa e Vale do Tejo 2000-2006 Regional Operational Programme, which was part of the European partnership among Lisbon, Valencia and Genoa, under MARE Regional Framework Operation, led by CCDR-LVT (Lisbon and Tagus Valley Coordination and Regional Development Commission). A method for the creation of mobility observatories, which includes the construction of an Enterprise Architecture, was proposed in this study. The goal of this dissertation was the elaboration of this Enterprise Architecture, to guarantee the alignment between business and information systems and associated supporting technologies, in order to minimize investment and exploitation costs, ensuring the sustainability of the observatory.

The Zachman Framework was used to create the Enterprise Architecture and their cells were filled using a hybrid approach combining several proposals. A part of the Enterprise Architecture, the Informational Architecture, includes the definition of an ontology, focusing mobility indicators, the key elements that will be processed by the Observatory and that might be exchanged with other observatories.

The Enterprise Architecture was built using the tools Enterprise Architect (Enterprise Architecture) and Protégé (ontology). The first was used to create a repository documenting the Architecture, exposing it to criticism and to update pressures. This tool has also the added advantage of creating an integrated model, based on UML 2.1, with some extensions, and of supporting MDA (Model Driven Architecture), enabling the creation of some framework cell artifacts from other framework cell artifacts through transformations, which, finally, may generate code based on models. The second tool was used to enable the reuse of existing ontologies and to publish the new ontology in an ontology repository.

In this document the theoretical framework is presented, as well as the problem context, a method for Enterprise Architecture creation, a case study using the proposed method and conclusions and future work lines.

The adopted solution, either toolwise or methodwise, demonstrated to be well adapted to the objectives, enabling the integrated modeling of an Enterprise Architecture using the Zachman Framework, ensuring the alignment between business and information systems and supporting technologies. The only limitations felt were at the production of CRUD matrices and in the interoperability between ontology and enterprise architecture tools. The integration with organization operations showed adequate since it enables to export to documents via RTF, to export to intranet via HTML and due to its MDA support, which enables to produce code from the artifacts insuring an agile code reconfiguration in case of business change.

Keywords: Enterprise Architecture, Zachman Framework, Ontology, UML, MDA, Mobility Observatory

## **Dedicatória**

À minha esposa  
Odett

Às minhas filhas  
Diana e Helena

## Agradecimentos

Ao Professor Doutor João Carlos Pascoal de Faria  
e ao  
Professor Doutor Pedro Manuel Moreira Vaz Antunes de Sousa  
pela orientação e disponibilidade para este trabalho de investigação

À equipa do estudo do Observatório de Mobilidade na Região Metropolitana de Lisboa  
Engenheiro Robert Stüssi  
Mestre Helder Ribeiro Dias Cristovão  
Dr. Vitor Pimentel Nogueira  
Dr.<sup>a</sup> Alexandra Almeida  
Arq.<sup>a</sup> Isabel Carvalho Seabra  
Dr. Manuel João Martins  
pelos contributos para a definição do problema

Aos meus colegas nesse estudo  
Professor Doutor António Manuel Pérez da Silva Babo  
Engenheiro Rui Sanches da Costa Barros  
Mestre Marco Amaro Oliveira  
pelos contributos técnicos para a definição do problema

À Comissão de Coordenação e Desenvolvimento Regional de Lisboa e Vale do Tejo  
na pessoa da sua Vice-Presidente Dr.<sup>a</sup> Paula Cristina Cunha  
pelo convite para a realização do estudo, o qual veio inspirar esta dissertação

## Abreviaturas

BSC	Balanced Scored Cards
CIM	Computation Independent Model
CMS	Content Management Server
CRUD	Create-Read-Update-Delete
CSV	Comma Separated Values
DAML	DARPA Agent Markup Language
EAP	Enterprise Architecture Planning
GIS	Geographical Information System
HTML	HyperText Markup Language
IDE	Integrated Development Environment
CCDRLVT	Comissão de Coordenação e Desenvolvimento Regional de Lisboa e Vale do Tejo
MDA	Model Driven Architecture
MDD	Model Driven Development
ODM	Ontology Definition Metamodel
OIL	Ontology Interchange Language ou Ontology Inference Layer
OLS	Ontology Lookup Service
OMG	Object Management Group
OWL	Web Ontology Language
PDCA	Plan-Do-Check-Act
PIM	Platform Independent Model
POSI	Pós-Graduação em Sistemas de Informação
PSM	Platform Specific Model
RDF	Resource Description Framework
RDFS	RDF Schema
RTF	Rich Text Format
SHOE	Simple HTML Ontology Extensions
SIM	Society for Information Management
SPSS	Statistical Package for the Social Sciences
UML	Unified Modeling Language
XMI	XML Metadata Interchange

## Lista de Figuras

Figura 2.1 - Modelo de Alinhamento Estratégico.....	5
Figura 2.2 – Sub-Arquitecturas de uma Arquitectura Empresarial e alinhamentos associados.....	7
Figura 2.3 - <i>Framework</i> de Zachman (versão 2003). ....	10
Figura 2.4 - Método Enterprise Architecture Planning.....	11
Figura 3.1 - Ciclo de Deming. ....	20
Figura 3.2 - Fases do método proposto de criação de um Observatório de Mobilidade.....	25
Figura 3.3 - Mapeamento entre sub-arquitecturas de uma Arquitectura Empresarial e <i>Framework</i> de Zachman. ....	30
Figura 4.1 - Estruturação inicial da Arquitectura Empresarial. ....	32
Figura 4.2 - Célula A – Coisas importantes para a organização.....	33
Figura 4.3 – Conteúdo do pacote Produtos e Serviços.....	34
Figura 4.4 - Diagrama de casos de uso primários.....	35
Figura 4.5 - Os processos de negócio primários realizam os casos de uso primários. ....	36
Figura 4.6 - Business Locations Diagram – Localização do Observatório. ....	37
Figura 4.7 - Actores relevantes no contexto do Observatório.....	37
Figura 4.8 - Missão, Visão, Objectos Estratégicos e Estratégias.....	39
Figura 4.9 - Relação entre objectivos, estratégias e requisitos.....	40
Figura 4.10 - Relação entre objectivos indirectos, estratégias e requisitos. ....	41
Figura 4.11 - Modelo semântico. ....	42
Figura 4.12 - Vista geral da ontologia de indicadores.....	46
Figura 4.13 - Ontologia de indicadores com sub-classes e atributos.....	47
Figura 4.14 - Formulário criado para carregamento dos dados associados ao indicador População Residente.....	47
Figura 4.15 - Passo 1: Exportação do Protégé para Enterprise Architect usando UML 1.4.....	48
Figura 4.16 - Aspecto geral da ontologia de indicadores após processamento manual.....	49
Figura 4.17 - Indicadores com a temática Transportes e Energia. ....	50
Figura 4.18 - Diagrama de processos de negócio primários.....	50
Figura 4.19 - Matriz CRUD – Entidades Informacionais vs Processos de Negócio.....	53
Figura 4.20 - Matriz CRUD – Entidades Informacionais vs Processos de Negócio (após manipulação). ....	54
Figura 4.21 - Vista parcial do modelo lógico de dados. ....	55
Figura 4.22 - Logística do Observatório.....	56
Figura 4.23 - Cronograma principal. ....	57

Figura 4.24 - Mapa de estratégia. ....	58
Figura 4.25 - Sistemas, entidades informacionais criadas e fluxos informacionais. ....	59
Figura 4.26 - Matriz Sistemas versus Processos de Negócio. ....	60
Figura 4.27 - Matriz de relacionamento entre sistemas e entidades informacionais (classes). ....	60
Figura 4.28 - Matriz de relacionamento entre sistemas e entidades informacionais (actores). ....	61
Figura 4.29 – Vista parcial do modelo físico de dados. ....	62
Figura 4.30 - Vista geral da célula J. ....	63
Figura 4.31 - Organigrama do Observatório. ....	64
Figura 4.32 - Matriz processos de negócio versus organização. ....	64
Figura 4.33 - Arquitectura distribuída do sistema. ....	65
Figura 4.34 - Diagrama de componentes com os pacotes aplicativos que implementam os sistemas. ....	67
Figura 4.35 – Diagrama de instalação com número de equipamentos e tipo. ....	68
Figura 4.36 - Sub-Arquitecturas da Arquitectura Empresarial e respectivos artefactos. ....	69
Figura 4.37 - Relatório HTML – Vista Arquitectura Empresarial. ....	70
Figura 4.38 - Relatório HTML – Vista <i>Framework</i> de Zachman. ....	70
Figura 4.39 - Relatório HTML – Pormenor de navegação na Arquitectura Informacional. ....	71
Figura A.1 - Ontologia Wines carregada no OWL Editor. ....	81
Figura A.2 - Resultado da exportação para Enterprise Architect a partir do OWL Editor. ....	81
Figura A.3 - NeOn Toolkit mostrando ontologia Wines, classe Port e respectivas propriedades e instâncias. ....	82
Figura A.4 - NeOn Toolkit mostrando ontologia Wines após conversão de OWL para F-Logic, com visualização de parte da ontologia (vinhos de sobremesa). ....	83
Figura A.5 - Parte da ontologia Wines num diagrama de classes após importação para Enterprise Architect. ....	84
Figura A.6 - Ontologia Wines visualizada no Protégé 3.4.1. ....	85
Figura A.7 - Parte do resultado da exportação em UML 1.4 a partir do Protégé. ....	85
Figura A.8 - Resultado parcial da exportação via NeOn, usando RDFS e UML. ....	86
Figura A.9 - Resultado parcial da exportação via NeOn, usando o formato TURTLE. ....	87

## Lista de Tabelas

Tabela 2.1 - Sequência de preenchimento proposta por Pereira & Sousa (2004).....	12
Tabela 2.2 - Artefactos propostos para cada célula da Framework de Zachman.....	12
Tabela 2.3 - Modelos usados no Enterprise Architect para o <i>add-in</i> com o <i>Framework</i> de Zachman.....	14
Tabela 3.1 - Sequência de preenchimento estendida cobrindo a Arquitectura Tecnológica.....	28
Tabela 3.2 - Sequência de preenchimento estendida, com as células relevantes para este trabalho.....	29
Tabela 3.3 – Artefactos usados na modelação da Arquitectura Empresarial do Observatório.....	29
Tabela A.1 - Síntese dos testes de importação da ontologia para Enterprise Architect.....	88

# Índice

Resumo.....	iii
Abstract.....	iv
Dedicatória.....	v
Agradecimentos.....	vi
Abreviaturas.....	vii
Lista de Figuras.....	viii
Lista de Tabelas.....	x
Índice.....	xi
1 Introdução.....	1
2 Estado da Arte.....	5
2.1 O alinhamento entre o Negócio e as Tecnologias de Informação.....	5
2.2 Arquitectura Empresarial.....	6
2.2.1 Os alinhamentos e a sua avaliação.....	8
2.2.2 A representação de uma Arquitectura Empresarial.....	9
2.2.3 A construção de uma Arquitectura Empresarial.....	11
2.2.4 Ferramentas de suporte a Arquitecturas Empresariais.....	14
2.3 Ontologias.....	15
2.3.1 Ferramentas de suporte a Ontologias.....	17
3 Método proposto para a construção de uma Arquitectura Empresarial.....	20
3.1 Contexto do problema abordado.....	20
3.1.1 Mobilidade sustentável.....	20
3.1.2 Observatórios de Mobilidade.....	20
3.1.3 Âmbito de um Observatório de Mobilidade.....	21
3.1.4 Requisitos gerais de um Observatório de Mobilidade.....	22
3.1.5 Os Indicadores de um Observatório de Mobilidade.....	23
3.1.6 Uma rede de Observatórios.....	24
3.2 A criação de um Observatório de Mobilidade.....	24
3.2.1 Levantamento da situação actual.....	25
3.2.2 Posicionamento estratégico.....	25
3.2.3 Arquitectura empresarial.....	25
3.2.4 Disseminação.....	27
3.2.5 Implementação.....	27
3.3 Método para a construção da Arquitectura Empresarial.....	27
4 Caso de Estudo: Arquitectura Empresarial do Observatório de Mobilidade.....	31
4.1 Introdução.....	31
4.2 Processo de construção.....	31
4.3 Fase 1 - Células A, B, C, D, E e F.....	32
4.3.1 Célula A – Lista de coisas importantes para a organização.....	32
4.3.2 Célula B – Lista de processos realizados pela organização.....	34
4.3.3 Célula C – Lista de locais onde a organização opera.....	35
4.3.4 Célula D - Lista de organizações relevantes para o Observatório.....	37
4.3.5 Célula E – Lista de eventos relevantes para a organização.....	38
4.3.6 Célula F – Lista de objectivos e estratégias da organização.....	38
4.4 Fase 2 – Célula G.....	40
4.4.1 Célula G – Modelo semântico.....	40
4.5 Fase 3 - Células H, M.....	50
4.5.1 Célula H – Modelo de processos da organização.....	50

4.5.2	Célula M – Modelo lógico de dados.....	55
4.6	Fase 4 (células I, K, L, N, S) .....	56
4.6.1	Célula I – Sistema logístico da organização .....	56
4.6.2	Célula K – Cronograma principal.....	56
4.6.3	Célula L – Plano de negócio.....	57
4.6.4	Célula N – Arquitectura de aplicações.....	58
4.6.5	Célula S – Modelo físico de dados .....	61
4.7	Fase 5 (células J, O, T).....	62
4.7.1	Célula J – Modelo de Workflow .....	62
4.7.2	Célula O – Arquitectura distribuída do sistema.....	63
4.7.3	Célula T – Projecto de sistema.....	65
4.8	Fase 6 (célula U) .....	67
4.8.1	Célula U – Arquitectura da tecnologia.....	67
4.9	Sub-Arquitecturas da Arquitectura Empresarial.....	68
5	Conclusões e trabalho futuro .....	72
6	Referências.....	75
Anexo A	Importação de ontologias .....	80
Anexo B	Script da tabela indicador.....	89

# 1 Introdução

## 1.1 Motivação

A ideia para a elaboração desta dissertação de Mestrado surgiu do interesse em aprofundar o tema do alinhamento dos Sistemas de Informação com os Processos de Negócio numa organização. Este interesse foi despoletado durante a realização do POSI (Pós-Graduação em Sistemas de Informação) do Instituto Superior Técnico, em 2005 e 2006, particularmente na cadeira de Sistemas de Informação Empresariais.

A oportunidade para concretizar esta ideia surgiu no âmbito do estudo de um Observatório de Mobilidade na Região Metropolitana de Lisboa. Este trabalho foi desenvolvido num projecto co-financiado pelo Programa Operacional Regional Lisboa e Vale do Tejo 2000-2006, o qual foi dinamizado no contexto da parceria europeia estabelecida entre Lisboa, Valência e Génova no âmbito da Operação Quadro Regional MARE, liderada pela CCDR-LVT (Comissão de Coordenação e Desenvolvimento Regional de Lisboa e Vale do Tejo).

## 1.2 Objectivos

Um Observatório de Mobilidade é um instrumento de monitorização das políticas de mobilidade. O seu funcionamento implica tipicamente a recolha de um conjunto de dados sobre mobilidade junto de um conjunto de parceiros públicos e privados. Implica também o processamento destes dados, de forma a gerar um conjunto de indicadores, os quais constituirão o painel de controlo do grau de implementação das políticas de mobilidade.

Um requisito fundamental na implementação de um Observatório é a garantia da sua sustentabilidade, pois deverá operar durante um período alargado de tempo para garantir que as séries temporais de indicadores captam os resultados das transformações induzidas no território. Uma forma de garantir esta sustentabilidade é garantir que o número de indicadores a processar é minimizado e que o seu processamento é automatizado. Esta minimização deve ser feita sem colocar em causa os objectivos do Observatório. Por outro lado é necessário garantir que a automatização é adequada aos objectivos do Observatório. Ou seja, é necessário garantir que existe um alinhamento entre os sistemas, a tecnologia que os suporta e o negócio da organização. Este é o problema que nos propomos abordar: otimizar os sistemas e a tecnologia, face aos objectivos do negócio.

## 1.3 Contribuições

Para contribuir para a sustentabilidade do Observatório, foi proposta uma metodologia para a sua criação, que incorpora o planeamento de um sistema de informação para esta nova organização, garantindo que este responde às exigências do negócio, quer numa fase inicial, quer durante o seu ciclo de vida.

Para tal propõe-se a construção de uma Arquitectura Empresarial, como instrumento de planeamento e de gestão de arquitecturas de sistemas de informação, que garante o alinhamento entre os sistemas, a tecnologia e o negócio. Este será o objectivo desta tese: a construção de uma Arquitectura Empresarial.

Dado que o Observatório de Mobilidade na Região Metropolitana de Lisboa poderá operar inserido numa rede de Observatórios, os quais irão intercambiar indicadores entre si, originando problemas de interoperabilidade semântica, recorrer-se-á ao uso de uma ontologia de indicadores para resolver esta questão, estruturando o conhecimento deste domínio. A existência desta ontologia permitirá aos diversos Observatórios saber que indicadores existem e os conteúdos associados.

A adaptabilidade da solução ao longo da sua vida será garantida pela publicação e exposição da Arquitectura Empresarial, permitindo que esta seja continuamente confrontada com a realidade, originando conseqüentemente sugestões de adaptação, de forma a garantir a sincronização entre o modelo e a realidade.

A construção da Arquitectura Empresarial será suportada por uma ferramenta, a qual permitirá:

- Suportar um *framework* de Arquitectura Empresarial;
- Validar os modelos do *framework* e elementos destes, particularmente por suportar intrinsecamente as linguagens e notações associadas;
- Permitir a publicação, constituindo-se num repositório “público”, expondo a Arquitectura Empresarial à crítica e à sua constante actualização, à medida que o negócio se adapta à envolvente externa;
- Suportar a geração de código a partir de modelos, aumentando a utilidade da Arquitectura Empresarial ao integrá-la no processo de produção de software e garantindo o alinhamento entre os modelos e o código associado;
- Tornar-se numa base de conhecimento sobre a organização.

A construção da ontologia de indicadores será também suportada por uma ferramenta específica, a qual permitirá a integração desta ontologia na Arquitectura Empresarial. O facto da ferramenta suportar linguagens de ontologias, permitirá interagir com repositórios, importando ou exportando conhecimento, na forma de ontologias.

O método usado para construir a Arquitectura Empresarial é uma adaptação e integração de alguns métodos existentes.

## **1.4 Estrutura da Dissertação**

Esta dissertação está estruturada em diversos capítulos, os quais são seguidamente descritos.

### **Capítulo 1 – Introdução**

Este capítulo, onde são apresentados o tema desta dissertação, a motivação, a oportunidade, o problema que se pretende resolver e os objectivos da dissertação. É também referida a estrutura do documento.

### **Capítulo 2 – Estado da Arte**

Neste capítulo é descrito o estado da arte dos temas relevantes para esta dissertação. Para tal apresenta secções específicas dedicadas à Arquitectura Empresarial e a Ontologias. No

primeiro caso são focados os seguintes tópicos: apresentação do conceito de Arquitectura Empresarial; descrição das diversas arquitecturas componentes de uma Arquitectura Empresarial; apresentação dos problemas do desalinhamento entre as várias arquitecturas componentes; descrição de heurísticas para medir o alinhamento entre arquitecturas componentes; como representar uma Arquitectura Empresarial, introduzindo o conceito de *framework* e apresentando o *Framework* de Zachman; como construir uma Arquitectura Empresarial, indicando vários métodos; e, finalmente, as vantagens da utilização de ferramentas de suporte a Arquitecturas Empresariais e a identificação de algumas.

A secção sobre Ontologias aborda os seguintes tópicos: apresentação do conceito de Ontologia e as suas aplicações; a classificação de Ontologias; as componentes de uma Ontologia; método para a construção de uma Ontologia; o ciclo de vida de uma Ontologia; as linguagens de Ontologias; os motores de busca específicos para Ontologias; e, finalmente, apresentação de algumas ferramentas de suporte de Ontologias.

### Capítulo 3 – Método proposto

Neste capítulo é apresentado o contexto e o método proposto para realizar o caso de estudo.

Inicialmente é apresentado o contexto do problema abordado, ou seja, o facto da mobilidade actual ser insustentável, devido à limitação dos recursos fósseis e ao aquecimento global por eles induzido. A evolução para uma mobilidade sustentável implica a alteração das políticas de mobilidade, cujos efeitos devem ser medidos ao longo do tempo para avaliar o seu sucesso. Tal é feito recorrendo a planos de mobilidade e a Observatórios de Mobilidade, os quais usam uma bateria de indicadores como painel de controlo.

Seguidamente são apresentados o âmbito e os requisitos gerais de Observatórios deste tipo, sendo dado um ênfase particular à sustentabilidade do próprio Observatório, a qual deverá ser garantida através da minimização do número de indicadores e de uma automatização alinhada com os objectivos do Observatório. São também apresentados os atributos dos indicadores de mobilidade e os requisitos para a criação de um Observatório deste tipo.

Com vista a satisfazer estes requisitos para a criação de um Observatório é apresentado um método para este efeito. Deste método faz parte integral a criação de uma Arquitectura Empresarial, durante a especificação do Observatório.

Segue-se a descrição do método usado para a criação da Arquitectura Empresarial. Na secção referente à Arquitectura Informacional é apresentada a abordagem para a criação da ontologia de indicadores. A Arquitectura Empresarial é criada usando como referencial o *Framework* de Zachman. Basicamente o método proposto é uma integração de várias propostas, particularmente no que toca aos modelos usados em cada célula, bem como à sequência de preenchimento das mesmas. Nem todas as células são preenchidas, pois algumas são irrelevantes para a vista que se pretende neste trabalho e noutros casos não existe informação suficiente que permite modelizar.

A criação da Arquitectura Empresarial é realizada sobre a ferramenta Enterprise Architect da empresa Sparx Systems, a qual assegura a integridade do modelo, dos diversos artefactos e dos elementos constituintes. Esta ferramenta suporta também a exportação para documentos via RTF (Rich Text Format), permitindo a produção e actualização expedita da documentação necessária, mas também permite a publicação em HTML (HyperText

Markup Language), permitindo a exposição “pública” do modelo da Arquitectura Empresarial, potenciando a sincronização entre o modelo e a realidade.

#### **Capítulo 4 – Caso de estudo**

Neste capítulo é apresentada de forma detalhada a construção da Arquitectura Empresarial seguindo o método proposto, aplicado ao caso de estudo seleccionado: a criação do Observatório de Mobilidade na Região Metropolitana de Lisboa.

Para cada célula do *Framework* de Zachman são apresentados e discutidos os artefactos seleccionados. Para ilustrar a explicação são apresentadas algumas figuras, com captura de écran das ferramentas usadas.

Após a criação da Arquitectura Empresarial, os artefactos considerados relevantes são organizados numa nova vista, estruturada segundo as Arquitecturas componentes da Arquitectura Empresarial. Esta vista destina-se a facilitar a geração de documentação, alinhada com a proposta geral de método para a criação de um Observatório de Mobilidade.

Finalmente é gerado um relatório da Arquitectura Empresarial em HTML, permitindo a sua consulta, por exemplo, numa Intranet, usando um simples *browser*. Esta publicação tem o objectivo estratégico, já referido, de expor o modelo, potenciando a sua sincronização com a realidade. Desta forma pretende evitar-se o clássico arquivamento das versões em papel, em pastas normalmente localizadas em locais inacessíveis à generalidade dos colaboradores de uma organização.

#### **Capítulo 5 – Conclusões e trabalho futuro**

As principais conclusões obtidas durante a elaboração deste trabalho são apresentadas neste capítulo. São também identificadas linhas de trabalho futuro, que poderão dar origem a trabalhos de investigação com níveis diferenciados de profundidade, tirando partido do conhecimento obtido na elaboração desta dissertação.

#### **Anexos**

A dissertação é completada por alguns anexos ilustrando alguns tópicos em detalhe, como testes de exportação de ontologias e *scripts* criados automaticamente para geração de tabelas de bases de dados.

## 2 Estado da Arte

### 2.1 O alinhamento entre o Negócio e as Tecnologias de Informação

O investimento em tecnologia não garante automaticamente o seu retorno. Muitas organizações têm sérios problemas para rentabilizar as suas opções tecnológicas, particularmente devido aos seus sistemas de informação não serem adequados ao seu negócio, quer devido a erro de projecto, quer devido à dinâmica do próprio negócio.

Em 1993 Henderson e Venkatraman abordaram este problema de optimização, com a apresentação do seu Modelo de Alinhamento Estratégico (Henderson & Venkatraman, 1999). Neste artigo afirmavam que a dificuldade em obter retorno dos investimentos em tecnologias de informação se devia, em parte, à falta de alinhamento entre as estratégias de negócio e de tecnologias de informação da organização. Afirmavam também que o alinhamento é um processo dinâmico, associado à própria dinâmica do negócio, obrigando a que as tecnologias de informação se ajustem de forma a garantir este alinhamento ao longo do tempo.

Na Figura 2.1 podemos observar o modelo proposto por Henderson e Venkatraman.

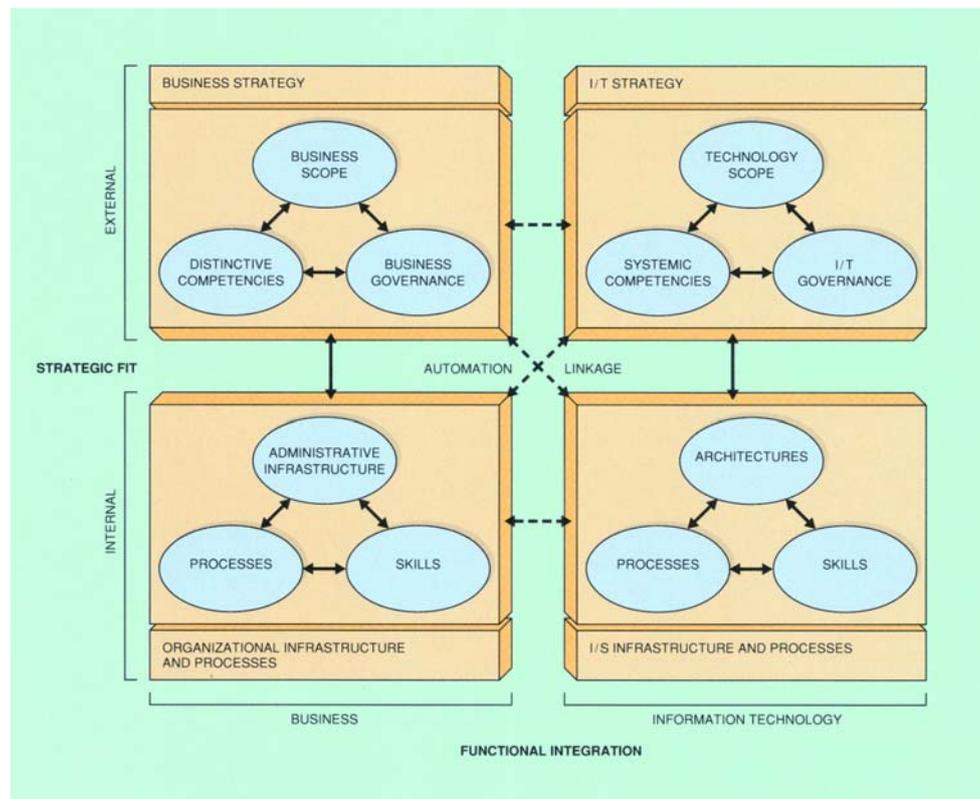


Figura 2.1 - Modelo de Alinhamento Estratégico  
Fonte: (Henderson & Venkatraman, 1999)

Como se pode observar este modelo apresenta quatro domínios fundamentais:

- Estratégia de negócio;
- Estratégia de tecnologias de informação;
- Infraestrutura organizacional e processos;
- Infraestrutura de tecnologia de informação e processos.

Entre os vários domínios existem diversos alinhamentos a garantir. No entanto salientam-se dois, que os autores denominaram de integração estratégica e integração funcional.

O primeiro corresponde à articulação entre a estratégia do negócio e a estratégia das tecnologias de informação face aos desafios externos. Este alinhamento reflecte a capacidade das tecnologias de informação de suportar e re-inventar o negócio, permitindo aumentar a competitividade da empresa e criar os chamados sistemas de informação estratégicos (Laudon & Laudon, 2004; Ward & Peppard, 2002).

O segundo corresponde à exigência de coerência interna entre os requisitos e expectativas da organização e a capacidade de resposta da função tecnologias de informação da organização.

Os benefícios do alinhamento entre os sistemas de informação e o negócio têm sido referidos na literatura (Chan & Reich, 2007). De salientar que esta questão não é abordada apenas por académicos, tendo-se tornado uma preocupação dos responsáveis operacionais, facto bem patente no inquérito lançado anualmente pela SIM (Society for Information Management) intitulado *Key Issues for IT Executives*. Por exemplo, em 2003 e 2004 esta questão ocupava o primeiro lugar na lista das preocupações dos executivos (Luftman, 2005).

Outros modelos têm sido propostos, tentando representar a empresa nesta óptica, variando nas diversas componentes e nas dimensões do alinhamento (Chan & Reich, 2007; Chan & Reich, 2007). No entanto, há mesmo investigadores que colocam em causa o próprio conceito. É o caso de Claudio Ciborra que considera que a investigação sobre este tema seguiu um caminho errado ao fechar as componentes estratégia (do negócio) e tecnologias de informação em duas caixas negras e ao preocupar-se com a linha que as une e não com o seu interior, assumindo-as como factos consumados. Segundo ele estes modelos não são adequados, o que a investigação demonstra, particularmente em grandes empresas, onde não existe liderança na implementação e uso de tecnologias de informação e a tecnologia está a vaguear, como se estivesse fora de controlo (Ciborra, 2002).

Com efeito, para podermos identificar problemas de alinhamento teremos de modelizar a empresa mais em detalhe, particularmente ao nível das tecnologias de informação. Uma forma de o fazermos é capturando a Arquitectura Empresarial da empresa.

## **2.2 Arquitectura Empresarial**

Não existe consenso sobre uma definição de Arquitectura Empresarial. A definição de arquitectura da norma IEEE 1471:2000 (Ieee, 2000), adoptada pela ISO/IEC como ISO/IEC 42010:2007 (Wikipedia/IEEE1471, 2009), é uma referência incontornável (Microsoft, 2009):

*The fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution.*

Na realidade coexistem várias definições para Arquitectura Empresarial (McIlree, 2006). Por exemplo, a definição proposta pelo Center for Information Systems Research (CISR) do MIT Sloan School of Management é a seguinte (Ross, 2007):

*The organizing logic for business process and IT infrastructure capabilities reflecting the integration and standardization requirements of the firm's operating model.*

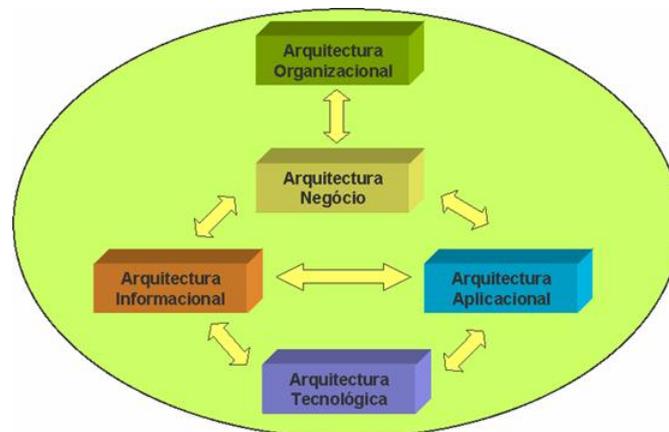
A Microsoft tem uma definição muito pragmática (Microsoft, 2009):

*An enterprise architecture (EA) is a conceptual tool that assists organizations with the understanding of their own structure and the way they work. It provides a map of the enterprise and is a route planner for business and technology change.*

*Normally an enterprise architecture takes the form of a comprehensive set of cohesive models that describe the structure and the functions of an enterprise. Important uses of it are in systematic IT planning and architecting, and in enhanced decision making.*

Na prática tal traduz-se num conjunto de modelos segundo várias perspectivas que retratam a empresa. Estes modelos podem estruturar-se segundo várias sub-arquitecturas: a Arquitectura Organizacional, a Arquitectura de Negócio, a Arquitectura Informacional, a Arquitectura Aplicacional e a Arquitectura Tecnológica (Mitre, 2004; Sousa *et al.*, 2007).

Na Figura 2.2 pode ser observada uma representação das várias sub-arquitecturas de uma Arquitectura Empresarial, bem como dos vários tipos de alinhamento existentes entre elas.



**Figura 2.2 – Sub-Arquitecturas de uma Arquitectura Empresarial e alinhamentos associados.**

Com base em Sousa *et al.* (2007) explicita-se de seguida o âmbito de cada sub-arquitectura.

- Arquitectura Organizacional - Esta arquitectura inclui os artefactos associados à missão, visão, estratégia e estrutura da organização. Inclui também políticas organizacionais e de recursos humanos.
- Arquitectura de Negócio - A Arquitectura de Negócio retrata os processos de negócio da organização.
- Arquitectura Informacional - Esta arquitectura contém as entidades informacionais (os dados) necessárias aos processos de negócio da organização. Trata-se de uma

vista de alto nível da informação-chave usada no negócio, de forma independente dos sistemas e da tecnologia.

- **Arquitectura Aplicacional** - A Arquitectura Aplicacional corresponde às aplicações que irão automatizar os processos de negócio e manipular as entidades informacionais a eles associadas. Para que cumpra os seus objectivos deverá ser obtida na sequência da definição da Arquitectura de Negócio, seguida da definição da Arquitectura Informacional.
- **Arquitectura Tecnológica** - Esta arquitectura inclui as tecnologias que implementam as aplicações e a infra-estrutura tecnológica complementar, como as redes e os sistemas de gestão de bases de dados.

### 2.2.1 Os alinhamentos e a sua avaliação

Foi anteriormente referida a necessidade de garantir o alinhamento entre o negócio e a tecnologia para garantir o retorno do investimento nesta última, assegurando que cumpre os objectivos do negócio. Numa Arquitectura Empresarial existem alinhamentos entre as várias arquitecturas componentes e dentro das mesmas. O estudo deste problema tem sido focado no triângulo Arquitectura de Negócio, Arquitectura Informacional e Arquitectura Aplicacional.

Os problemas do desalinhamento podem ser resumidos de uma forma sucinta (Pereira & Sousa, 2003) e ilustrados com alguns exemplos práticos (Sousa *et al.*, 2004).

- **Arquitectura de Negócio versus Arquitectura Informacional** - Quando o negócio exige informação que não existe no sistema de informação ou quando este último processa informação que não é necessária para o negócio. Um exemplo clássico é o pedido de um relatório específico, relevante para o negócio, mas que o sistema de informação não disponibiliza, obrigando a trabalho manual, tipicamente usando folhas de cálculo.
- **Arquitectura de Negócio versus Arquitectura Aplicacional** - Quando os utilizadores das aplicações executam trabalho adicional ao definido nos processos de negócio. Um exemplo é a impressão individual de facturas, porque a aplicação não suporta impressão múltipla.
- **Arquitectura Aplicacional versus Arquitectura Informacional** - Quando a informação não é gerida de forma adequada pelas diversas aplicações, obrigando a um esforço significativo adicional para garantir a coerência da mesma entre as aplicações. Outro exemplo é o esforço adicional necessário de programação, ao migrar dados entre aplicações que usam estruturas de dados diferentes, mas com a mesma informação.

A questão é como avaliar estes alinhamentos de forma objectiva, evitando estes problemas. Diversas heurísticas e métricas associadas têm sido publicadas (Pereira & Sousa, 2003; Sousa *et al.*, 2004; Sousa *et al.*, 2007). Recentemente foi proposta uma técnica inovadora baseada numa metáfora entre estes desalinhamentos e as doenças humanas, recorrendo a abordagens da medicina (Carvalho & Sousa, 2008). Seguidamente listam-se algumas heurísticas referidas, dada a sua relevância para este trabalho.

#### Alinhamento Arquitectura de Negócio – Arquitectura Informacional

- As entidades informacionais são criadas apenas por um processo.

- Os processos de negócio criam, actualizam e/ou apagam pelo menos uma entidade informacional.
- As entidades informacionais são lidas pelo menos por um processo de negócio.

#### Alinhamento Arquitectura de Negócio – Arquitectura Aplicacional

- Cada processo de negócio deve ser suportado por, pelo menos, uma aplicação.
- As aplicações devem estar associadas a, pelo menos, um processo de negócio.

#### Alinhamento Arquitectura Aplicacional – Arquitectura Informacional

- Uma entidade informacional é gerida apenas por uma aplicação.
- A gestão dos dados deve ser automática entre as aplicações.

### 2.2.2 A representação de uma Arquitectura Empresarial

Mas como representar a Arquitectura Empresarial, na sua multidimensionalidade, com os seus vários modelos e perspectivas? Recorrendo a um *Framework*. Os *Frameworks* suportam a representação de objectos complexos, permitindo observá-los segundo várias vistas e níveis de detalhe. Existem várias centenas, nomeadamente:

- AGATE – *Atelier de Gestion de l'Architecture des Systèmes d'Information et de Communication*, um *framework* promovido pela *Délégation Générale pour l'Armement* Francesa (Agate, 2005).
- CAF – *C4ISR Architecture Framework* do Departamento de Defesa dos EUA (C4ISR, 1997).
- DoDAF – *Department of Defense Architecture Framework* dos EUA (Dodaf, 2007).
- e-GIF – *e-Government Interoperability Framework* do Reino Unido (e-Gif, 2005). E2AF – *Extended Enterprise Architecture Framework* do *Institute for Enterprise Architecture Developments* (E2af, 2005).
- FEAF – *Federal Enterprise Architecture Framework* (Feaf, 1999).
- IAF – *Integrated Architecture Framework* da Cap Gemini (Iaf, 1993).
- IIF – *Australia Government Information Interoperability Framework* (Iif, 2006).
- *Microsoft Framework for Enterprise Architecture* (Microsoft, 2009).
- MODAF – *Ministry of Defense Architecture Framework* (Modaf, 2008).
- NAF – *NATO Architecture Framework* (Nato, 2005).
- TEAF – *Treasury Enterprise Architecture Framework* (Teaf, 2000).
- TOGAF – *The Open Group Architecture Framework* (Togaf, 2009).
- *Framework* de Zachman (Zachman, 1987).

Note-se a popularidade destes instrumentos na Administração Pública e mais especificamente na Defesa, os quais são ambientes muito complexos. Os *frameworks* são usados como instrumento de referência para a auditoria, de garantia de interoperabilidade e

de coerência global, numa altura em que assistimos à integração de sistemas e de dados, devido às exigências do Governo Electrónico e da Guerra em Rede.

O *Framework* de Zachman foi o precursor e é considerado uma norma *de facto* neste campo, tendo inspirado muitos dos *frameworks* anteriormente referidos. Foi criado por John Zachman em 1987 (Zachman, 1987) e relaciona numa matriz as **perspectivas** dos vários actores (*Planner, Owner, Designer, Builder e Subcontractor*) segundo linhas, com um conjunto de seis questões elementares em colunas (*What, How, Where, Who, When e Why*), as **dimensões**. Em cada célula teremos a descrição dos vários componentes e seus artefactos, usando diversos formatos e notações. Trata-se de uma abordagem generalista, mas extremamente exhaustiva, permitindo caracterizar com grande detalhe a Arquitectura Empresarial de uma dada organização. Aliás, dada a sua generalidade, pode representar qualquer objecto complexo. Segundo Zachman o *framework* é na realidade uma ontologia, composta por um conjunto estruturado de componentes essenciais de um objecto, que devem ser explicitados para criar, operar e alterar esse objecto (Zachman, 2008). Na Figura 2.3 podemos observar o *Framework* de Zachman.

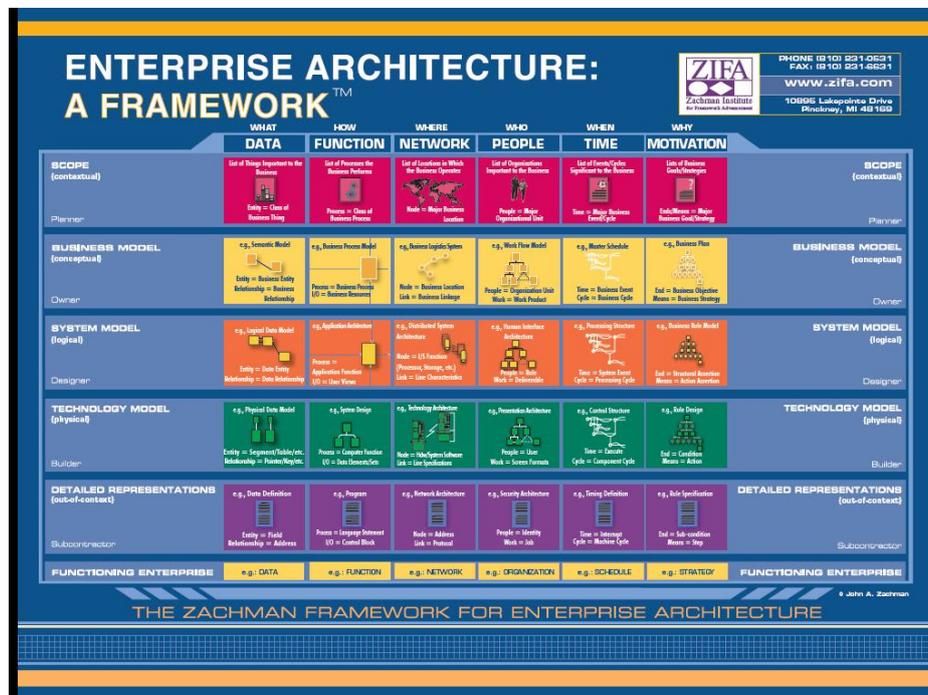


Figura 2.3 - *Framework* de Zachman (versão 2003).

Este *framework* tem sofrido alguns ajustes ao longo do tempo, particularmente ao nível do grafismo e do uso de uma linguagem mais precisa. No entanto os conceitos fundamentais mantiveram-se inalterados (Zachman, 2009).

Neste trabalho será usado este *framework*, por ser considerado o *framework* principal para Arquitectura Empresarial. A versão usada é a de 2003, para garantir consistência entre os métodos e as ferramentas usadas.

### 2.2.3 A construção de uma Arquitectura Empresarial

A construção de uma Arquitectura Empresarial usando o *Framework* de Zachman coloca alguns desafios, dada a complexidade e o número potencial de artefactos envolvidos, nas diferentes perspectivas e dimensões. Adicionalmente Zachman não indica nenhuma ordem para o preenchimento, nem indica os modelos a usar. Este tópico tem sido alvo de várias propostas.

Spewak propõe um método intitulado EAP (*Enterprise Architecture Planning*) para a construção de uma Arquitectura Empresarial (Spewak & Hill, 1992). No entanto esta metodologia apenas endereça as duas primeiras perspectivas do *Framework* de Zachman (*Planner, Owner*) e as três primeiras dimensões do *framework* (*What, How e Where*), as únicas à data. Segundo o autor, a EAP foca a definição dos dados, aplicações e tecnologia para a organização no seu todo, em vez de os definir para fins específicos.

Segundo este método as necessidades em termos de dados devem ser identificadas antes da definição das aplicações. Tal significa passar da definição da Arquitectura de Negócio para a definição da Arquitectura Informacional e só depois para a definição da Arquitectura Aplicacional.

Na Figura 2.4 podem ser observados os componentes deste método.

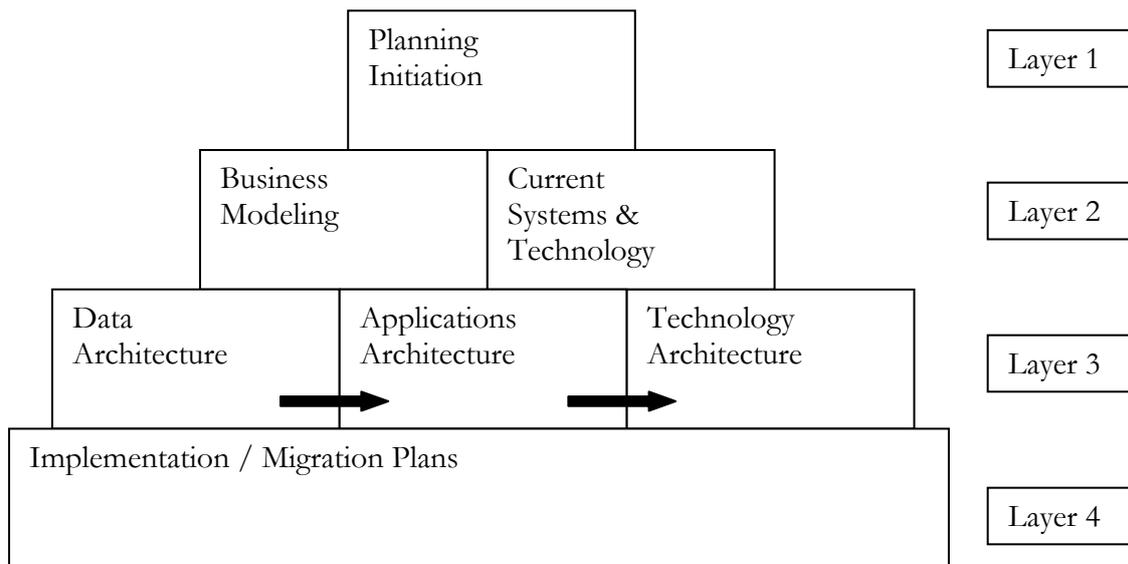


Figura 2.4 - Método Enterprise Architecture Planning  
Fonte: (Spewak & Hill, 1992)

Como se pode observar trata-se de um método em quatro fases:

- *Where We Start* - destina-se a preparar todo o processo subsequente;
- *Where We Are Today* – consiste num levantamento da situação actual em termos de processos de negócio e de sistemas e tecnologias associadas;
- *Where We Want to Be in the Future* – trata-se da definição das arquitecturas de dados, aplicações e tecnologia;
- *How We Get There* – consiste na criação de planos de implementação e de migração.

Existe uma preocupação em garantir o alinhamento entre dados, aplicações e tecnologia através de construção de uma série de matrizes de relacionamento.

Um método que cobre as três primeiras perspectivas do *Framework* de Zachman foi proposto por Pereira & Sousa (2004). Consiste na definição dos vários artefactos de cada célula e numa sequência para o seu preenchimento. Na tabela seguinte podemos observar a sequência de preenchimento proposta por estes autores. O conteúdo de cada célula é constituído por identificação (letra), sequência de preenchimento (número da fase) e precedências (identificações de células precedentes). Por exemplo, a primeira linha não tem precedências e corresponde à fase 1, não existindo ordem para o seu preenchimento, ao contrário das células das linhas seguintes.

	WHAT	HOW	WHERE	WHO	WHEN	WHY
SCOPE (Planner)	A,1	B,1	C,1	D,1	E,1	F,1
ENTERPRISE MODEL (Owner)	G,2,A	H,3,(B+G)	I,4,(C+H)	J,5,(D+I)	K,4,(E+H)	L,4,(F+H)
SYSTEM MODEL (Designer)	M,3,G	N,4,H	O,5,N	P,6,(J+N)	Q,5,N	R,5,(L+N)
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION

Tabela 2.1 - Sequência de preenchimento proposta por Pereira & Sousa (2004).

Na Tabela 2.2 podem ser observados os diversos artefactos propostos (Pereira & Sousa, 2004) para cada célula da *Framework* de Zachman.

	WHAT	HOW	WHERE	WHO	WHEN	WHY
SCOPE (Planner)	List of things important for the business	List of Processes the Business Performs	List of Locations in which the Business Operates	List of Organizations important to the Business	List of Events Significant to the Business	List of Business Goals/Strategies
	- List	- Hierarchical list or tree	- Hierarchical list or tree	- List	- List	- Indented List
ENTERPRISE MODEL (Owner)	Semantic Model	The Business Process Model	The Business Logistics System	Work Flow Model	Master Schedule	Business Plan
	- Entities Diagram - Entities Dictionary	- Entities vs Processes Matrix - Processes Dictionary - Activity Diagram	- Functional Decomposition	- Organization Chart - Processes vs Organization Matrix	- Business Execution Plan	- Business Table
SYSTEM MODEL (Designer)	Logical Data Model	Application Architecture	The Distributed Systems Architecture	Human Interface Architecture	Processing Structure	Business Rules
	- Classes Diagram	- Systems vs Processes Matrix - Systems vs Entities Matrix - Systems Dictionary	- Systems Diagram	- Systems vs Roles Matrix	- State Diagram	- Systems vs Business Rules Matrix
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION

Tabela 2.2 - Artefactos propostos para cada célula da Framework de Zachman.

Foi também proposto um método usando a norma MDA (Model Driven Architecture) do OMG (Object Management Group) (Omg, 2009) para construir os diversos modelos

(Frankel *et al.*, 2003). Mais recentemente esta abordagem foi aprofundada (Ostadzadeh *et al.*, 2007). O MDA introduz a noção de MDD (Model Driven Development), ou seja, permite a criação do código a partir dos modelos. Trata-se de um conceito que permitirá a execução dos modelos, conduzindo a que, no futuro, o código das aplicações sejam os modelos destas.

A MDA estrutura um sistema em três modelos, que correspondem a três fases no processo de criação de código:

- CIM (Computation Independent Model) – Este modelo representa uma vista do sistema que é independente da perspectiva computacional (por exemplo, modelo e vocabulário de domínio);
- PIM (Platform Independent Model) – Este modelo já tem uma perspectiva computacional, mas com independência face à plataforma específica de computação (por exemplo, um modelo de dados lógico);
- PSM (Platform Specific Model) – Este modelo corresponde à vista específica tendo em conta a plataforma computacional usada (por exemplo, um modelo de dados físico).

A passagem entre os vários modelos exige um esforço de mapeamento entre cada par de modelos, para permitir automatizar a criação de código, quando se repetir posteriormente o processo.

Esta abordagem além de permitir gerar automaticamente código, protege os investimentos de uma empresa, ao manter a modelização do negócio, quando se altera a tecnologia que o suporta.

Segundo os autores a utilização do MDA permite cobrir quase todas as células, exceptuam-se:

- A primeira linha, correspondente à perspectiva do *Planner*, por considerarem que Zachman indicou que se tratavam de listas e não de modelos;
- A última linha, correspondente à perspectiva da organização em operação, por não incorporar modelos, mas sim a “realidade” (código, dados, pessoas, etc);
- As células *Who*, nas perspectivas *Builder* e *Subcontractor*, por não haver modelos adequados.

A vantagem principal desta abordagem reside não só na coerência proposta para os diversos modelos, mas também por associar a geração automática de código, potenciando um alinhamento automático entre a modelização do negócio e a sua automatização.

A existência de ferramentas que suportam o desenvolvimento e a manutenção de Arquitecturas Empresariais conduziu à apresentação de propostas proprietárias para a sua modelização, dado não existirem abordagens totalmente normalizadas. Um exemplo é o caso da Sparx Systems Pty Ltd, na modelização de Arquitecturas Empresariais usando o *Framework* de Zachman no seu produto Enterprise Architect (Sparx, 2009). A solução proposta consiste num *add-in* que recorre ao UML (Unified Modeling Language), versão 2.1, à criação de um *profile* UML para o *Framework* de Zachman e a extensões para várias

áreas como modelação de requisitos e teste (Sparks, 2008). Os modelos usados podem ser vistos na Tabela 2.3.

	WHAT	HOW	WHERE	WHO	WHEN	WHY
SCOPE (Planner)	List of things important for the business	List of Processes the Business Performs	List of Locations in which the Business Operates	List of Organizations important to the Business	List of Events Significant to the Business	List of Business Goals/Strategies
	- Business Data (2)	- High Level Business Process (2)	- Business Locations (2)	- Organization Chart (2)	- Business Events (2)	- Business Motivation (2)
ENTERPRISE MODEL (Owner)	Semantic Model	The Business Process Model	The Business Logistics System	Work Flow Model	Master Schedule	Business Plan
	- Data Map (2) - Add-In Generated Process Map (2)	- Process Analysis (2)	- Business Logistics (2)	- BPMN (2)	- Event Schedule (2)	- Strategy Map (2) - Mind Mapping (3)
SYSTEM MODEL (Designer)	Logical Data Model	Application Architecture	The Distributed Systems Architecture	Human Interface Architecture	Processing Structure	Business Rules
	- Class ( <i>PIM</i> ) (1)	- Activity (1)	- Data Distribution Architecture (2)	- Use Case (1)	- State Transition (1)	- Business Rule Model (2) - Requirements (3)
TECHNOLOGY MODEL (Builder)	Physical Data Model	System Design	Technology/Architecture	Presentation Architecture	Control Structure	Rule Design
	- Physical Data Model (2)	- Class ( <i>PSM</i> ) (1) - Component (1)	- Deployment (1)	- User Interface (3)	- Interaction (1) - Communication (1)	- Rule Design (2)
DETAILED REPRESENTATIONS (Sub-contractor)	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Specification
	- Data Definition (2) ( <i>Enterprise Architect DDL Generation</i> )	( <i>Enterprise Architect Code Generation</i> )	- Network Architecture (2)	- Security Architecture (2)	- Timing (1)	- Rule Specification (2)
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION

Diagramas UML (1); UML *Profile* para Zachman *Framework* (2); Extensões Enterprise Architect (3)

Tabela 2.3 - Modelos usados no Enterprise Architect para o *add-in* com o *Framework* de Zachman.  
Fonte: (Sparks, 2008)

## 2.2.4 Ferramentas de suporte a Arquitecturas Empresariais

Algumas das desvantagens que têm sido apontadas às Arquitecturas Empresariais consistem no facto de gerarem muita documentação e demorarem muito tempo a serem construídas. O aparecimento de ferramentas que suportam a modelização de Arquitecturas Empresariais veio minimizar estas desvantagens, permitindo alguns benefícios adicionais, como:

- Validar os artefactos usados, por suportarem as linguagens ou notações associadas, contribuindo para o alinhamento no interior de cada célula e entre elas, evitando erros;
- Permitir a edição cooperativa, contribuindo para reduzir tempos ao envolver mais recursos simultaneamente;

- Permitir a publicação, constituindo-se num repositório “público”, expondo a Arquitectura Empresarial à crítica e à sua constante actualização, à medida que o negócio se adapta à envolvente externa;
- Suportar a geração de código a partir de modelos, aumentando a utilidade da Arquitectura Empresarial ao integrá-la no processo de produção de software e garantindo o alinhamento entre os modelos e o código associado;
- Ser um referencial comum entre os recursos humanos associados ao negócio e os associados à tecnologia;
- Tornar-se numa base de conhecimento sobre a organização.

Entre as ferramentas que suportam o *Framework* de Zachman, salientam-se as seguintes:

- ARIS (Ids Scheer, 2009);
- Enterprise Architect (Sparx, 2009);
- EVA Netmodeler (Enterprise Value Architect Netmodeler)(Promis, 2009);
- IBM Rational System Architect (Ibm, 2009).

Algumas destas ferramentas associam à modelização do *framework* a possibilidade de geração de código, seguindo uma abordagem MDA. Trata-se de um valor acrescentado importante, pois reforça o alinhamento entre a modelização do negócio e a sua automatização pela tecnologia.

Tendo em conta este critério importante e a relação funcionalidade/custo optou-se pela utilização da ferramenta Enterprise Architect (Business and Software Engineering Edition, versão 7.5.845).

## 2.3 Ontologias

Como foi anteriormente referido iremos recorrer a uma ontologia para partilhar o conhecimento do domínio ao nível dos indicadores de mobilidade. Curiosamente, Zachman considera o seu *framework* uma ontologia, a qual é composta por um conjunto estruturado de componentes essenciais de um objecto, que devem ser explicitados para criar, operar e alterar esse objecto (Zachman, 2008). Mas vejamos outras definições deste conceito.

Segundo Gruber (1993), uma ontologia é “uma especificação explícita dos objectos, conceitos e outras entidades de um dado domínio, bem como das suas relações”. Outra definição mais pragmática define ontologia como “um vocabulário comum para investigadores que necessitem de partilhar informação sobre um domínio e que inclui definições computáveis dos conceitos básicos do domínio e das relações entre eles” (Noy & McGuinness, 2001). Segundo estes mesmos autores uma ontologia poderá servir para:

- Partilhar conhecimento sobre a estrutura da informação entre pessoas ou agentes de software;
- Permitir a reutilização do conhecimento do domínio;
- Explicitar assumpções sobre o domínio;
- Separar conhecimento do domínio de conhecimento operacional;

- Analisar conhecimento do domínio.

As ontologias são cada vez mais usadas para partilhar o conhecimento de um domínio. A sua utilização ultrapassou a comunidade da Inteligência Artificial e podemos encontrá-las não só em aplicações científicas, como também em aplicações empresariais. No campo científico podemos encontrar exemplos na medicina e na biologia, com vocabulários estruturados, organizando o conhecimento. Na área empresarial salientam-se, pela sua visibilidade, as categorizações de produtos em sites de comércio electrónico ou de leilões, agrupando os diversos produtos e, claro, as arquitecturas empresariais, como Zachman referiu.

As ontologias podem ser classificadas segundo vários critérios (Berdier & Roussey, 2006; Palazzo, 2006):

- Formalização (altamente informais, semi-informais, semi-formal, rigorosamente formal);
- Detalhe (leve, pesada)
- Objectivo (de aplicação, de referência)
- Especificidade (especialista, geral, conceptual, linguística, de metadados, de tarefas e métodos).

Uma ontologia é constituída por várias componentes, nomeadamente:

- Descrição dos conceitos do domínio (chamados Classes ou Conceitos);
- Propriedades de cada conceito (chamadas Slots, Papéis ou Propriedades) e
- Restrições destas propriedades (chamadas Facetas ou Restrições de Papel).

Se juntarmos um conjunto de instâncias individuais de cada classe, obteremos uma base de conhecimento.

A construção de ontologias deve seguir um método com 7 passos (Noy & McGuinness, 2001; Palazzo, 2006):

- Determinar o domínio e o âmbito da Ontologia;
- Avaliar a reutilização de Ontologias existentes;
- Enumerar os termos importantes da Ontologia;
- Definir as Classes e a sua hierarquia;
- Definir as Propriedades das Classes;
- Definir as Restrições das Propriedades;
- Criar as Instâncias das Classes.

Uma ontologia tem também um ciclo de vida característico (TopQuadrant, 2003):

- Criação;
- Preenchimento;

- Validação;
- Implementação;
- Manutenção;
- Evolução.

A manipulação automática de uma ontologia implica a utilização de linguagens que possam ser interpretadas por máquinas. Este é um dos conceitos base na implementação da Web Semântica (SemanticWeb, 2009). As linguagens de ontologias são linguagens formais, maioritariamente normalizadas, embora também existam normas proprietárias. O W3C (World Wide Web Consortium) (W3C, 2009) suporta a OWL (Web Ontology Language), uma família de linguagens composta por OWL DL, OWL Lite e OWL Full. O OWL é o culminar de uma evolução que passou por linguagens anteriores como:

- RDF (Resource Description Framework), um modelo de dados para metainformação, usado para descrever ou modelizar a informação de recursos localizados na Web (Wikipedia/Rdf, 2009);
- RDFS (RDF Schema), a qual permite estruturar RDFs em ontologias (Wikipedia/Rdfs, 2009);
- OIL (Ontology Interchange Language ou Ontology Inference Layer), uma linguagem desenvolvida no âmbito do projecto europeu IST OntoKnowledge (Wikipedia/Oil, 2009);
- DAML (DARPA Agent Markup Language), desenvolvida pela DARPA e baseada em RDF (Wikipedia/Daml, 2009);
- SHOE (Simple HTML Ontology Extensions), um conjunto de extensões HTML que dão conteúdo semântico a páginas web (Wikipedia/Daml, 2009);
- DAML+OIL, uma linguagem combinando os esforços dos dois lados do Atlântico, juntando DAML, OIL e SHOE e que culminou no esforço de normalização do W3C com o OWL (Wikipedia/Daml+Oil, 2009).

Para favorecer a reutilização de ontologias existem vários motores de busca específicos, nomeadamente:

- OLS (Ontology Lookup Service), do European Bioinformatics Institute, permite pesquisar ontologias da área da biologia e medicina (Ebi, 2009);
- ONTOSEARCH2, da Universidade de Aberdeen, permite pesquisar centenas de ficheiros OWL (Aberdeen, 2009);
- SWOOGLE, da Universidade de Maryland, permite pesquisar mais de 10.000 ontologias (Umbc, 2009).

### **2.3.1 Ferramentas de suporte a Ontologias**

Para suportar uma ontologia ao longo do seu ciclo de vida são usadas ferramentas específicas. Tal facilita também o processo de exportação da ontologia para integração numa ferramenta de Arquitectura Empresarial. Existem inúmeras ferramentas embora apresentem algumas limitações, particularmente no que toca à vertente de exportação, interoperabilidade e trabalho colaborativo (Youn & McLeod, 2006). Foram feitos alguns testes com as seguintes:

- NEON (NeOn, 2009);
- OWLEditor (Owl Editor, 2009);
- Protégé 2000 (Protégé, 2009);
- TopBraid Composer (TopBraid, 2009).

O requisito fundamental para seleccionar a ferramenta de edição de ontologias foi a possibilidade de interoperar com a ferramenta de Arquitectura Empresarial, Enterprise Architect, possibilitando a integração de ontologias nalguns artefactos da *Framework* de Zachman.

Com este objectivo foram avaliadas várias ferramentas, nomeadamente OWLEditor, NeON Toolkit e Protégé (3.4, 3.4.1 e 4.0 Beta). Os resultados podem ser consultados em anexo.

O OWL Editor é um editor de OWL gratuito, desenvolvido pela Model Futures (Owl Editor, 2009). Esta empresa é especializada em Arquitectura Empresarial e Ontologias. Esteve envolvida no desenvolvimento do *Framework* MoDAF (Modaf, 2008).

Esta ferramenta é oferecida sem licença, nem garantia. A versão testada foi a 0.2.0.37, uma versão Alfa. As funcionalidades são limitadas, permitindo a construção de hierarquias de classes, comentar cada classe e juntar atributos do Dublin Core (The Dublin Core Metadata Initiative, 2009). Permite a importação de OWL, EXPRESS XML (gerado pela ferramenta Eurostep EXPRESS Parser – EEP) e XMI (XML Metadata Interchange) versão 1.1. É preferido o XMI criado por exportadores baseados no IDE (Integrated Development Environment) NetBeans. Ao nível da exportação permite enviar um ficheiro CSV (Comma Separated Values), mas a parte mais interessante é a possibilidade de exportação directa para o modelo SPARX EA, usado no Enterprise Architect.

O NeOn Toolkit surgiu no âmbito do projecto Europeu NeOn IST-2005-027595 (NeOn, 2009) o qual pretende fazer avançar o estado da arte na utilização de ontologias em aplicações semânticas de grande dimensão em organizações distribuídas. O projecto pretende suportar ontologias em rede, criadas colaborativamente e em constante evolução. O NeOn Toolkit é uma ambiente de engenharia de ontologias, construído sobre o IDE Eclipse. Suporta a edição, visualização, consulta e importação/exportação usando as linguagens F-Logic (FLogic, 2009), RDFS e OWL. É complementado por um conjunto de *plug-ins* comerciais que permite expandir a funcionalidade básica. A versão avaliada foi a 1.2.2 build B904.

O Protégé foi desenvolvido pelo Stanford Center for Biomedical Informatics Research da Stanford University School of Medicine (Protégé, 2009). Trata-se de uma plataforma em código aberto gratuita, baseada em Java e extensível, que permite a criação, visualização e manipulação de ontologias.

O Protégé surge actualmente em três variantes: Protégé 3.X, Protégé 4.0 e WebProtégé 0.5 alpha. Esta última variante destina-se à elaboração colaborativa de ontologias e ainda está numa versão alfa, sendo muito limitada funcionalmente, particularmente ao nível de exportação. A grande diferença entre as versões 3.X e a 4.0 reside no facto da primeira suportar OWL 1.0, RDFS e Frames, enquanto que a última suporta apenas OWL 2.0, com a agravante de ter muito menos *plug-ins*, particularmente para conversão para UML. Suporta

também a possibilidade de representar a ontologia num diagrama UML usando ODM (Ontology Definition Metamodel) (Odm, 2009) do OMG, o que poderá ser interessante em termos de interoperabilidade no futuro. A avaliação focou-se numa primeira fase no Protégé 3.4 e, numa segunda fase, no Protégé 3.4.1, quando este foi disponibilizado.

## 3 Contexto do problema e método de construção da Arquitectura Empresarial

Este capítulo apresenta o método de construção de uma Arquitectura Empresarial que será seguido no caso de estudo. De forma a enquadrar a sua utilização apresenta-se previamente o contexto do problema a abordar.

### 3.1 Contexto do problema abordado

#### 3.1.1 Mobilidade sustentável

A mobilidade é um factor crucial no estilo de vida actual. Torna-se difícil imaginar um mundo sem mobilidade. Os padrões de mobilidade são cada vez mais complexos, particularmente nas regiões metropolitanas, com deslocações pendulares casa-trabalho de natureza multimodal. No entanto a mobilidade que conhecemos está ameaçada por não ter uma natureza sustentável, dada a escassez dos recursos fósseis que consome e o aquecimento global que induz.

Para manter e desenvolver a mobilidade é importante agir, particularmente garantindo a sua sustentabilidade. Este é um dos grandes objectivos que tem conduzido à criação de planos de mobilidade. No entanto, para que a implementação de qualquer plano seja bem sucedida, deve existir um acompanhamento, uma medição dos seus resultados, que permita avaliar e garantir o sucesso da referida implementação.

#### 3.1.2 Observatórios de Mobilidade

Os Observatórios de Mobilidade têm surgido exactamente com o fim de funcionarem como ferramentas de gestão da mobilidade, medindo os resultados da implementação dos referidos planos de mobilidade, através de um painel de controlo constituído por uma bateria de indicadores seleccionados.

Se considerarmos que a gestão da mobilidade pode ser encarada como um processo de melhoria contínua, podemos modelizá-la segundo um ciclo PDCA (Plan-Do-Check-Act), também chamado ciclo de Deming ou de Shewhart (Wikipedia/PDCA, 2009), o qual pode ser observado na Figura 3.1.

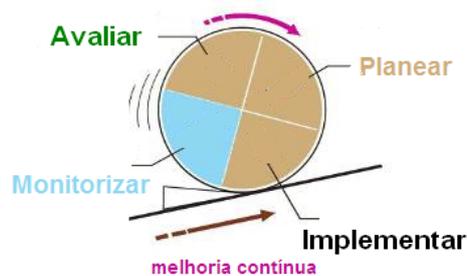


Figura 3.1 - Ciclo de Deming.

Desta forma, a gestão da mobilidade encarada como um processo de melhoria contínua, teria várias fases (Gaspar *et al.*, 2008):

- Planeamento (Plan) – A primeira fase, na qual seriam elaborados planos globais ou de pormenor no campo da mobilidade, recorrendo, por exemplo, ao conhecimento da realidade existente e à simulação.
- Implementação (Do) – Posteriormente seriam executadas acções de implementação dos planos anteriormente definidos, com criação de cadernos de encargos, lançamento de concursos, avaliação das propostas, selecção de fornecedores e acompanhamento da implementação. Nesta fase podemos incluir também a exploração quotidiana dos sistemas, soluções ou equipamentos criados.
- Monitorização (Check) – Seguidamente teríamos de avaliar os resultados, através da obtenção de indicadores sobre a mobilidade, de forma a medir o sucesso da implementação dos planos anteriormente definidos e permitir a tomada de decisão. Esta fase incluiria também a divulgação dos resultados obtidos.
- Avaliação (Act) – Finalmente teríamos um conjunto de acções tendentes a corrigir eventuais desvios significativos detectados na implementação dos planos ou a alterar os próprios planos. E o ciclo fechar-se-ia com a alteração dos planos originais e o recomeço de todo o processo na fase de planeamento ou, neste caso, re-planeamento.

Uma perspectiva funcional abrangente do Observatório corresponderia ao assumir de todas as fases. Uma perspectiva típica, associada à própria designação (observatório), limitar-se-ia às funções de controlo e de monitorização.

### **3.1.3 Âmbito de um Observatório de Mobilidade**

O âmbito de um Observatório de Mobilidade pode ser bastante abrangente, numa perspectiva funcional, como foi anteriormente referido, ou focado no papel da monitorização e controlo, como é o caso presente. No entanto, mesmo com esta focagem funcional, existe um nível importante de variabilidade do âmbito, associado à sua escala geográfica e às temáticas cobertas.

A escala geográfica (micro, meso ou macro) condiciona o tipo de informação a processar. Por exemplo, não faz sentido a nível local, num concelho, contabilizar Km de via férrea, enquanto que a nível nacional faz sentido.

A quantidade de temáticas está associada à natureza multidisciplinar da problemática da mobilidade. Para avaliar o sucesso de uma política de mobilidade é necessário considerar vários temas (Gaspar *et al.*, 2008), como, por exemplo:

- Sócio-economia – que contribui com dados gerais sobre a sociedade e desenvolvimento económico, como população, rendimento e emprego;
- Ordenamento do Território – que contribui com informação sobre a ocupação do território, como a densidade populacional ou a superfície das cidades;
- Ambiente – que contribui com dados sobre a poluição provocada pelos veículos;
- Energia – que contribui com informação sobre a eficiência energética dos veículos e sobre a utilização de energias alternativas;

- Infraestruturas de transportes – que contribui com dados sobre a extensão das redes de transportes e sobre a oferta de estacionamento, por exemplo;
- Serviços de transportes – que contribui com dados sobre os serviços prestados pelos vários operadores, como longitude das redes, densidades das mesmas ou número de paragens;
- Tecnologias associadas – que contribui com informação sobre a incorporação de tecnologias de transportes inteligentes, como o uso de sistemas de navegação ou a utilização de bilhética sem contacto;
- Segurança – que contribui com dados sobre a sinistralidade;
- Satisfação dos utilizadores – que avalia a satisfação dos utilizadores face aos serviços prestados.

### 3.1.4 Requisitos gerais de um Observatório de Mobilidade

Além da cobertura funcional, do âmbito e das temáticas associadas, um Observatório de Mobilidade deverá respeitar um conjunto de requisitos gerais (Gaspar *et al.*, 2008):

- O Observatório deverá ser um instrumento para a implementação de uma Estratégia de Mobilidade.
- Deverá ser uma referência para todo o trabalho no campo da Mobilidade na região a ele associada.
- Deverá suportar uma discussão técnica e imparcial sobre a Mobilidade.
- Deverá contribuir para a sensibilização do público em geral na adopção de padrões de mobilidade sustentável.
- Deverá permitir a captação de informação de diferentes origens, capaz de ser tratada e organizada de forma eficaz, através da construção de indicadores que possam representar as tendências de evolução e desempenho do sector dos transportes de pessoas e mercadorias.
- Deverá divulgar esses indicadores de forma a que possam ser utilizados por diferentes agentes, públicos e privados, na sua acção de planeamento, programação e preparação de actuação estratégica (com a correspondente definição de metas), das acções concretas para as atingir e verificação dos resultados atingidos, no contexto de uma Estratégia de Mobilidade.
- Deverá permitir a divulgação de boas práticas, reconhecidas pelos resultados obtidos por actuações realizadas noutros locais e realidades.
- Deverá permitir a reflexão sobre a forma de fazer evoluir a informação no sentido de a colocar ao serviço da resolução dos problemas prioritários definidos pelas agendas da governação central, regional e local.
- Para que o Observatório a criar tenha capacidade de cumprir todos estes desígnios, deverá ter autonomia funcional.
- Deverá ainda ter uma forte liderança e capacidade de diálogo com as diversas entidades passíveis de fornecerem informação, assim como com outros Observatórios nas áreas de cruzamento de interesses - ordenamento do território, ambiente, e energia, por exemplo – questão essencial para o êxito do Observatório a criar.
- Deverá assegurar a sua sustentabilidade económica.

A sustentabilidade económica é uma preocupação fundamental na construção de um Observatório deste tipo. Ela pode ser potenciada atendendo a algumas orientações fundamentais seguidamente enunciadas (Gaspar *et al.*, 2008).

### **Contenção quanto à quantidade de informação a recolher**

A mobilidade é um tema bastante abrangente e facilmente podem ser identificados inúmeros indicadores, com algum interesse. No entanto, importa não esquecer que cada indicador implica recolha de dados, os quais são posteriormente usados para calcular o indicador em causa. Trata-se, portanto, de um esforço não desprezável, se o indicador não estiver já disponível noutra entidade.

A multiplicação de indicadores poderá ter como efeito a dispersão do esforço da equipa e poderá condicionar a produção do painel de indicadores com a periodicidade exigida, conduzindo à perda de credibilidade do próprio Observatório. Desta forma, o número de indicadores deve ser minimizado e deve apenas corresponder ao número estritamente necessário para medir a implementação dos objectivos da Estratégia de Mobilidade.

### **Cooperação com agentes-fonte**

O funcionamento do Observatório deverá ser baseado na construção de um rede de cooperação envolvendo os vários actores do sector, no âmbito da qual cedem a sua informação, os seus dados e/ou indicadores e, em contrapartida, recebem conhecimento na forma da análise crítica da mobilidade regional e sobre boas práticas. Este mecanismo de funcionamento permite ao Observatório operar com custos baixos de aquisição da matéria-prima e reduz a sua máquina administrativa, potenciando a sua sustentabilidade económica.

### **Operar com recursos financeiros limitados**

O Observatório não deverá ter problemas para aceder a financiamento para a sua criação, tipicamente associado a fundos comunitários. No entanto, importa salientar que estes mecanismos apoiam o investimento e não a exploração, pelo que o Observatório deverá ser dimensionado para ter uma estrutura de recursos leve e flexível.

### **3.1.5 Os Indicadores de um Observatório de Mobilidade**

O principal resultado de um Observatório de Mobilidade são os indicadores produzidos. Os indicadores estão intimamente ligados ao âmbito do Observatório. Quanto maior o seu âmbito, mais indicadores deverá produzir.

A quantidade de indicadores e o esforço de produção associado condicionam a sustentabilidade do Observatório. Um consumo acrescido de recursos face ao nível de resultados ou um atraso na produção dos mesmos, pode condicionar decisivamente a sustentabilidade da operação, quer numa perspectiva operacional, quer numa perspectiva política.

A automatização, ou seja, a informática, tem um papel importante na minimização dos custos e na garantia da sustentabilidade. Por isso, o dimensionamento da bateria de indicadores e, conseqüentemente, dos recursos associados, é decisivo para o sucesso do Observatório e, indirectamente, das políticas de mobilidade em análise.

A definição de cada indicador deverá ser clara, devendo incluir a seguinte metainformação (Gaspar *et al.*, 2008):

- A definição dos conceitos envolvidos;
- A fonte da informação de base;
- A fórmula de cálculo, se for o caso;
- A forma de recolha da informação de base e tecnologias de apoio a mobilizar;
- Quem recolhe a informação;
- Com que periodicidade;
- Data da última informação disponível;
- O universo de abrangência (quarteirão, freguesia, cidade, concelho, AML; etc.).

### 3.1.6 Uma rede de Observatórios

A natureza multidisciplinar da problemática da mobilidade conduz à utilização de indicadores de várias temáticas. É natural que nas temáticas onde já existem Observatórios, como a Sócio-economia (INE), sejam usados indicadores já calculados, minimizando o esforço próprio. Por outro lado, podem coexistir Observatórios de Mobilidade com âmbitos diferenciados. Tal já se verifica a nível internacional, permitindo o *benchmarking*, por exemplo, entre países ou entre áreas metropolitanas. Nada impede que tal aconteça em Portugal.

Para tirar partido de uma rede de Observatórios de Mobilidade (e não só) importa garantir a interoperabilidade entre eles, particularmente ao nível semântico. Tal implica definir uma ontologia que estruture o conhecimento do domínio, particularmente ao nível dos indicadores, permitindo o seu re-aproveitamento, para uso directo, consolidação ou outro tipo de cálculos.

Neste contexto, numa perspectiva mais dinâmica, a busca e a descoberta de indicadores seria uma funcionalidade interessante, o que conduziria à criação de um repositório ou directório de indicadores e de fontes.

## 3.2 A criação de um Observatório de Mobilidade

A criação de um Observatório implica pois a existência de uma estratégia, mas implica também a criação de uma organização, processos de trabalho, definição de dados de base e de indicadores, aplicações informáticas associadas, tecnologias de suporte e todos os custos associados. Tal poderá criar problemas de sustentabilidade económica ao próprio Observatório, se o seu projecto não for estritamente alinhado com os objectivos estratégicos iniciais, minimizando o desperdício de recursos. Note-se que o sucesso da implementação de uma política de mobilidade mede-se ao longo de anos e que o Observatório deverá estar operacional durante este período, produzindo os indicadores necessários em tempo útil, de forma a garantir o sucesso da implementação das políticas de mobilidade.

Para responder ao desafio da criação de um Observatório de Mobilidade, alinhado com os seus objectivos estratégicos, com uma estruturação organizacional e de sistemas

informáticos adequada, otimizando os recursos, de forma a potenciar a sustentabilidade económica, foi proposta um método (Gaspar, 2008) composto por diversas fases, de acordo com a Figura 3.2.



Figura 3.2 - Fases do método proposto de criação de um Observatório de Mobilidade.

### 3.2.1 Levantamento da situação actual

Esta fase destina-se a avaliar a situação existente e tomá-la como ponto de partida, quer ao nível das melhores práticas neste campo, quer ao nível da região metropolitana onde se pretende implementar um Observatório de Mobilidade.

Para tal serão desenvolvidos contactos com outros Observatórios, podendo ser realizadas visitas aos mesmos.

Será avaliada a arquitectura institucional da região, o enquadramento legal e serão realizados contactos com as várias entidades relevantes, nomeadamente: operadores de transportes, entidades responsáveis pelas infra-estruturas de transporte (por exemplo: portos, aeroportos, auto-estradas) e com as entidades responsáveis pela governança (por exemplo: autoridades metropolitanas de transportes, autarquias, governos regionais).

### 3.2.2 Posicionamento estratégico

Esta fase destina-se a fazer uma definição estratégica do posicionamento do Observatório e a definir o mesmo. Como num exercício de planeamento estratégico serão definidas a Visão, Missão e Objectivos Estratégicos do Observatório e posteriormente será definido o seu Modelo de Operação e elaborado um Plano de Operação a cinco anos. O objectivo é o de definir claramente quais os resultados que o Observatório deverá produzir, nomeadamente ao nível de serviços, indicadores, relatórios e acções de divulgação. Nesta definição serão tidos em conta os proveitos e custos associados, perspectivando os orçamentos e a operação ao longo de um período de cinco anos, com o objectivo de garantir a sustentabilidade do seu funcionamento.

### 3.2.3 Arquitectura empresarial

A elaboração da Arquitectura Empresarial tem como objectivo detalhar as várias componentes da organização que irá suportar o Observatório, os quais são seguidamente descritos.

#### Arquitectura Organizacional

Esta arquitectura irá descrever a organização que irá suportar o funcionamento do Observatório, indicando estrutura, perfis dos colaboradores e responsabilidades.

## **Arquitectura de Negócio**

Esta arquitectura irá descrever os processos de negócio principais, segundo o Modelo da Cadeia de Valor de Porter (Laudon & Laudon, 2004) que irão ser desenvolvidos no âmbito do Observatório, nomeadamente as diversas actividades que os constituem, quais as entradas e saídas, qual a sua sequenciação e quais os elementos (perfis) responsáveis pela sua execução.

## **Arquitectura Informacional**

A Arquitectura Informacional descreve as entidades informacionais que irão ser usadas pelo Observatório. Trata-se da informação que irá ser recebida de várias entidades e também gerada pelo próprio Observatório e que consistirá na sua maioria em indicadores. Esta Arquitectura descreverá estas entidades, particularmente em termos de formato, conteúdo, proveniência, frequência de obtenção e forma de obtenção.

### *A construção da Ontologia de Indicadores*

Dada a importância da produção de indicadores num Observatório de Mobilidade e dada a tendência para operar no seio de uma rede de Observatórios de Mobilidade ou com Observatórios de outras temáticas complementares, torna-se necessário a criação de uma Ontologia de Indicadores de forma a assegurar a interoperabilidade semântica.

Esta componente será tratada no âmbito da Arquitectura Informacional mas implica alguns passos específicos que são seguidamente detalhados.

Na construção da Ontologia de Indicadores de Mobilidade foi seguido um método em 7 passos (Noy & McGuinness, 2001; Palazzo, 2006):

- Determinar o domínio e o âmbito da Ontologia;
- Avaliar a reutilização de Ontologias existentes;
- Enumerar os termos importantes da Ontologia;
- Definir as Classes e a sua hierarquia;
- Definir as Propriedades das Classes;
- Definir as Restrições das Propriedades;
- Criar as Instâncias das Classes.

De forma a permitir a partilha e extracção de conhecimento a ontologia foi construída usando uma ferramenta de edição de ontologias, permitindo a exportação (e importação) para repositórios de ontologias usando as linguagens específicas existentes. De forma a garantir a integração com a ferramenta de modelização da Arquitectura Empresarial a ontologia é importada, tomando a forma de diagramas UML de classes e de objectos.

## **Arquitectura Aplicacional**

Dado que se pretende maximizar a eficiência do Observatório, os seus processos serão, sempre que possível, automatizados, para reduzir os custos de operação. Para tal serão usadas diversas aplicações informáticas que irão manipular os dados anteriormente definidos na Arquitectura Informacional, segundo os processos de negócio definidos na

Arquitectura de Negócio. Esta arquitectura irá identificar e especificar as diversas aplicações informáticas necessárias.

### **Arquitectura Tecnológica**

O funcionamento desta infra-estrutura é assegurado por um conjunto de sistemas de software e hardware, como, por exemplo, sistemas de gestão de base de dados, sistemas de informação geográfica, servidores aplicativos, máquinas servidoras e redes informáticas.

#### **3.2.4 Disseminação**

A operação de um Observatório deste tipo implica o estabelecimento de mecanismos de cooperação em rede com a comunidade associada à mobilidade na região metropolitana em causa. Para tal importa motivar toda esta comunidade para os objectivos que se pretendem atingir. Este é o objectivo desta fase, que incluirá a preparação de materiais de disseminação e a realização de reuniões com a comunidade já referida. Ou seja, trata-se de uma fase de disseminação prévia, dirigida aos parceiros potenciais, explicando os objectivos e vantagens do Observatório, com o objectivo de os atrair para a parceria. Recordamos que esta parceria é crítica para a sustentabilidade do Observatório, pois o seu modelo de negócio baseia-se na troca de informação (recebe dados e dá indicadores) e não na sua venda (e compra).

#### **3.2.5 Implementação**

Esta fase consiste na construção do Observatório.

Note-se que todas as fases deverão ser acompanhadas por um grupo de pilotagem, constituído por um conjunto de entidades relevantes, tipicamente associadas à governância da mobilidade na região metropolitana em estudo e que estão interessadas na criação de um Observatório. Este grupo terá como objectivo acompanhar as actividades técnicas, apoiar a realização dos trabalhos e validar os resultados.

### **3.3 Método para a construção da Arquitectura Empresarial**

O foco desta tese é a construção da Arquitectura Empresarial enquadrada no método proposto para a criação de um Observatório de Mobilidade. Recorde-se que se pretende garantir o alinhamento da infra-estrutura tecnológica com os objectivos da organização, como forma de contribuir para a sustentabilidade económica da operação do Observatório e que para tal se recorre à construção da Arquitectura Empresarial como instrumento para tal.

Para capturar a Arquitectura Empresarial será usado o *Framework* de Zachman. Desta forma garante-se que a cobertura de todas as dimensões e perspectivas relevantes, de uma forma sistemática. Os diversos artefactos deverão ser representados segundo um modelo integrado, para garantir integridade da modelação e permitir a re-utilização de elementos de um artefacto.

De forma a facilitar a actualização da Arquitectura Empresarial é importante que a mesma seja digitalizada e suportada por uma ferramenta adequada, a qual deverá manipular o repositório com toda a informação.

E, finalmente, para garantir que a Arquitectura Empresarial assume o seu papel estratégico, ao nível do negócio, da informação necessária, das aplicações associadas e da tecnologia de suporte, é necessário que seja exposta. Exposta na medida em que deve ser explicitada através da publicação, para que os diversos actores a possam usar na optimização do presente e no planeamento do futuro. Uma forma de o conseguir é a publicação numa Intranet, exportando o modelo para HTML.

A ferramenta Enterprise Architect responde a estes requisitos (Sparx, 2009).

O passo seguinte é a construção da Arquitectura Empresarial. Zachman não deu indicações detalhadas sobre como deveria ser preenchido o seu *Framework* e construída a Arquitectura Empresarial. Existem diversas abordagens, as quais já foram referidas no Capítulo 2.

Para construir a Arquitectura Empresarial propõe-se uma abordagem híbrida, combinando alguns aspectos de vários métodos e tendo em conta a utilização do Enterprise Architect. O método híbrido a usar incluirá:

- A EAP de Spewak & Hill (1992), no que toca à sequência de criação da Arquitectura Aplicacional após a criação da Arquitectura Informacional e à elaboração de matrizes de CRUD (Create-Read-Update-Delete) avaliando a utilização de entidades informacionais por processos de negócio de forma a identificar aplicações;
- Alguns artefactos e a sequência proposta por Pereira & Sousa (2004), estendida para a Arquitectura Tecnológica;
- Alguns modelos propostos pela Sparx no Enterprise Architect (Sparks, 2008).

Como foi referido a sequência proposta por Pereira & Sousa (2004) cobre apenas as três primeiras linhas ou perspectivas da *Framework* de Zachman (*Planner, Owner e Designer*). Para obtermos a Arquitectura Tecnológica necessitamos também da quarta perspectiva *Builder* ou *Technology Model*. A proposta de sequência de preenchimento estendida pode ser observada na Tabela 3.1.

	WHAT	HOW	WHERE	WHO	WHEN	WHY
SCOPE (Planner)	A,1	B,1	C,1	D,1	E,1	F,1
ENTERPRISE MODEL (Owner)	G,2,A	H,3,(B+G)	I,4,(C+H)	J,5,(D+I)	K,4,(E+H)	L,4,(F+H)
SYSTEM MODEL (Designer)	M,3,G	N,4,H	O,5,N	P,6,(J+N)	Q,5,N	R,5,(L+N)
TECHNOLOGY MODEL (Builder)	S,4,M	T,5,(N+S)	U,6,(O+T)	V,7,P	W,6,(Q+T)	X,6,(R+T)
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION

**Tabela 3.1 - Sequência de preenchimento estendida cobrindo a Arquitectura Tecnológica.**

Face aos objectivos deste trabalho, nem todas as células necessitam de ser preenchidas, nem existe informação para algumas. A sequência final, incluindo apenas as células consideradas relevantes é a indicada na Tabela 3.2. Os artefactos que irão ser usados podem ser consultados na Tabela 3.3.

Definição da Arquitectura Empresarial  
O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

	WHAT	HOW	WHERE	WHO	WHEN	WHY
SCOPE (Planner)	A,1	B,1	C,1	D,1	E,1	F,1
ENTERPRISE MODEL (Owner)	G,2,A	H,3,(B+G)	I,4,(C+H)	J,5,(D+I)	K,4,(E+H)	L,4,(F+H)
SYSTEM MODEL (Designer)	M,3,G	N,4,H	O,5,N	P,6,(J+N)	Q,5,N	R,5,(L+N)
TECHNOLOGY MODEL (Builder)	S,4,M	T,5,(N+S)	U,6,(O+T)	V,7,P	W,6,(Q+T)	X,6,(R+T)
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION

Tabela 3.2 - Sequência de preenchimento estendida, com as células relevantes para este trabalho.

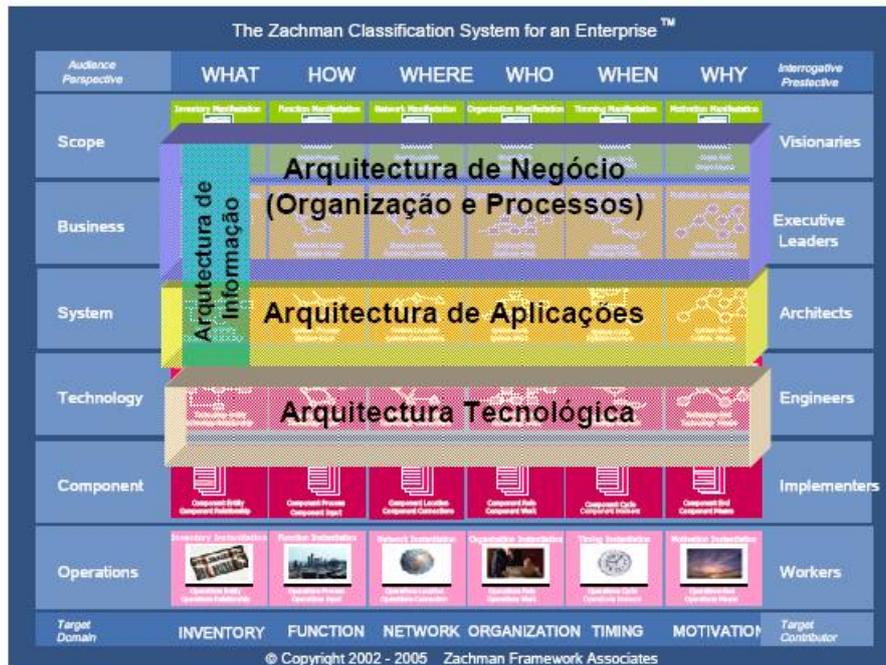
	WHAT	HOW	WHERE	WHO	WHEN	WHY
SCOPE (Planner)	List of things important for the business	List of Processes the Business Performs	List of Locations in which the Business Operates	List of Organizations important to the Business	List of Events Significant to the Business	List of Business Goals/Strategies
	- Business Data (2)	- Business Use Case (2)	- Business Locations (2)	- Use Case (1)	- Business Events (2)	- Business Motivation (2)
ENTERPRISE MODEL (Owner)	Semantic Model	The Business Process Model	The Business Logistics System	Work Flow Model	Master Schedule	Business Plan
	- Data Map (2) - Entities Dictionary	- Process Analysis (2) - Entities vs Processes Matrix - Process Dictionary	- Business Logistics (2)	- Use Case (1) - Organization Chart - Processes vs Organization Matrix	- Event Schedule (2)	- Strategy Map (2)
SYSTEM MODEL (Designer)	Logical Data Model	Application Architecture	The Distributed Systems Architecture	Human Interface Architecture	Processing Structure	Business Rules
	- Class (PIM) (1)	- Component (1) - Systems vs Business Processes Matrix - Systems vs Informational Entities Matrix - Systems Dictionary	- Deployment (1)	- Use Case (1)	- State Transition (1)	- Business Rule Model (2) - Requirements (3)
TECHNOLOGY MODEL (Builder)	Physical Data Model	System Design	Technology/Architecture	Presentation Architecture	Control Structure	Rule Design
	- Physical Data Model (2)	- Component (1)	- Deployment (1)	- User Interface (3)	- Interaction (1) - Communication (1)	- Rule Design (2)
DETAILED REPRESENTATIONS (Sub-contractor)	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Specification
	- Data Definition (2) (Enterprise Architect Code Generation)	(Enterprise Architect Code Generation)	- Network Architecture (2)	- Security Architecture (2)	- Timing (1)	- Rule Specification (2)
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION

Diagramas UML (1); UML Profile para Zachman Framework (2); Extensões Enterprise Architect (3)

Tabela 3.3 – Artefactos usados na modelação da Arquitectura Empresarial do Observatório

Finalmente e para facilitar a comunicação da Arquitectura Empresarial, os artefactos são reorganizados segundo as arquitecturas componentes. O objectivo é usá-los para gerar parte do conteúdo de uma memória descritiva ou de um caderno de encargos. Consideramos que o *framework* de Zachman embora exaustivo, não é tão intuitivo para este fim.

O mapeamento entre as diversas sub-arquitecturas de uma Arquitectura Empresarial e as diversas células do *Framework* de Zachman pode ser observado na Figura 3.3; Sousa, 2005).



**Figura 3.3 - Mapeamento entre sub-arquitecturas de uma Arquitectura Empresarial e *Framework* de Zachman.**

Como se pode verificar a correspondência é a seguinte:

- A Arquitectura Tecnológica corresponde à perspectiva (linha) *Technology (Builder)*;
- A Arquitectura Aplicacional corresponde à perspectiva *System (Designer)*;
- A Arquitectura Informacional corresponde à dimensão *What*, nas suas três primeiras células, contando a partir do topo;
- A Arquitectura de Negócio e Arquitectura Organizacionais correspondem às duas primeiras perspectivas, exceptuando a dimensão *What* (primeira coluna)
  - A Arquitectura Organizacional corresponde às células *Where, Who* e *Why* da perspectiva *Scope (Planner)* e à célula *Why* da perspectiva *Business (Owner)*;
  - A Arquitectura de Negócio corresponde às restantes.

## 4 Caso de Estudo: Arquitectura Empresarial do Observatório de Mobilidade

### 4.1 Introdução

Este trabalho foi desenvolvido no âmbito de um projecto co-financiado pelo Programa Operacional Regional Lisboa e Vale do Tejo 2000-2006, o qual foi dinamizado no contexto da parceria europeia estabelecida entre Lisboa, Valência e Génova no âmbito da Operação Quadro Regional MARE, liderada pela Comissão de Coordenação e Desenvolvimento Regional de Lisboa e Vale do Tejo (CCDR-LVT). O objectivo foi o de estudar a criação de um Observatório da Mobilidade Metropolitana na Região de Lisboa.

De forma a contribuir para a sustentabilidade económica deste Observatório, foi proposta uma metodologia de desenvolvimento, que pretende garantir o alinhamento com os seus objectivos estratégicos, optimizando os recursos associados, particularmente de natureza informática.

A peça chave desta metodologia é a construção de uma Arquitectura Empresarial, como foi anteriormente descrito. Seguidamente será apresentada a sua construção detalhada.

### 4.2 Processo de construção

A construção da Arquitectura Empresarial será suportada na ferramenta Enterprise Architect, a qual suporta a *Framework* de Zachman, além, obviamente, da modelação UML, neste caso UML 2.1.

O primeiro passo é estruturar o projecto de construção da Arquitectura Empresarial. O modelo terá dois pacotes:

- um correspondente ao *Framework* de Zachman, que servirá como base estruturante para a criação da Arquitectura Empresarial;
- e um segundo correspondente à Arquitectura Empresarial específica, onde os diversos elementos criados na *Framework* de Zachman serão replicados e organizados.

Na Figura 4.1 pode ser observada esta estruturação, sendo possível observar um diagrama de pacotes UML com as várias sub-arquitecturas de uma Arquitectura Empresarial, correspondendo cada uma a um pacote. As relações de dependência entre as várias sub-arquitecturas são representadas segundo a proposta de Sousa *et al.* (2007).

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

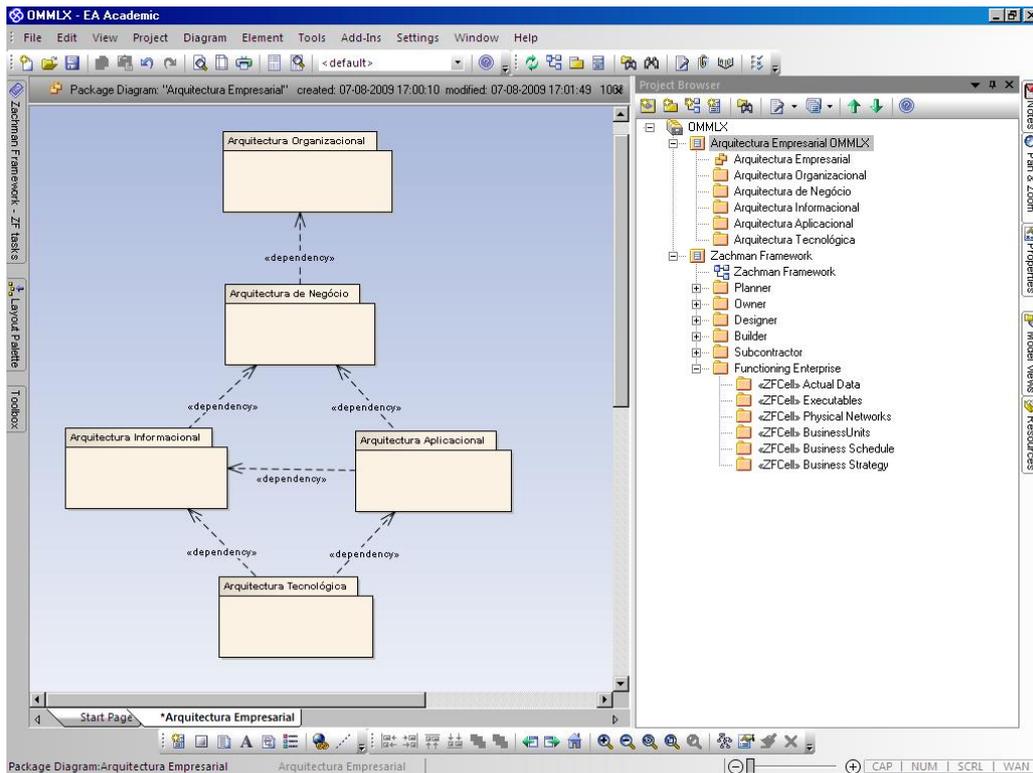


Figura 4.1 - Estruturação inicial da Arquitectura Empresarial.

Para construir a Arquitectura Empresarial vamos usar o método proposto no capítulo 3.

### 4.3 Fase 1 - Células A, B, C, D, E e F

#### 4.3.1 Célula A – Lista de coisas importantes para a organização

Esta célula captura as coisas relevantes para a organização. A Sparx propõe um *Business Data Diagram* para este efeito. Trata-se essencialmente de um diagrama de classes. A proposta é de organizar as coisas relevantes segundo três grupos: activos, produtos e serviços e pessoas. No caso concreto do Observatório salientamos as seguintes coisas:

##### Activos

- Volume de Dados – usados para calcular indicadores, sendo importados a partir de parceiros;
- Indicador – importados de parceiros ou calculados internamente, com vista a constituir o painel de controlo de acompanhamento da implementação da política de mobilidade, segundo os objectivos e estratégias do sector e do Observatório;

##### Produtos e Serviços

- Relatório - com uma frequência anual este relatório analisa a evolução da mobilidade na região e compara com outras regiões de referência;
- Seminário – actividade tipicamente anual de disseminação do último relatório e de partilha de conhecimentos entre interessados e especialistas no tema da mobilidade;
- Estudo – estudo de questões específicas, no campo da mobilidade na região, a pedido de parceiros ou clientes;

- Website – onde se dissemina de forma contínua informação sobre mobilidade, nomeadamente sob a forma dos relatórios anuais, informação sobre os seminários e eventualmente estudos cujos clientes permitam a publicação;

#### Recursos humanos relevantes

- Presidente da CCDR-LVT – o qual articulará o lançamento do Observatório com a estratégia regional Lisboa 2020 e com os programas de financiamento disponíveis;
- Presidente do IMTT – o qual irá incubar o Observatório na sua fase inicial no IMTT e irá, eventualmente, articular uma rede de Observatórios de Mobilidade;
- Presidente da AMTL – o qual enquadrará o Observatório na AMTL na sua fase final.

Na Figura 4.2 podem ser observados os três pacotes associados a estes três itens. No caso dos pacotes Activos e Produtos e Serviços, os elementos não são visualizados devido a fazerem parte de outro pacote.

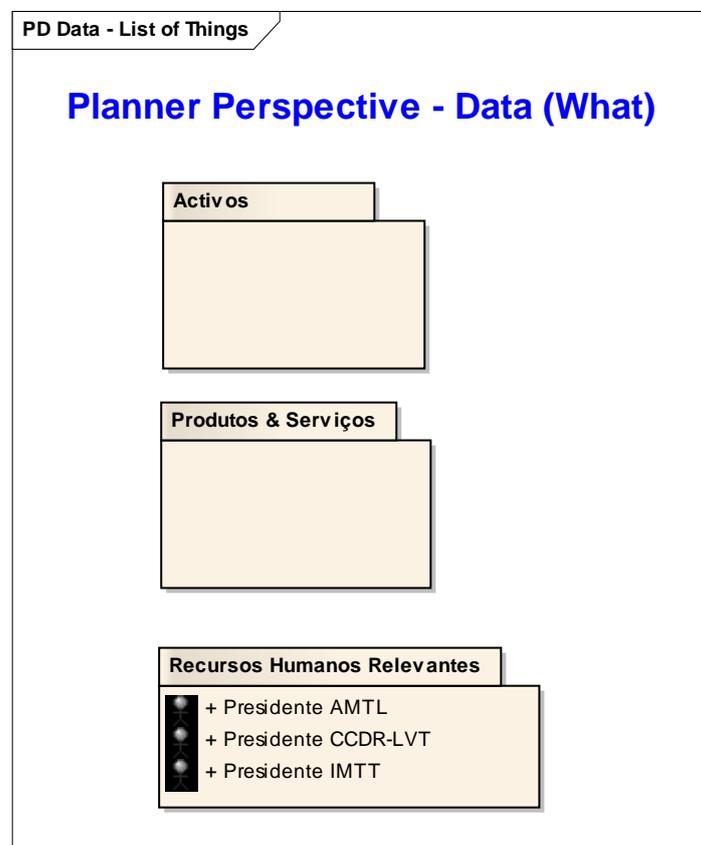


Figura 4.2 - Célula A – Coisas importantes para a organização.

Para os visualizar é necessário abrir os pacotes e visualizar os diagramas associados, dos quais fazem parte os elementos referidos. Tal pode ser observado, para o caso do pacote Produtos e Serviços, na Figura 4.3. Numa fase inicial de modelação os elementos foram colocados directamente neste pacote, no entanto, posteriormente, para efectuar algumas transformações, foi necessário agrupá-los de forma diferente. Embora Zachman proponha uma lista, consideramos que esta abordagem se assemelha a uma lista gráfica. Para tal eliminamos o detalhe nas classes, escondendo, por exemplo, atributos e operações. Note-se que a classe Resultado é abstracta pois nunca é concretizada, tratando-se de uma generalização.

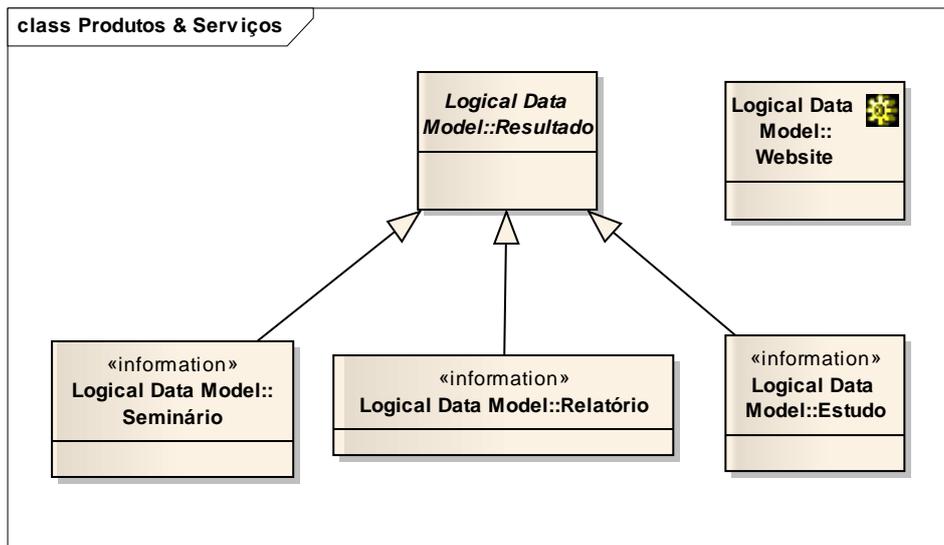


Figura 4.3 – Conteúdo do pacote Produtos e Serviços.

#### 4.3.2 Célula B – Lista de processos realizados pela organização

Para capturar a lista de processos realizados pela organização o Enterprise Architect propõe um *High Level Business Process Diagram*, ou em alternativa, um *Business Usecase Diagram*. Ambos se assemelham, particularmente se no primeiro identificarmos processos e actores. Optamos pelo segundo devido a uma maior familiaridade com esta abordagem. Trata-se basicamente de um diagrama de casos de uso, onde estes representam os processos de negócio da organização. No nosso caso concreto optou-se por capturar apenas os casos de uso primários ou processos de negócio primários, segundo o Modelo de Cadeia de Valor de Porter (Laudon & Laudon, 2004), por serem os que efectivamente acrescentam valor e distinguem o Observatório de outras organizações. Por outro lado o Observatório irá ser integrado noutra entidade que lhe disponibilizará esses processos. Numa primeira fase no IMTT (Instituto da Mobilidade e dos Transportes Terrestres) e, posteriormente, na AMTL (Autoridade Metropolitana de Transportes de Lisboa).

O diagrama referido mostra quatro pacotes, que estruturam os casos de uso:

- Casos de uso primários – os que acrescentam valor e distinguem a organização;
- Casos de uso de suporte - não identificados neste caso concreto;
- Processos de negócio primários – que implementam os casos de uso primários;
- Processos de negócio de suporte – não identificados neste caso concreto.

Clicando no pacote de casos de uso primários é possível ver o diagrama de casos de uso da Figura 4.4, que mostra os vários casos de uso primários, a relação entre eles e os actores que com eles interagem.

Clicando no pacote com os processos de negócio primários é possível visualizar os mesmos e a relação existente com os casos de uso primários (*realization*). Tal pode ser observado na Figura 4.5.

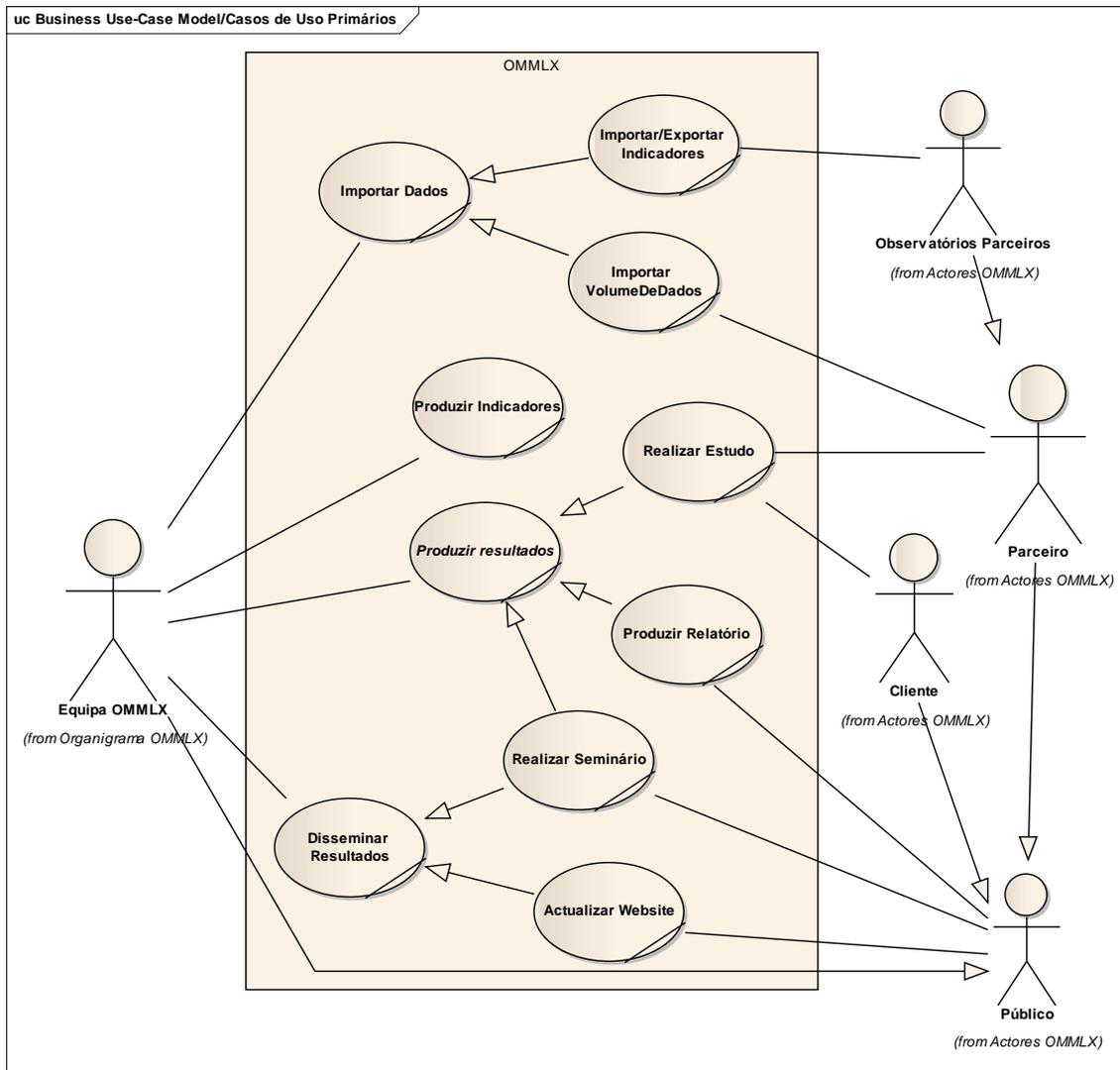


Figura 4.4 - Diagrama de casos de uso primários.

### 4.3.3 Célula C - Lista de locais onde a organização opera

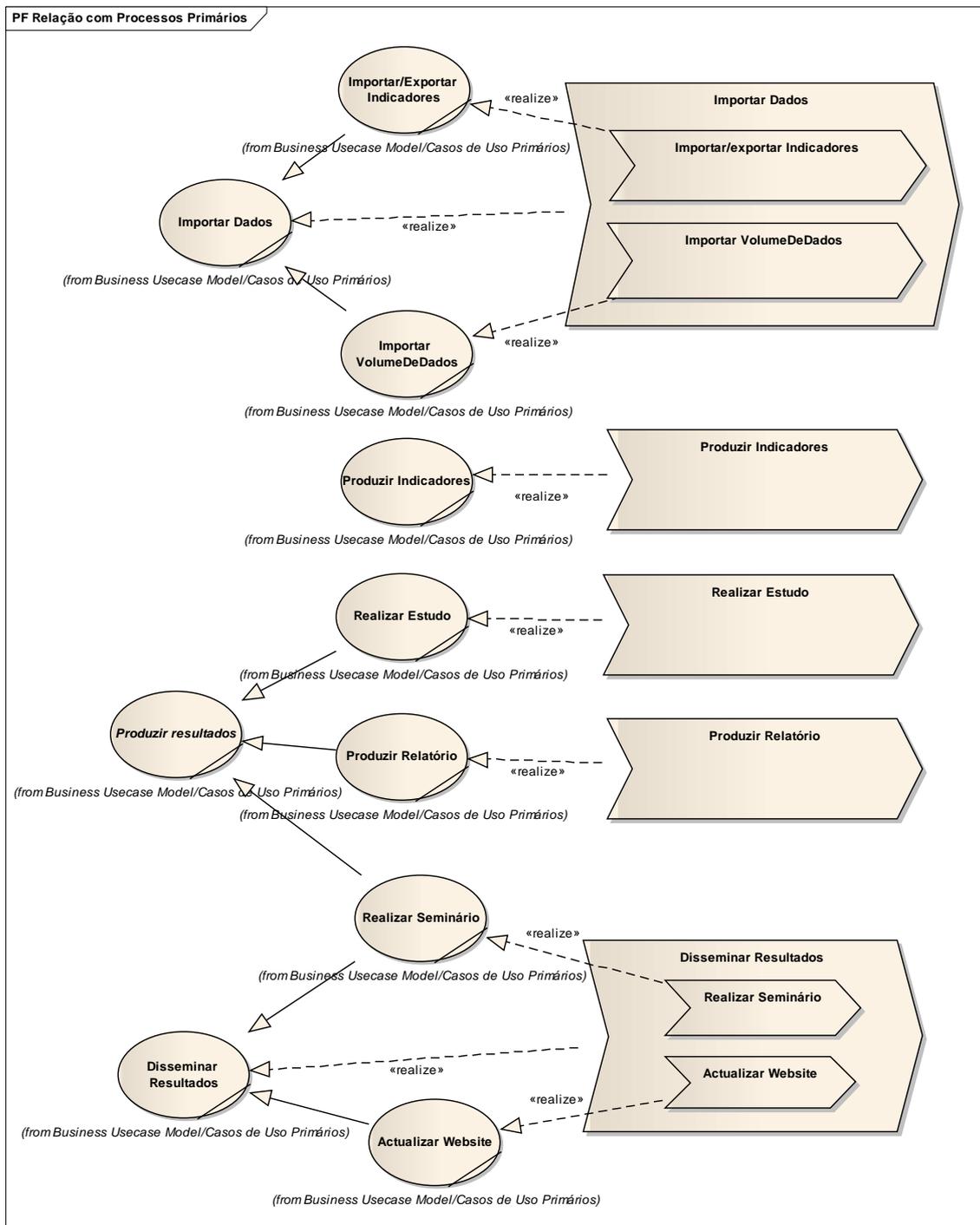
Esta célula pretende capturar os locais relevantes para a organização. Neste caso a representação é simples, dado o Observatório ficar co-localizado numa fase inicial com o IMTT e posteriormente com a Autoridade Metropolitana dos Transportes de Lisboa.

O diagrama proposto pelo Enterprise Architect, usando o perfil Zachman *Framework* intitula-se *Business Locations Diagram*, o qual se assemelha a um diagrama de instalação. Neste caso usa-se uma classe com o estereótipo *Headquarters* para representar a sede do Observatório localizada em Lisboa.

O diagrama associado pode ser observado na Figura 4.6.

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa



**Figura 4.5 - Os processos de negócio primários realizam os casos de uso primários.**

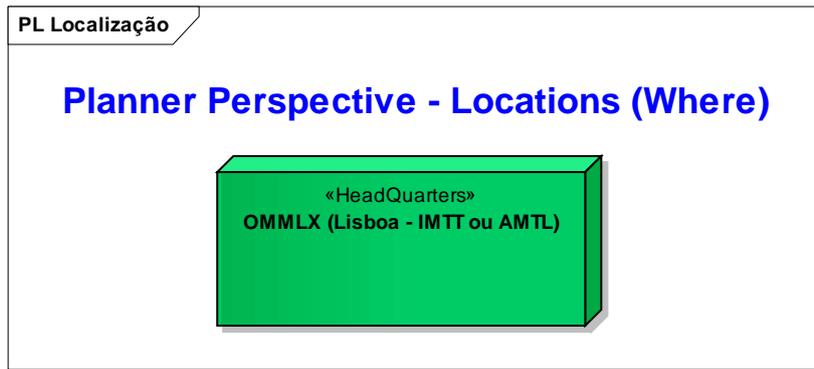


Figura 4.6 - Business Locations Diagram – Localização do Observatório.

#### 4.3.4 Célula D - Lista de organizações relevantes para o Observatório

Esta célula destina-se a capturar as organizações que se relacionam com o Observatório. O diagrama sugerido pelo Enterprise Architect intitula-se *Organization Chart*, o qual é uma variante de um diagrama de casos de uso. Optou-se por usar directamente este último tipo de diagrama.

A Figura 4.7 apresenta os actores relevantes no contexto do Observatório.

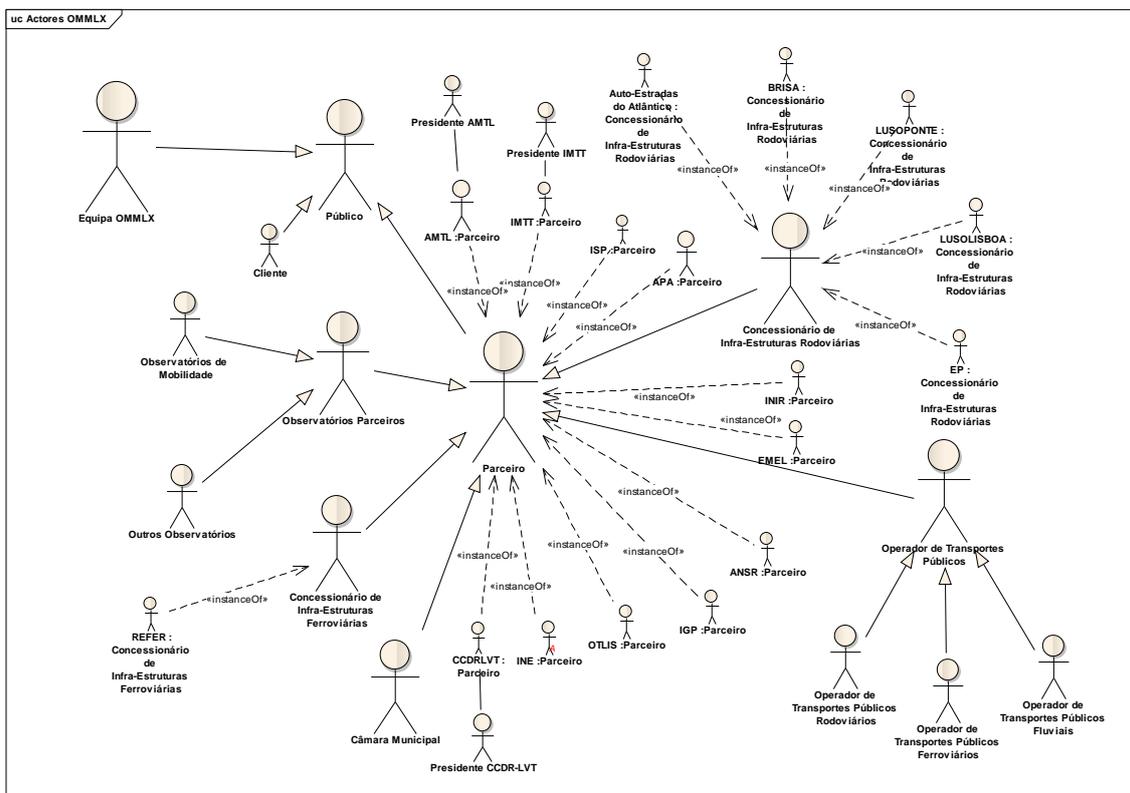


Figura 4.7 - Actores relevantes no contexto do Observatório.

#### 4.3.5 Célula E – Lista de eventos relevantes para a organização

A Sparx propõe um *Business Events Diagram* para capturar os eventos relevantes para a organização. Neste caso concreto do Observatório consideram-se eventos relevantes os seguintes:

- Pedido de Estudo;
- Apresentação de Estudo;
- Início da recolha de informação (para novo relatório anual);
- Publicação do Relatório Anual;
- Realização do Seminário Anual.

Estes eventos foram modelizados recorrendo ao estereótipo *business event* do perfil *Eriksson-Penker Business Extensions* (Eriksson & Penker, 2000). Uns eventos são representados como eventos iniciais e outros como finais.

#### 4.3.6 Célula F – Lista de objectivos e estratégias da organização

O diagrama proposto pelo Enterprise Architect para agregar os artefactos associados a este tipo de informação é um *Business Motivation Diagram*. Trata-se de uma extensão de um diagrama de classes. Nele podemos encontrar uma classe com o estereótipo *Mission*, que descreve nas notas a Missão e a Visão do Observatório. Um documento associado permite obter mais informação sobre a Missão e a Visão, particularmente justificando a sua construção.

No mesmo diagrama foram organizados em pacotes os Objectivos Estratégicos e os Objectivos Estratégicos Indirectos do Observatório. Estes últimos derivam da política de mobilidade, cabendo ao Observatório contribuir para a sua implementação através da monitorização da mesma, graças a um conjunto de indicadores específicos. Podemos também encontrar no mesmo diagrama, agrupados em pacotes, as estratégias associadas a cada um destes tipos de objectivos e os requisitos que originam. O diagrama global pode ser observado na Figura 4.8.

Clicando nos vários pacotes é possível ver os seus elementos em detalhe. Os Objectivos Estratégicos, as Estratégias e os Requisitos associados estão relacionados através de conectores *Realization*, garantindo a concretização do plano estratégico, a rastreabilidade desta concretização e, em última análise, a coerência do seu conteúdo. Esta abordagem será mantida ao longo deste trabalho, permitindo uma validação automática do modelo, tirando partido de uma funcionalidade do Enterprise Architect. Esta abordagem pode ser observada na Figura 4.9.

Nesta fase do trabalho existe apenas rastreabilidade entre objectivos, estratégias e requisitos. Dado que estes últimos ainda não estão satisfeitos por casos de uso e estes por componentes de software, a validação automática do modelo acusa erro nesta fase para todos os requisitos já identificados. Tal pode ser observado no texto seguinte onde são indicados 7 *warnings* associados aos 7 requisitos:

*MVR7F0001 - warning (Criar indicadores específicos com objectivos quantitativos (Requirement)):  
Criar indicadores específicos com objectivos quantitativos is unrealized*

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

MVR7F0001 - warning (Criar mecanismos de troca de informação (Requirement)): Criar mecanismos de troca de informação is unrealized

MVR7F0001 - warning (Elaborar relatório anualmente (Requirement)): Elaborar relatório anualmente is unrealized

MVR7F0001 - warning (Criar website (Requirement)): Criar website is unrealized

MVR7F0001 - warning (Realizar seminário (Requirement)): Realizar seminário is unrealized

MVR7F0001 - warning (Os indicadores deverão estar directamente associados aos objectivos estratégicos directos ou indirectos (Requirement)): Os indicadores deverão estar directamente associados aos objectivos estratégicos is unrealized

MVR7F0001 - warning (A estrutura organizacional deverá ser autónoma (Requirement)): A estrutura organizacional deverá ser autónoma is unrealized

Validation complete - 0 error(s), 7 warning(s)

custom Missão, Visão, Objectivos Estratégicos, Estratégias e Requisitos Estratégicos OMMLX

### Planner Perspective - Motivation (Why)

«Mission»  
Missão e Visão

notes

Missão

Supportar a implementação de uma Estratégia Metropolitana de Mobilidade, de forma sustentável, através de uma avaliação crítica da evolução da mesma, suportada em indicadores gerados com base em informação recolhida junto de todas as fontes relevantes e disseminando publicamente os resultados.

Visão

O Observatório de Mobilidade na Região Metropolitana de Lisboa deverá ser a base técnica para todas as actividades de planeamento, implementação, monitorização, acção correctiva e sensibilização ao nível da Mobilidade nesta região, suportando a implementação de uma Estratégia Metropolitana de Mobilidade e contribuindo decisivamente para a adopção de uma Mobilidade Sustentável.

Clickar com tecla direita e seleccionar Edit Linked Document para aceder a documento anexo

**Objectivos Estratégicos OMMLX**

- + Definir um painel de controlo
- + Criar uma rede de cooperação institucional
- + Publicar um relatório periódico de análise crítica da Mobilidade
- + Desenvolver uma estratégia de disseminação dos resultados
- + Desenvolver a operação do Observatório de forma sustentável
- + Assegurar a autonomia institucional do Observatório

**Estratégias OMMLX**

- + Basear a rede na troca de informação
- + Criar estrutura autónoma
- + Definir um conjunto de indicadores e objectivos quantitativos
- + Publicar relatório anual
- + Disseminar resultados via website e acções presenciais
- + Minimizar número de indicadores e maximizar informatização
- + A estrutura organizacional deverá ser autónoma
- + Criar mecanismos de troca de informação
- + Criar website
- + Elaborar relatório anualmente
- + Os indicadores deverão estar directamente associados aos objectivos estratégicos directos ou indirectos
- + Realizar seminário

**Objectivos Estratégicos Indirectos OMMLX**

- + Contenção do crescimento em mancha-de-óleo na "cidade-região"
- + Diminuição dos níveis de poluição
- + Melhoria gradual das motorizações
- + Reduzir níveis de incumprimento da qualidade do ar
- + Reduzir a idade média dos veículos
- + Diversificar as fontes de energia utilizadas no sector
- + Promover a transferência modal
- + Monitorizar os gastos colectivos em energia
- + Aumentar a quota de veículos movidos a energias alternativas
- + Resolução de congestionamento
- + Doseamento da oferta dos diferentes tipos de estacionamento
- + Diminuição do tempo médio de viagem
- + Diminuição da população que normalmente não efectua viagens
- + Equilibrar a repartição modal entre modos
- + Aumentar a integração entre meios
- + Diminuição dos percursos em transporte individual
- + Diminuição drástica do nível de sinistralidade rodoviária
- + Melhoria da qualidade de vida
- + Comparar a evolução com outras "cidades-região"

**Estratégias Indirectas OMMLX**

- + Criar indicador ou indicadores específicos
- + Realizar benchmarking
- + Criar indicadores específicos com objectivos quantitativos
- + Incorporar benchmarking com outras cidades-região

Figura 4.8 - Missão, Visão, Objectos Estratégicos e Estratégias.

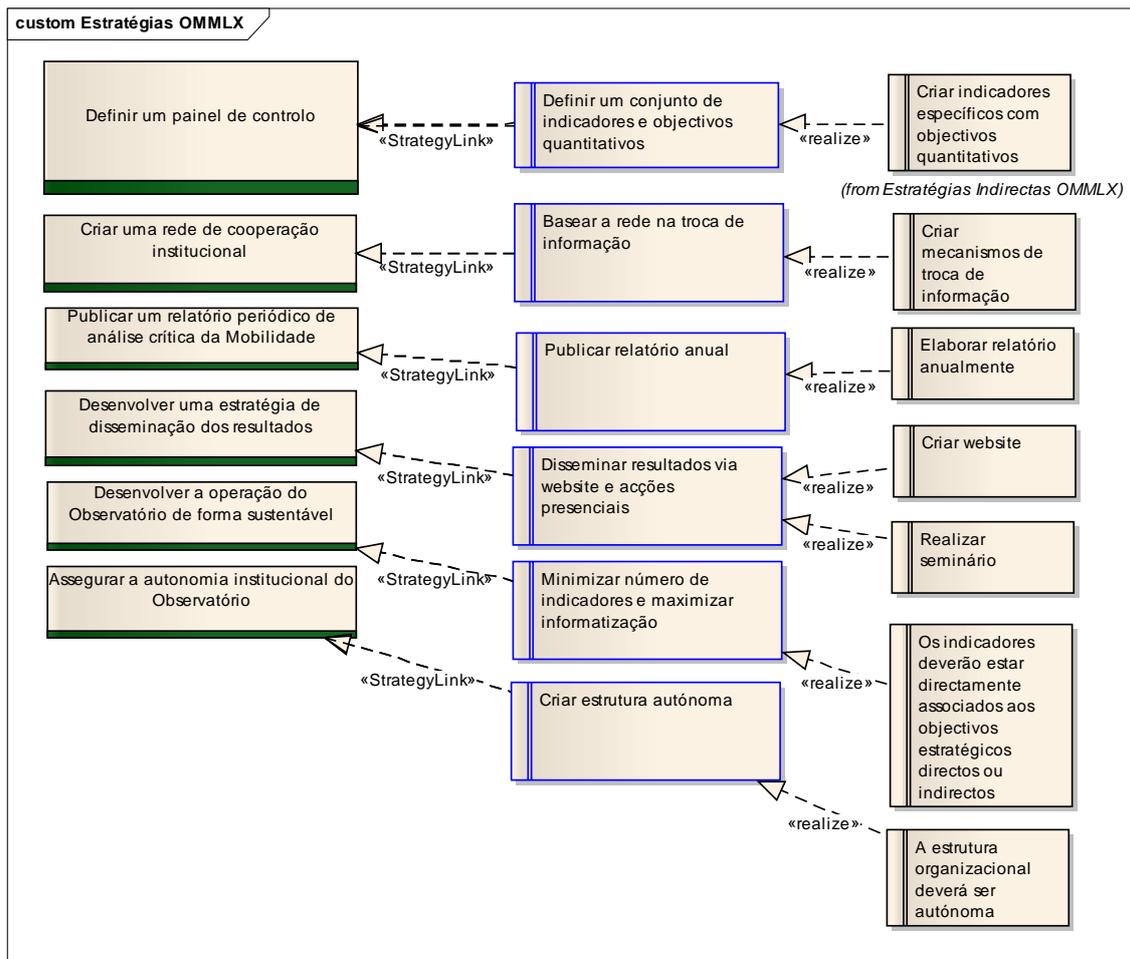


Figura 4.9 - Relação entre objectivos, estratégias e requisitos.

Na Figura 4.10 podemos observar as estratégias indirectas. Neste caso existem apenas 2 *warnings* associados aos 2 requisitos não satisfeitos:

MVR7F0001 - warning (Criar indicadores específicos com objectivos quantitativos (Requirement)):  
Criar indicadores específicos com objectivos quantitativos is unrealized  
MVR7F0001 - warning (Incorporar benchmarking com outras cidades-região (Requirement)):  
Incorporar benchmarking com outras cidades-região is unrealized  
Validation complete - 0 error(s), 2 warning(s)

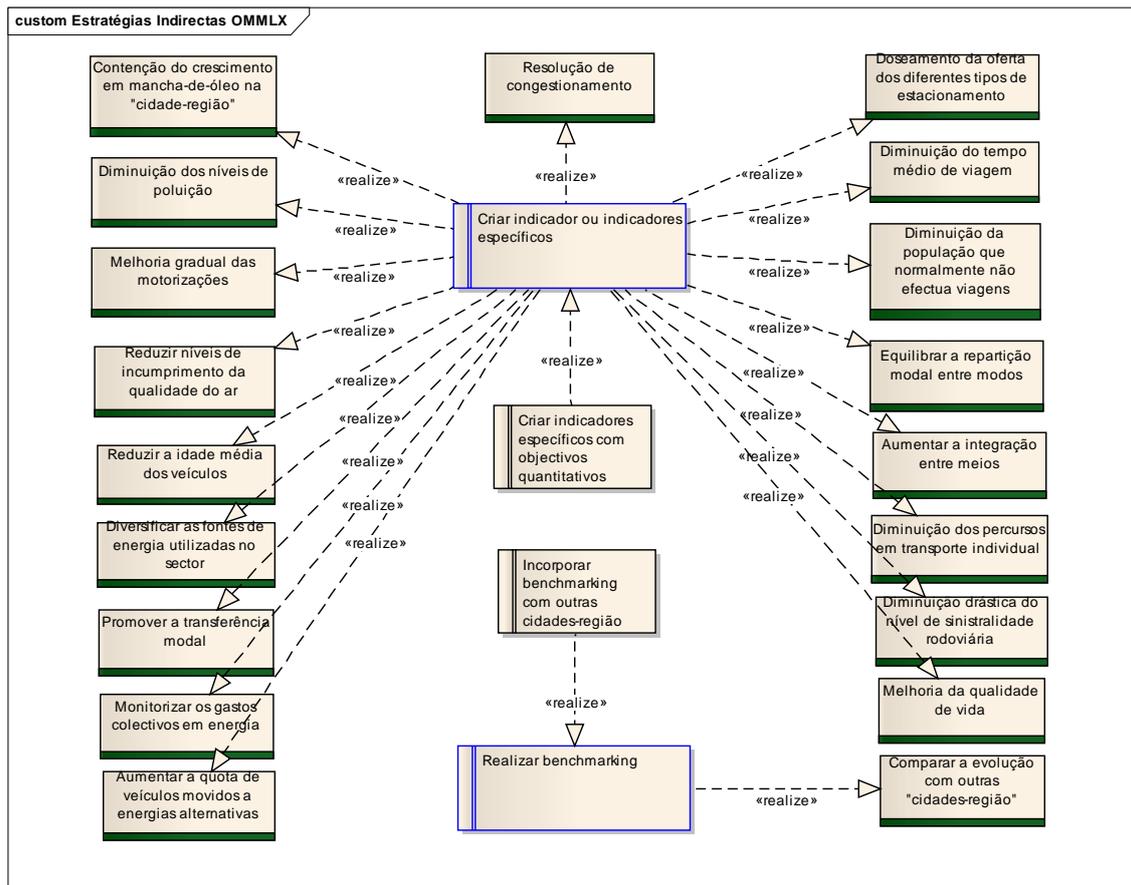
## 4.4 Fase 2 – Célula G

### 4.4.1 Célula G – Modelo semântico

O *Data Map Diagram* proposto pela Sparx para capturar a informação referente ao modelo semântico, corresponde a um diagrama de classes com as entidades informacionais do negócio e os actores que as manipulam. Esta informação complementa-se com um dicionário das entidades informacionais.

## Definição da Arquitectura Empresarial

### O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

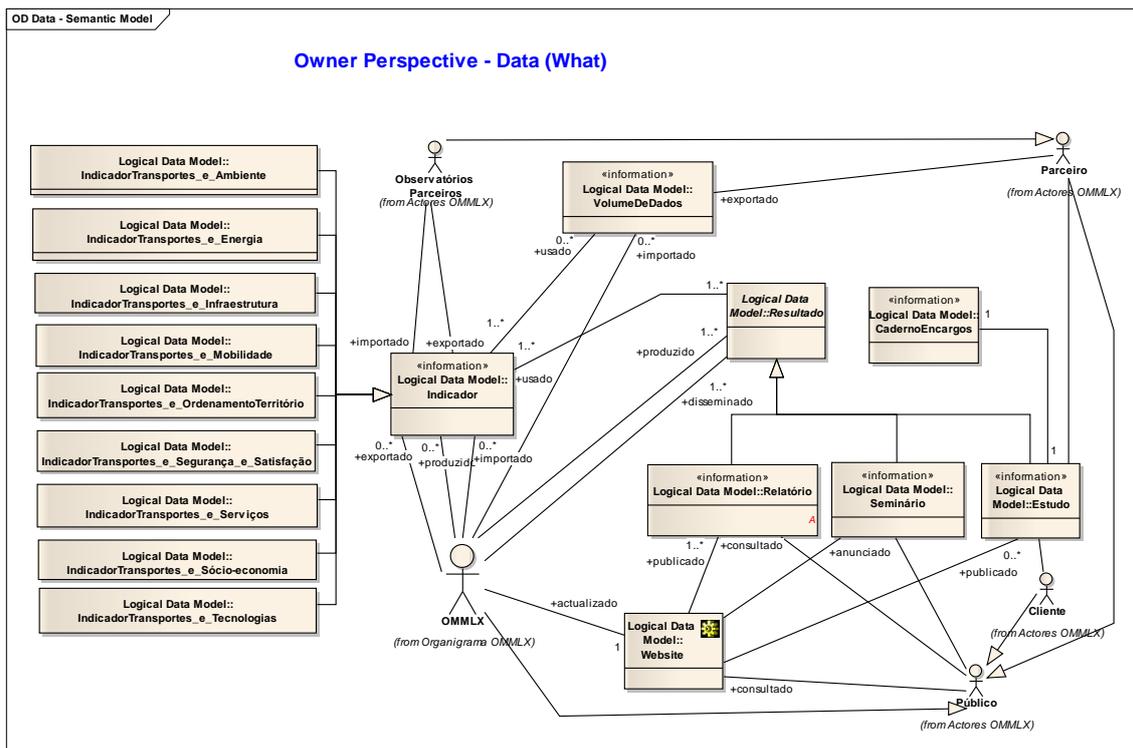


**Figura 4.10 - Relação entre objectivos indirectos, estratégias e requisitos.**

Na Figura 4.11 podemos observar o modelo semântico, onde se podem visualizar as principais entidades informacionais do Observatório. É dado destaque à ontologia de indicadores, com as suas sub-classes temáticas, a qual será descrita em detalhe mais abaixo nesta secção. São capturados os papéis que estas entidades desempenham ao interagirem com os diversos actores, bem como as multiplicidades associadas a algumas associações. Por exemplo, um indicador poderá ser importado de um Observatório parceiro ou então produzido, usando volumes de dados. Um indicador poderá ser exportado para observatórios parceiros. Os indicadores são usados na produção de resultados. Estes resultados produzidos e disseminados pelo Observatório, podem tomar a forma de relatório anual, de seminário anual ou de estudos avulsos a pedido de clientes e parceiros. Os dois primeiros resultados são de natureza pública e destinam-se ao público em geral, embora interessem mais à comunidade de especialistas. Os volumes de dados são tipicamente fornecidos por parceiros consistindo em informação em bruto, que será posteriormente processada pelo Observatório. Os observatórios parceiros entregam tipicamente informação já processada, na forma de indicadores.

## Definição da Arquitectura Empresarial

### O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa



**Figura 4.11 - Modelo semântico.**

No campo das multiplicidades das associações salientam-se algumas situações. Um volume de dados, caso exista devido a importação, está sempre associado a um ou mais indicadores. No entanto, um indicador poderá não estar associado a um volume de dados, pois poderá ser importado directamente. Os resultados irão usar pelo menos um indicador e o Observatório terá de produzir, pelo menos, um resultado e deverá disseminá-lo.

Seguidamente descrevem-se em detalhe as diversas entidades informacionais. Na primeira foram usados nos atributos os metadados do Dublin Core.

<b>Entidade:</b>	E01 – Relatório
<b>Identificação:</b>	Código do Relatório (Identifier)
<b>Objectivo:</b>	Registrar a informação associada aos relatórios anuais produzidos pelo Observatório.
<b>Atributos:</b>	Title Date Creator Subject Description Publisher Contributor Type Format Source Language Relation Rights Coverage

Definição da Arquitectura Empresarial  
O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

<b>Entidade:</b>	E02 – Estudo
<b>Identificação:</b>	Código de estudo
<b>Objectivo:</b>	Registar a informação associada aos estudos produzidos pelo Observatório.
<b>Atributos:</b>	Título Data Cliente Caderno de Encargos Equipa Ficheiro

<b>Entidade:</b>	E03 – Seminário
<b>Identificação:</b>	Ano
<b>Objectivo:</b>	Registar a informação referente à organização e realização do seminário anual do Observatório.
<b>Atributos:</b>	Título Data Local Programa Ficheiro com apresentações Palestrante Participante Orçamento Actas das reuniões preparatórias

<b>Entidade:</b>	E04 – VolumeDeDados
<b>Identificação:</b>	URI de volume de dados
<b>Objectivo:</b>	Registar a informação referente a um volume de dados importado de um parceiro do Observatório.
<b>Atributos:</b>	Title Date Creator Subject Description Publisher Contributor Type Format Source Language Relation Rights Recolha Coverage Periodicidade ValorVolumeDeDados

Definição da Arquitectura Empresarial  
O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

<b>Entidade:</b>	E05 – Indicador
<b>Identificação:</b>	URI do indicador
<b>Objectivo:</b>	Registar a informação referente aos indicadores importados ou produzidos pelo Observatório.
<b>Atributos:</b>	Title Date Creator Subject Description Publisher Contributor Type Format Source Language Relation Rights Recolha Coverage Periodicidade Unidade Importação Método ValorIndicador

<b>Entidade:</b>	E06 – Cliente
<b>Identificação:</b>	NIF
<b>Objectivo:</b>	Registar os dados associados aos clientes
<b>Atributos:</b>	Nome Morada Contacto Estudo

<b>Entidade:</b>	E07 – Parceiro
<b>Identificação:</b>	NIF
<b>Objectivo:</b>	Registar os dados associados aos parceiros
<b>Atributos:</b>	Nome Morada Contacto Estudo VolumeDeDados

<b>Entidade:</b>	E08 – Observatório Parceiro
<b>Identificação:</b>	NIF
<b>Objectivo:</b>	Registar os dados associados aos observatórios parceiros
<b>Atributos:</b>	Nome Morada Contacto Indicador importado Indicador exportado

<b>Entidade:</b>	E09 – Website
<b>Identificação:</b>	URL
<b>Objectivo:</b>	Registar os dados associados ao website
<b>Atributos:</b>	Empresa criadora Morada empresa criadora Contacto empresa criadora ISP (Internet Service Provider) Morada ISP Contacto ISP Contrato ISP Utilizadores Log Ficheiros versões

<b>Entidade:</b>	E10 – CadernoEncargos
<b>Identificação:</b>	Código de Estudo
<b>Objectivo:</b>	Registar a informação associada ao caderno de encargos
<b>Atributos:</b>	Título Objectivo Prazo Orçamento Ficheiro

### A ontologia de indicadores

A entidade informacional mais importante é o indicador. É em torno dos indicadores que o Observatório opera, importando-os de outros parceiros ou calculando-os a partir de volumes de dados cedidos por parceiros. A análise da mobilidade regional é feita com base nos indicadores, sustentando o relatório anual, o seminário e apoiando estudos avulsos que sejam encomendados por clientes ou parceiros.

Dada a interligação existente, nomeadamente na troca de indicadores com parceiros e observatórios parceiros é importante que o modelo semântico seja partilhado pela comunidade ou seja, que o conhecimento do domínio seja partilhado e entendido pela comunidade. Esta questão é particularmente fundamental ao nível dos indicadores, pois só desta forma será possível garantir a interoperação semântica, permitindo que sejam posteriormente manipulados de forma adequada (adição simples, temporal ou geográfica, por exemplo).

A solução identificada para a resolução deste problema foi a criação de uma Ontologia de Indicadores, estruturando e permitindo a partilha de conhecimento neste campo. A ontologia foi criada usando uma ferramenta de edição de ontologias de forma a permitir uma edição, publicação e exportação simplificada. A ferramenta usada foi o Protégé 3.4.1.

Um dos primeiros passos na construção de uma ontologia é tentar re-utilizar ontologias já existentes (Palazzo, 2006). Infelizmente não foi identificada nenhuma ontologia na área da mobilidade com os conceitos pretendidos, apenas ontologias na área dos transportes ou da mobilidade computacional. A busca foi feita usando motores de busca tradicionais e motores de busca de ontologias, como o Swoogle (Swoogle, 2009). No entanto ao nível dos metadados foi possível utilizar o Dublin Core, na sua versão 1.1, para caracterizar os Indicadores. De forma a salientar esta re-utilização os atributos da classe Indicador obtidos a partir do Dublin Core não foram traduzidos, mantendo-se em inglês.

Na Figura 4.12 é possível observar uma vista geral da ontologia, com a hierarquia de classes de indicadores de mobilidade e com as diversas instâncias de indicadores. Note-se que está seleccionada a sub-classe referente aos indicadores de transportes e sócio-economia, permitindo visualizar as instâncias, ou melhor, os 5 indicadores desta sub-classe. Para visualizar a ontologia foi utilizado o *plug-in* Jambalaya para visualização gráfica de ontologias no Protégé.

Na Figura 4.13 é possível observar novamente a estrutura da ontologia de indicadores, com as sub-classes, salientando-se a visualização de parte dos atributos da classe indicador e de algumas restrições associadas. Note-se também que foi re-utilizada a ontologia para metadados Dublin Core 1.1, estando os atributos associados em inglês e começando por “dc:”. É o caso do atributo “formato” (dc:format), o qual tem o tipo *string* e uma restrição de cardinalidade de 1, ou seja, cada indicador tem apenas um formato. Outra restrição é o facto de “formato” ter o valor “text/xml”, o valor considerado mais adequado seguindo a recomendação do Dublin Core e usando o vocabulário controlado de tipos de media para a Internet criado pela IANA (Iana, 2009).

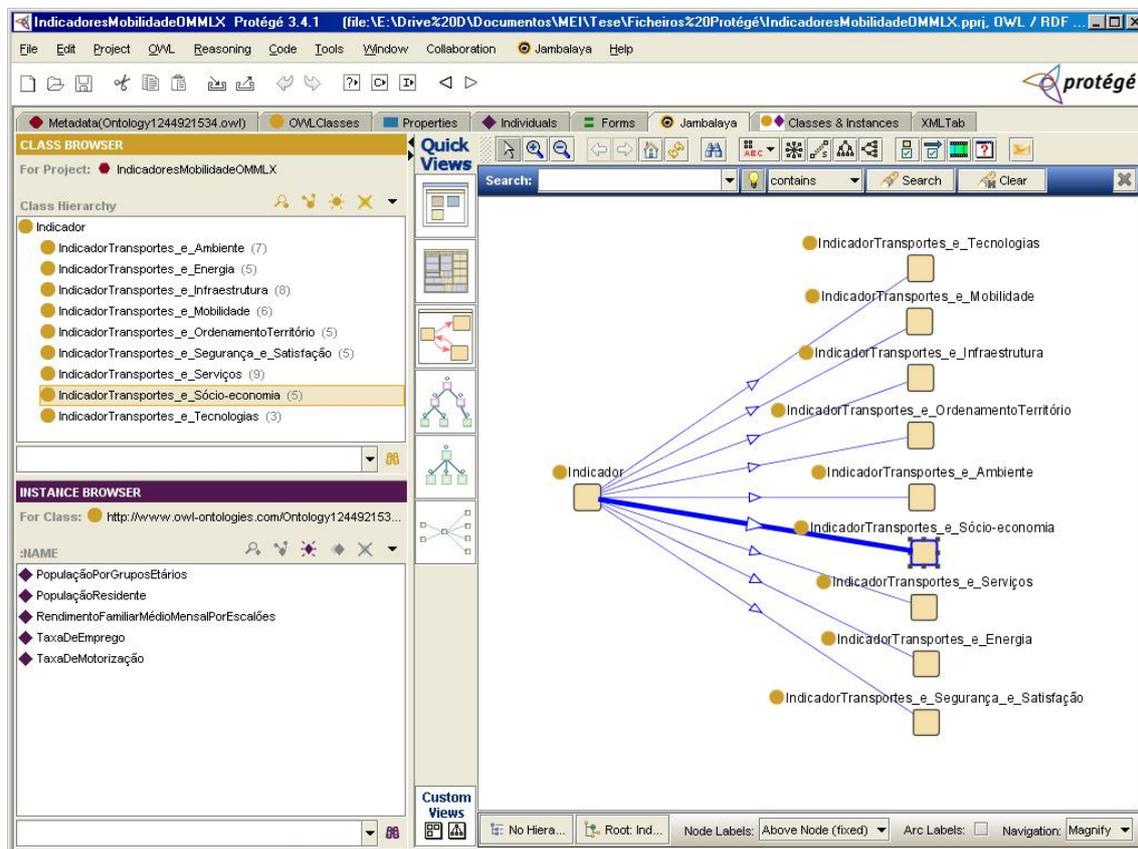


Figura 4.12 - Vista geral da ontologia de indicadores.

## Definição da Arquitectura Empresarial

### O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

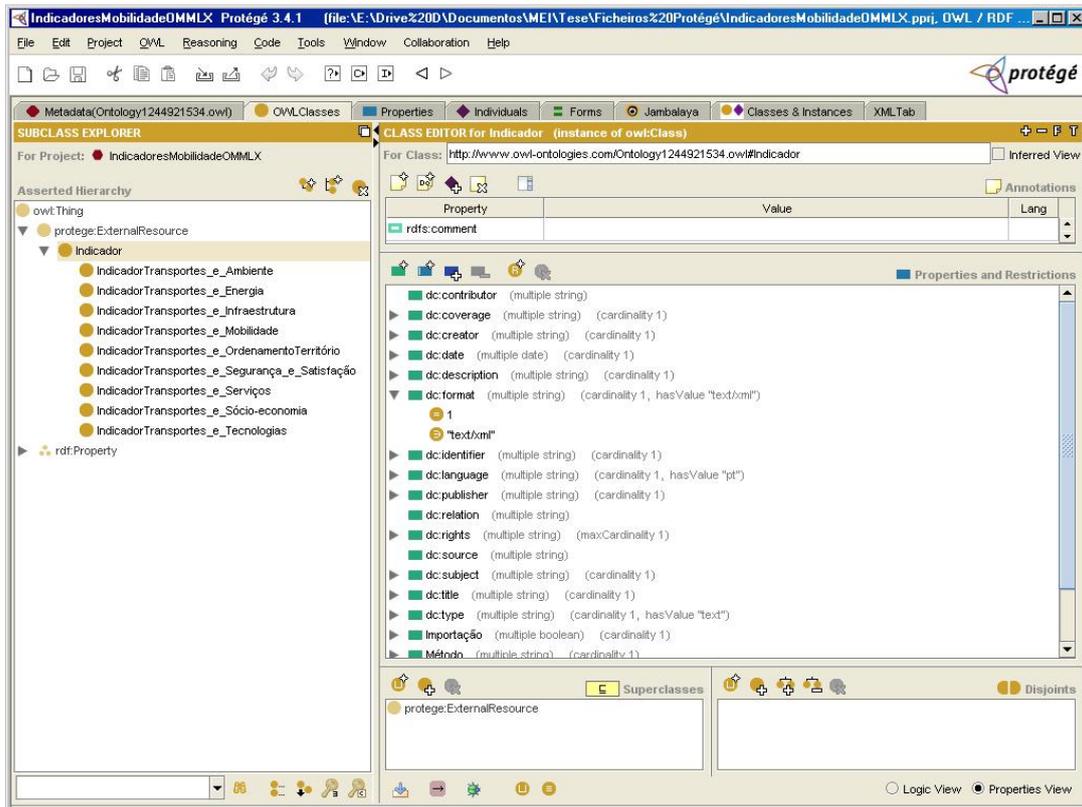


Figura 4.13 - Ontologia de indicadores com sub-classes e atributos.

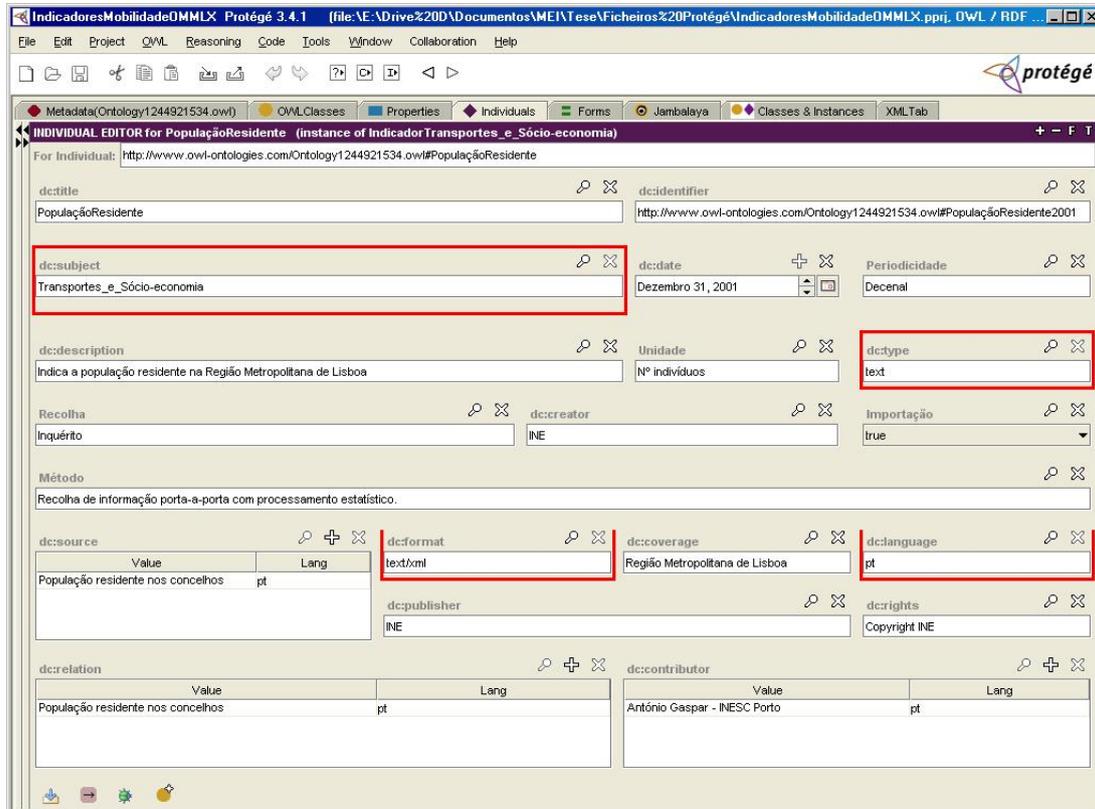


Figura 4.14 - Formulário criado para carregamento dos dados associados ao indicador População Residente.

O Protégé permite criar formulários para carregamento das instâncias da ontologia. Na Figura 4.14 pode ser observado um exemplo de formulário criado para carregamento dos dados associados ao indicador População Residente, da sub-classe Transportes e Sócio-economia.

Dado não existir uma solução óptima para a conversão do Protégé para o Enterprise Architect seguiu-se uma abordagem que permitisse maximizar a informação convertida.

O primeiro passo consistiu na exportação a partir do Protégé para o Enterprise Architect em UML 1.4. Este passo permitiu a conversão da classe indicador com todos os seus atributos e das sub-classes. As instâncias e as generalizações perderam-se. O resultado pode ser observado na Figura 4.15. Com este passo pretende-se capturar a riqueza de atributos da superclasse.

O segundo passo consistiu na importação pelo OWL Editor do ficheiro OWL produzido pelo Protégé. Desta forma foi possível obter a estrutura da ontologia em termos de classes e sub-classes, com as respectivas generalizações. São também obtidas as diversas instâncias, embora sem associação às sub-classes de indicadores. Finalmente, são também obtidos os atributos da classe indicador como classes individuais, mas estes já foram correctamente obtidos no passo anterior.



Figura 4.15 - Passo 1: Exportação do Protégé para Enterprise Architect usando UML 1.4.

O terceiro passo consiste na integração destes dois pacotes recorrendo a processamento manual. Do pacote importado via UML 1.4 é usada apenas a classe indicador com a totalidade dos seus atributos. Do pacote importado via OWL Editor usa-se a estrutura de

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

classes, eliminam-se os atributos da classe indicador (que tinham sido convertidos em classes), eliminam-se algumas classes do metamodelo que também foram importadas e, finalmente, convertem-se as classes correspondentes às instâncias nas ditas instâncias, associando-as às respectivas sub-classes de indicadores, de acordo com a temática. Os nomes foram simplificados, dado incorporarem URIs demasiado extensos.

Na Figura 4.16 pode ser visto um aspecto geral da ontologia. Na Figura 4.17 pode ser observado com detalhe um dos ramos, respeitante à sub-classe de indicadores com a temática Transportes e Energia, onde podem ser visualizados os 5 indicadores (objectos) associados.

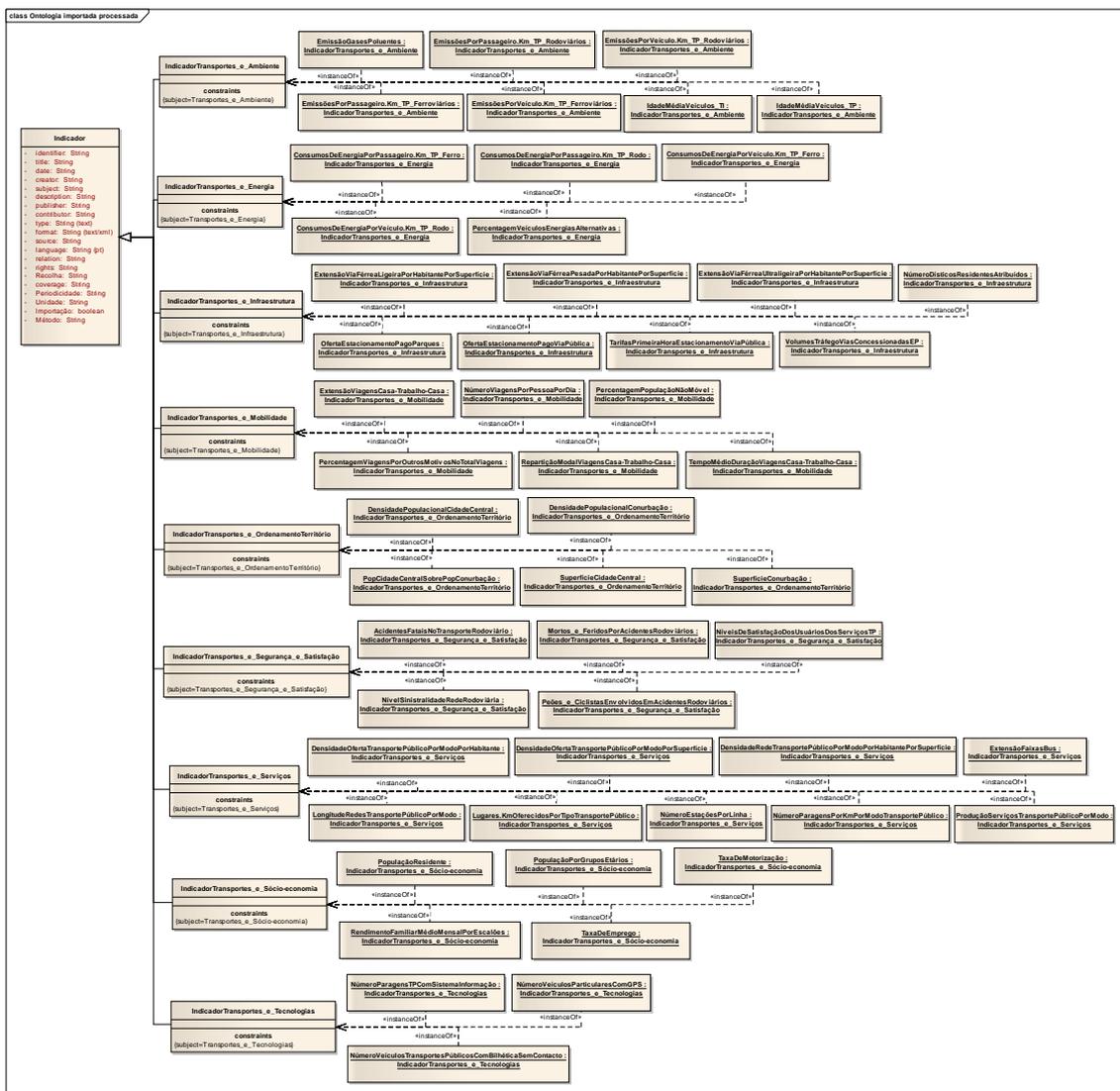


Figura 4.16 - Aspecto geral da ontologia de indicadores após processamento manual.

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

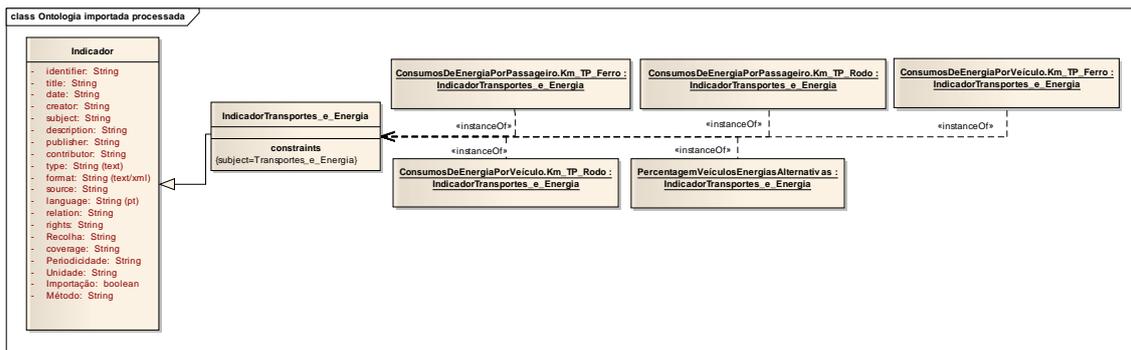


Figura 4.17 - Indicadores com a temática Transportes e Energia.

### 4.5 Fase 3 - Células H, M

#### 4.5.1 Célula H – Modelo de processos da organização

A informação desta célula é capturada através de um *Process Analysis Diagram*. Trata-se de um diagrama que permite visualizar os diversos processos desenvolvidos pela organização. Neste caso são apresentados apenas os que constituem os processos primários do Observatório e que o distinguem de outras organizações.

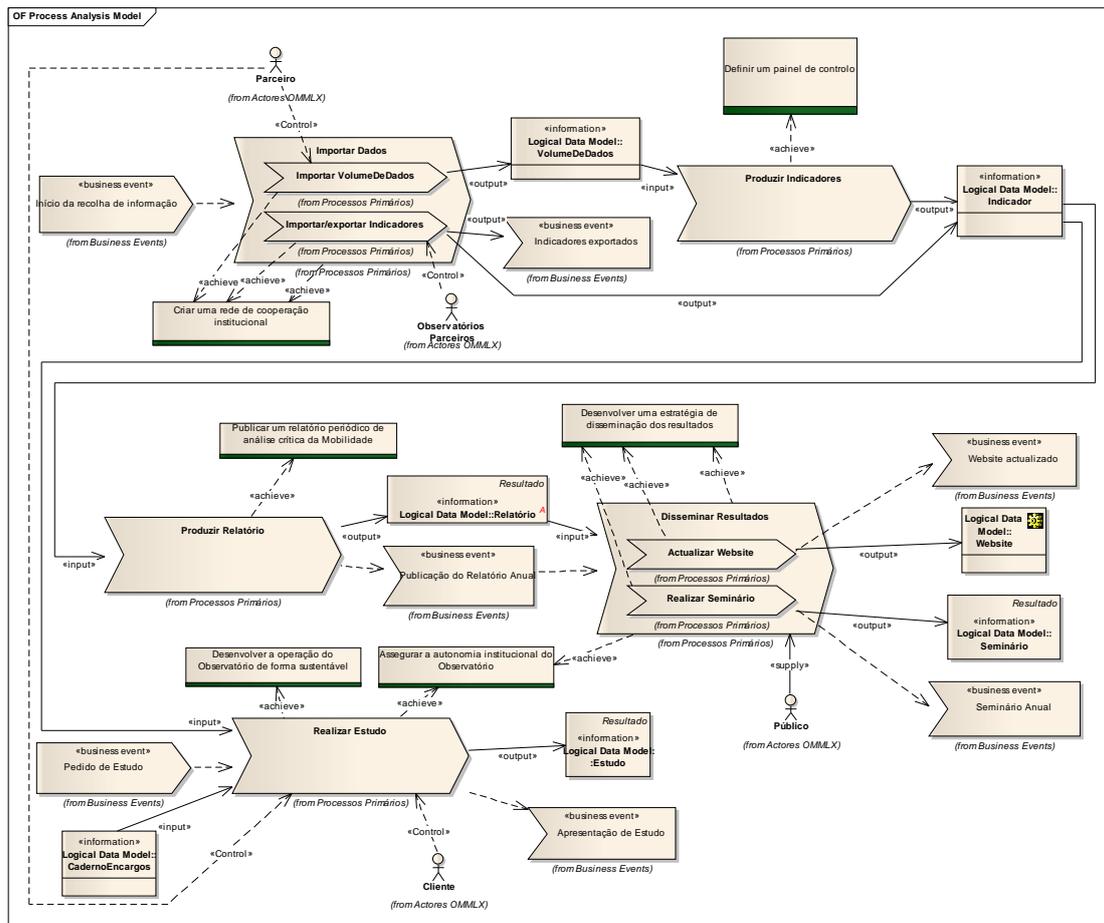


Figura 4.18 - Diagrama de processos de negócio primários.

Esta análise será complementada com a elaboração de uma matriz de entidades informacionais versus processos de negócio, com o objectivo de identificar aplicações ou módulos aplicativos. A descrição dos processos de negócio num dicionário complementa esta célula. Esta informação está residente também nas notas do Enterprise Architect associadas a estes elementos. Os processos de negócio não são detalhados ao nível de diagramas de actividades em UML ou BPMN (Business Process Modeling Notation) devido não existir informação suficiente.

O *Process Analysis Diagram* pode ser visualizado na Figura 4.18. Note-se que foi usada a notação proposta por Eriksson & Penker (2000). O potencial do repositório fica evidente ao combinarmos entidades informacionais (objectos do negócio ou actores), processos, eventos e objectivos estratégicos numa vista complexa que retrata o funcionamento da organização. Os perfis dos actores internos ao Observatório não são representados para facilitar a visualização do diagrama, pois relacionam-se com muitos processos criando muitas linhas. Esta relação é evidenciada na matriz entre processos de negócio e organização, na célula J.

Os detalhes sobre os diferentes processos são seguidamente indicados. Os macro-processos Importar Dados e Disseminar Resultados não são descritos, apenas os seus sub-processos.

<b>Processo:</b>	P01 - Importar VolumeDeDados
<b>Objectivos:</b>	Importar volumes de dados de parceiros
<b>Descrição:</b>	Este processo importa volumes de dados de parceiros que irão posteriormente ser usados para calcular indicadores.
<b>Entradas:</b>	Início da recolha de informação (evento)
<b>Saídas:</b>	VolumeDeDados
<b>Participantes:</b>	Parceiro, Núcleo Técnico
<b>Indicadores:</b>	Número de volumes de dados cedidos por parceiro Número de volumes de dados usados

<b>Processo:</b>	P02 – Importar/exportar Indicadores
<b>Objectivos:</b>	Importar e exportar indicadores para observatórios parceiros
<b>Descrição:</b>	Este processo importa e exporta indicadores de e para observatórios parceiros, de forma automática, recorrendo a uma ontologia de indicadores, um repositório de indicadores, com operacionalização através de <i>web services</i> .
<b>Entradas:</b>	Início da recolha de informação (evento)
<b>Saídas:</b>	Indicadores, Indicadores exportados (evento)
<b>Participantes:</b>	Observatórios parceiros, Núcleo Técnico
<b>Indicadores:</b>	Número de indicadores exportados para cada observatório parceiro Número de indicadores importados de cada observatório parceiro

<b>Processo:</b>	P03 – Produzir Indicadores
<b>Objectivos:</b>	Produzir indicadores a partir de volumes de dados importados
<b>Descrição:</b>	Este processo destina-se a produzir indicadores de mobilidade a partir de volumes de dados importados de parceiros.
<b>Entradas:</b>	VolumeDeDados
<b>Saídas:</b>	Indicador
<b>Participantes:</b>	Núcleo Técnico
<b>Indicadores:</b>	Número de indicadores produzidos Tempo de produção de cada indicador

Definição da Arquitectura Empresarial  
O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

<b>Processo:</b>	P04 – Realizar Estudo
<b>Objectivos:</b>	Realizar um estudo sobre mobilidade
<b>Descrição:</b>	Este processo suporta a realização de estudos de mobilidade a pedido de um cliente ou parceiro, com base na apresentação de um caderno de encargos.
<b>Entradas:</b>	Caderno Encargos, Pedido de Estudo (evento), Indicador
<b>Saídas:</b>	Estudo, Apresentação de Estudo (evento)
<b>Participantes:</b>	Cliente, Parceiro, Núcleo Técnico, Direcção, Consultor Externo
<b>Indicadores:</b>	Valor dos estudos produzidos anualmente Número de estudos por cliente/parceiro Valor de cada estudo

<b>Processo:</b>	P05 – Produzir Relatório
<b>Objectivos:</b>	Produzir o relatório anual sobre mobilidade
<b>Descrição:</b>	Este processo produz o relatório anual sobre mobilidade na Região Metropolitana de Lisboa, com base nos indicadores produzidos ou importados pela equipa do Observatório.
<b>Entradas:</b>	Indicador
<b>Saídas:</b>	Relatório, Publicação do Relatório Anual (evento)
<b>Participantes:</b>	Núcleo Técnico, Direcção, Secretariado, Consultor Externo
<b>Indicadores:</b>	Tempo de produção do relatório Número de indicadores incluídos

<b>Processo:</b>	P06 – Realizar Seminário
<b>Objectivos:</b>	Realizar seminário anual sobre mobilidade
<b>Descrição:</b>	Este processo destina-se a suportar a organização e a realização de um seminário anual sobre mobilidade, onde são apresentados os resultados da actividade do Observatório e trocadas experiências com peritos externos e com o público em geral.
<b>Entradas:</b>	Relatório, Publicação do Relatório Anual (evento)
<b>Saídas:</b>	Seminário, Seminário Anual (evento)
<b>Participantes:</b>	Direcção, Secretariado, Público
<b>Indicadores:</b>	Orçamento por edição Número de participantes por edição

<b>Processo:</b>	P07 – Actualizar Website
<b>Objectivos:</b>	Actualizar o website do Observatório
<b>Descrição:</b>	Este processo destina-se a suportar a actualização do website, com os resultados da actividade do Observatório.
<b>Entradas:</b>	Relatório, Publicação do Relatório Anual (evento)
<b>Saídas:</b>	Website, Website actualizado (evento)
<b>Participantes:</b>	Núcleo Técnico, Secretariado, Público
<b>Indicadores:</b>	Número de actualizações por ano Número de visitas ao website

De forma a identificarmos potenciais aplicações informáticas podemos nesta fase elaborar uma matriz de CRUD (Create-Read-Update-Delete) entre as entidades informacionais e os processos de negócio, neste caso apenas primários. Esta matriz foi elaborada usando o EXCEL, dado as funcionalidades de manipulação de matrizes do Enterprise Architect serem limitadas. Para simplificar a notação considera-se que quando há *read* e *create* simultaneamente, se deve registar apenas *create*.

A primeira fase da elaboração desta matriz pode ser observada na Figura 4.19.

Matriz Entidades Informacionais vs Processos de Negócio	Relatório	Estudo	Seminário	VolumeDeDados	Indicador	Cliente	Parceiro	Observatório Parceiro	Website	CadernoEncargos
<i>Importar VolumeDeDados</i>				C			R			
<i>Importar/exportar indicadores</i>				U						
Produzir indicadores				R	C			R		
Realizar estudo	R	C	R	R	R	C	R			C
Produzir relatório	C				R					
<i>Realizar seminário</i>	R	R	C			R	R	R		
<i>Actualizar website</i>	R	R	R						C	
									U	
C - Create										
R - Read										
U - Update										
D - Delete										

Figura 4.19 - Matriz CRUD – Entidades Informacionais vs Processos de Negócio.

Note-se que são apresentados os processos mais elementares. Alguns são sub-processos de processos ou macro-processos mais complexos. Por exemplo, Importar VolumeDeDados e Importar/exportar indicadores são subprocessos de Importar dados. Quando se representam os sub-processos não são representados os processos que os incluem.

Cada entidade informacional só pode ser criada por um processo de negócio, caso contrário teremos problemas de coerência de dados. No caso da entidade informacional indicador verifica-se que esta é criada pelo processo Importar/exportar indicadores e pelo processo Produzir indicadores. No entanto uma análise mais cuidada permite-nos afirmar que são situações disjuntas: um indicador que seja importado não faz parte do grupo de indicadores produzidos internamente.

Verifica-se também que existem duas entidades informacionais que não são criadas por nenhum processo: Parceiro e Observatório Parceiro. Assume-se que dado estarmos apenas a representar processos primários, estas entidades serão criadas por processos de suporte não representados.

Outra anomalia reside na verificação que nenhuma entidade informacional é apagada. Tal justifica-se por se pretender apresentar séries temporais com a maior duração possível. Quanto aos actores, estes são entidades com uma vida longa: tipicamente operadores de transportes públicos, com concessões bastante dilatadas no tempo. No entanto facilmente poderíamos considerar a possibilidade de *delete*, nas mesmas células onde se encontram os *create*, mas tal teria um significado prático limitado.

A manipulação das linhas e das colunas desta matriz, tentando “clusterizar” os *creates* numa cascata, permite identificar as aplicações necessárias. O resultado desta manipulação pode ser observado na figura seguinte.

Matriz Entidades Informacionais vs Processos de Negócio	VolumeDeDados	Indicador	Observatório Parceiro	Parceiro	Estudo	CadernoEncargos	Cliente	Relatório	Seminário	Website
<i>Importar VolumeDeDados</i>	C U			R						
<i>Importar/exportar indicadores</i>		C	R							
Produzir indicadores	R	C								
Realizar estudo	R	R		R	C	C	C	R	R	
Produzir relatório		R						C		
<i>Realizar seminário</i>			R	R	R		R	R	C	
<i>Actualizar website</i>					R			R	R	C U
<b>C - Create</b>										
<b>R - Read</b>										
<b>U - Update</b>										
<b>D - Delete</b>										

Figura 4.20 - Matriz CRUD – Entidades Informacionais vs Processos de Negócio (após manipulação).

Como se pode verificar estão identificados alguns clusters de processos de negócio, que irão constituir o núcleo de aplicações específicas.

O primeiro *cluster* corresponde a um grupo de processos associados à interoperabilidade e à produção dos indicadores. A inclusão da produção dos indicadores justifica-se por garantir a coerência dos dados, face a potenciais conflitos com a importação. Pretende-se maximizar a automatização deste processo. Os membros são os seguintes:

- Importar VolumeDeDados;
- Importar/exportar indicadores;
- Produzir indicadores.

O segundo *cluster* suporta apenas o processo Realizar Estudo. Trata-se de um processo bastante especializado e sem sinergias com os restantes. A automatização é limitada. Os estudos poderão diferir substancialmente uns dos outros.

O terceiro *cluster* enquadra o processo Produzir Relatório. Trata-se de um processo complexo, envolvendo componentes técnicas especializadas e de processamento de texto e de gráficos. Terá uma natureza repetitiva, mas a sua automatização será limitada.

O quarto *cluster* suporta o processo Realizar Seminário. Trata-se de um processo cujo objectivo é a realização de um evento técnico. A sua natureza é distinta dos restantes. As perspectivas de automatização também são limitadas.

O quinto *cluster* é responsável pelo processo Actualizar Website. Trata-se de um processo tipicamente manual com a publicação de novos conteúdos e a publicitação de eventos à medida que se perspectivam.

### 4.5.2 Célula M – Modelo lógico de dados

O modelo lógico de dados é capturado usando um diagrama de classes. Trata-se de uma versão mais rica dos anteriores diagramas de classes referentes às entidades informacionais. Alguns pormenores de implementação são também retratados, nomeadamente a separação entre estrutura da entidade e respectivo valor. Trata-se do caso dos pares de classes Indicador e ValorIndicador e VolumeDeDados e ValorVolumeDeDados. Note-se que também foram usados metadados genéricos da ontologia Dublin Core, versão 1.1, nas classes ValorIndicador, VolumeDeDados e ValorVolumeDeDados.

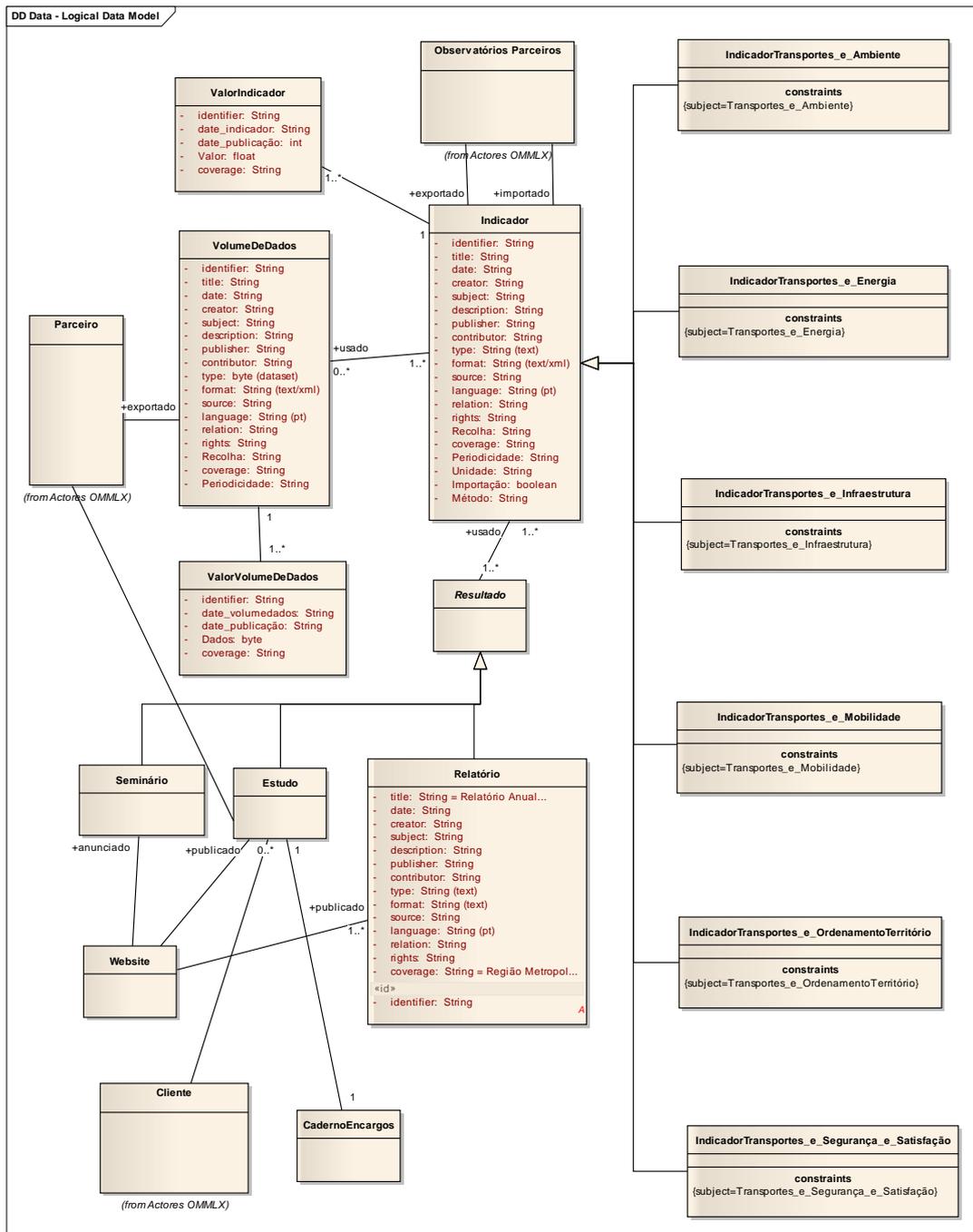


Figura 4.21 - Vista parcial do modelo lógico de dados.

Na Figura 4.21 pode ser observada uma vista parcial do modelo lógico de dados. Nesta figura observamos as classes Indicador, ValorIndicador, VolumeDeDados, ValorVolumeDeDados e as classes associadas aos Produtos e Serviços do Observatório e a ontologia de indicadores. Algumas classes não têm atributos para simplificar a representação gráfica, tendo em conta que não se pretende fazer desenvolvimento aplicacional. Exemplos de atributos podem ser encontrados na descrição das entidades informacionais associadas.

## 4.6 Fase 4 (células I, K, L, N, S)

### 4.6.1 Célula I – Sistema logístico da organização

O sistema logístico da organização é retratado através de um *Business Logistics Diagram*, o qual é baseado num diagrama de instalação. No caso concreto do Observatório trata-se de um diagrama relativamente simples, na medida em que se pretende uma organização pequena e concentrada apenas num local. A comunicação será preferencialmente electrónica e *offline* através da Internet para o público em geral (*website*) e para os clientes (*email*) e via Extranet para os parceiros, através de mecanismos automatizados de troca de informação (importação/exportação de indicadores e importação de volumes de dados). Esta realidade está modelizada na Figura 4.22.

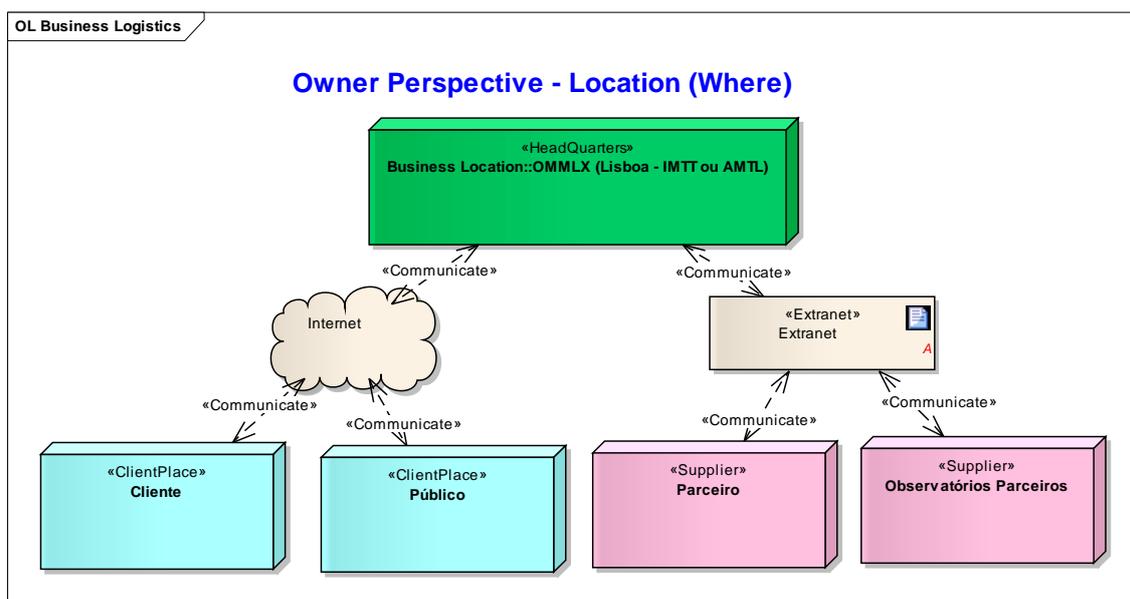


Figura 4.22 - Logística do Observatório.

### 4.6.2 Célula K – Cronograma principal

O cronograma principal mostra os eventos mais relevantes da organização. Para tal é sugerido o uso de um *Event Schedule Diagram*, o qual se baseia num diagrama de actividades.

Os eventos mais relevantes associados ao Observatório são apresentados na Figura 4.23. Estes eventos já tinham sido identificados na perspectiva anterior do *Framework* de

Zachman, mas neste caso concreto são apresentados segundo a sequência em que se desenrolam.

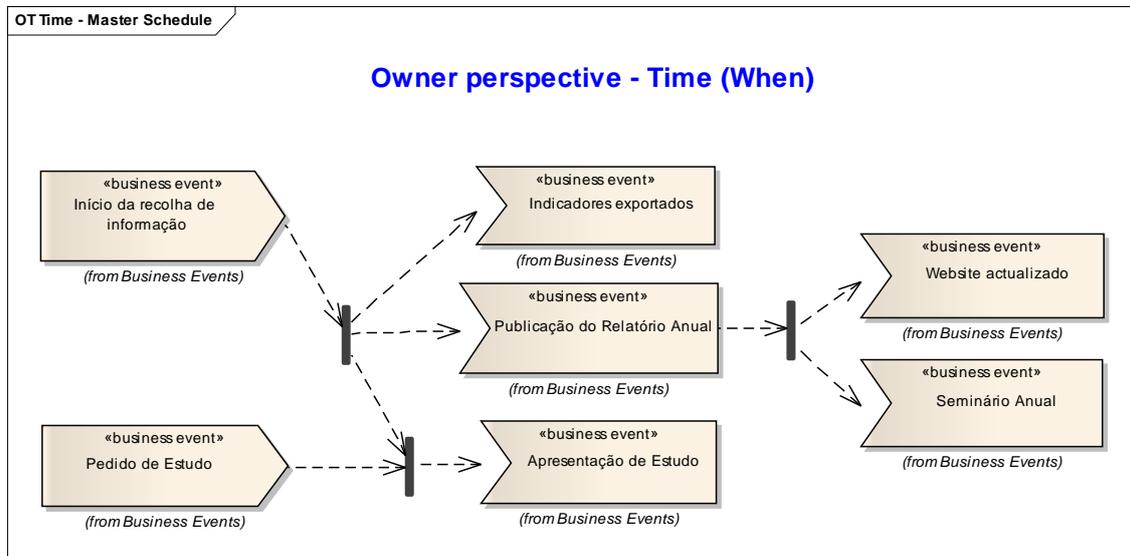


Figura 4.23 - Cronograma principal.

#### 4.6.3 Célula L – Plano de negócio

A Sparx propõe que o plano de negócio seja apresentado usando um *Strategy Map Diagram*. Este diagrama é uma extensão de um diagrama de classes, permitindo visualizar a relação entre os objectivos estratégicos e as estratégias para os implementar. A relação entre estes dois tipos de elementos é feita através de uma relação de dependência estereotipada como um *StrategyLink*.

Propõe-se organizarmos as estratégias segundo a abordagem para a elaboração de BSC (Balanced Scored Cards) (Kaplan & Norton, 1992). Esta abordagem permite definir posteriormente um painel de controlo de gestão, baseado nos BSCs, que irá medir a implementação das estratégias definidas. O próximo passo seria definir objectivos através de *tagged values* associados às diversas estratégias. Seguindo a abordagem de BSCs teremos as estratégias agrupadas segundo 4 *swimlanes* horizontais, representando outras tantas perspectivas:

- Perspectiva financeira;
- Perspectiva cliente;
- Perspectiva processos internos;
- Perspectiva aprendizagem e crescimento.

Na Figura 4.24 podemos observar o diagrama correspondente a esta célula.

## Definição da Arquitectura Empresarial

### O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

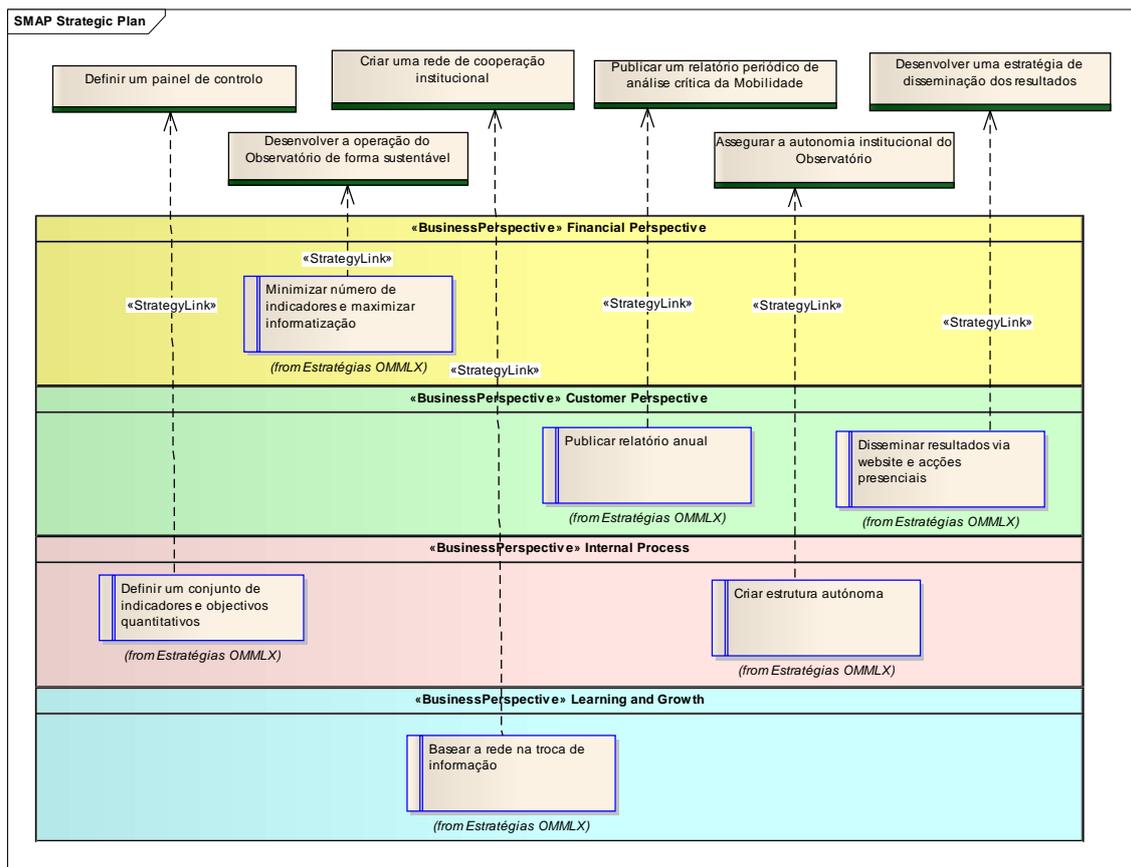


Figura 4.24 - Mapa de estratégia.

#### 4.6.4 Célula N – Arquitectura de aplicações

A proposta da Sparx para esta célula é um diagrama de actividades. Parece-nos pobre para descrever uma parte tão importante. A proposta de Pereira & Sousa (2004) parece-nos mais adequada: matrizes de sistemas versus entidades informacionais, sistemas versus processos de negócio e uma descrição dos sistemas num dicionário. Estes serão descritos por extenso na célula T e nas notas de cada componente desta célula. Nesta célula iremos usar um diagrama de componentes, mostrando os vários sistemas em jogo, anteriormente identificados através da matriz de CRUD entre processos de negócios e entidades informacionais. Iremos também usar as matrizes referidas.

Na Figura 4.25 podemos observar os vários sistemas, as entidades informacionais que estes sistemas criam e gerem, assim como os fluxos de informação ao nível das entidades informacionais que os vários sistemas usam. Note-se que estes fluxos foram identificados previamente na matriz de CRUD anteriormente referida através das operações de leitura (*read*).



Source: Processos Primários Type: Activity  
 Target: Sistemas e Fluxos Informacionais Type: Packag

	Sistema de Apoio à Realização de Seminários	Sistema de Gestão de Conteúdos de Website	Sistema de Produção de Indicadores	Sistema de Produção de Relatórios	Sistema de Suporte à Realização de Estudos
Actualizar website		↑			
Disseminar resultados	↑	↑			
Exportar indicadores			↑		
Importar dados			↑		
Importar indicadores			↑		
Produzir indicadores			↑		
Produzir relatório				↑	
Realizar estudo					↑
Realizar seminário	↑				

Figura 4.26 - Matriz Sistemas versus Processos de Negócio.

Relationship Matrix: MatrizSistemasEIClasses

Source: Sistemas e Fluxos Informacionais Type: Package Link Type: Dependency Profile: v  
 Target: Logical Data Model Type: Class Direction: Source -> Target

	CadernoEncargos	Estudo	Indicador	IndicadorTransportes_e_Ambiente	IndicadorTransportes_e_Energia	IndicadorTransportes_e_Infraestrutura	IndicadorTransportes_e_Mobilidade	IndicadorTransportes_e_OrdenamentoTerritório	IndicadorTransportes_e_Segurança_e_Satisfação	IndicadorTransportes_e_Serviços	IndicadorTransportes_e_Sócio-economia	IndicadorTransportes_e_Tecnologias	Relatório	Resultado	Seminário	ValorIndicador	ValorVolumeDeDados	VolumeDeDados	Website
Sistema de Apoio à Realização de Seminários		↑											↑		↑				
Sistema de Gestão de Conteúdos de Website		↑											↑		↑				↑
Sistema de Produção de Indicadores			↑	↑	↑	↑	↑	↑	↑	↑	↑	↑				↑	↑	↑	
Sistema de Produção de Relatórios			↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑						
Sistema de Suporte à Realização de Estudos	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑	

Figura 4.27 - Matriz de relacionamento entre sistemas e entidades informacionais (classes).

Na Figura 4.27 e Figura 4.28 podemos ver a matriz de relacionamento entre os sistemas e as entidades informacionais. Dado as entidades informacionais com a informação referente aos actores estar num pacote diferente, tornou-se necessário partir a matriz em duas partes. Note-se que só foram trabalhadas as super-classes. As sub-classes só são usadas quando

são relevantes. A relação usada foi *dependency*, ou seja, os sistemas dependem das entidades informacionais para atingir os seus objectivos.

	AMTL	Câmara Municipal	Cliente	Concessionário de Infra-Estruturas Ferroviárias	Concessionário de Infra-Estruturas Rodoviárias	Observatórios de Mobilidade	Observatórios Parceiros	Operador de Transportes Públicos	Operador de Transportes Públicos Ferroviários	Operador de Transportes Públicos Fluviais	Operador de Transportes Públicos Rodoviários	Operadores Transportes Públicos	Outros Observatórios	Parceiro	Público
Sistema de Apoio à Realização de Seminários			↑				↑								↑
Sistema de Gestão de Conteúdos de Website															
Sistema de Produção de Indicadores							↑								↑
Sistema de Produção de Relatórios															↑
Sistema de Suporte à Realização de Estudos			↑												↑

Figura 4.28 - Matriz de relacionamento entre sistemas e entidades informacionais (actores).

#### 4.6.5 Célula S – Modelo físico de dados

O modelo físico de dados é representado usando um *Logical Diagram*. Na prática trata-se de um diagrama de classes enriquecido. Este diagrama foi gerado automaticamente, tirando partido das capacidades de geração de código do Enterprise Architect. Esta operação foi efectuada usando uma transformação sobre o pacote correspondente às classes do modelo de dados lógico (ou célula M), obrigando a que todas as classes deste modelo estejam neste pacote. A transformação seleccionada foi para *Data Definition Language (DDL)* e o pacote de destino indicado corresponde à presente célula da *Framework* de Zachman.

Neste diagrama as classes correspondem a tabelas da base de dados com a indicação dos atributos que irão servir como chaves primárias e como chaves estrangeiras. Este modelo inclui algumas tabelas novas que surgem da conversão entre um modelo de dados orientado a objectos e um modelo de dados relacional. Surgem também novos atributos de identificação com o sufixo ID e tipo inteiro para servirem como chaves primárias. As regras de mapeamento de modelos lógicos UML para esquemas relacionais podem ser consultadas em Silva & Videira (2005).

Na Figura 4.29 pode ser observada uma vista parcial deste modelo, resultante da transformação automática.

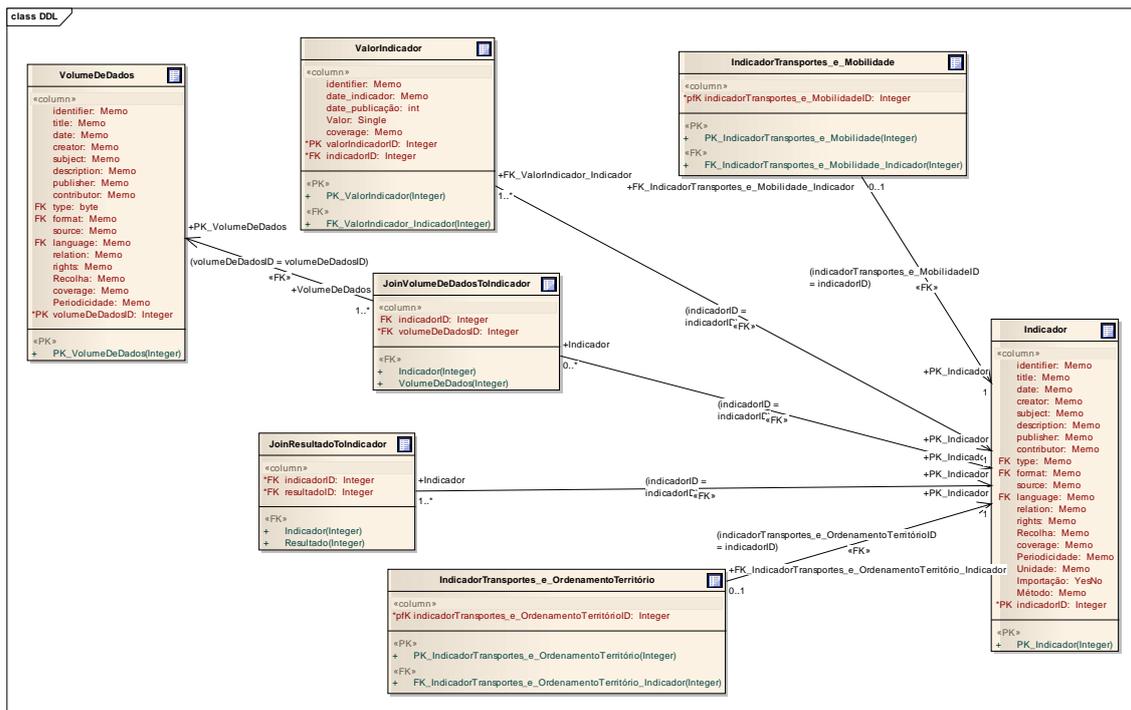


Figura 4.29 – Vista parcial do modelo físico de dados.

O próximo passo e que corresponderia à perspectiva seguinte, *Subcontractor*, seria a criação da base de dados. Esta perspectiva contempla as representações detalhadas segundo as diversas dimensões. Desta forma além da base de dados teríamos, entre outros, o código das aplicações, a arquitectura da rede informática e a arquitectura de segurança. O objectivo deste trabalho não contempla esta perspectiva, mas dado que o Enterprise Architect inclui funcionalidades poderosas de geração de código baseado em modelos (MDA), torna-se muito simples gerar o *script* em SQL de criação da base de dados. Para tal basta seleccionar o pacote com o modelo físico e na opção Project/Database Engineering escolher Generate Package DDL. Em poucos segundos este *script* é criado. A título de exemplo pode ser consultado no Anexo B o *script* gerado automaticamente para a tabela Indicador.

## 4.7 Fase 5 (células J, O, T)

### 4.7.1 Célula J – Modelo de Workflow

Para capturar a informação referente a esta célula a Sparx sugere um *Work Flow Model*, baseado num diagrama BPMN. Parece-nos que esta representação tem mais a ver com a dimensão *How* do que com a dimensão *Who*, pelo que optamos pela representação dos artefactos sugerida por Pereira & Sousa (2004), nomeadamente o organigrama da organização e uma matriz processos versus organização.

Na Figura 4.30 pode ser observada uma vista geral desta célula. O organigrama encontra-se num pacote específico e pode ser consultado na Figura 4.31. Clicando numa unidade organizacional (por exemplo Secretariado) ou num perfil funcional (por exemplo Secretária) é possível visualizar nas notas, respectivamente, os objectivos funcionais e a descrição funcional.

A matriz processos de negócio versus organização encontra-se representada por um *hyperlink*, o qual permite aceder a uma matriz de relacionamento, uma funcionalidade gráfica do Enterprise Architect que permite ver as relações entre dois pacotes. Neste caso entre o pacote com os processos de negócio primários e o pacote com o organigrama. O tipo de relação seleccionada é *control*, um estereótipo da relação *dependency*, usado nas extensões de Eriksson-Penker para modelação de negócio usando UML. Neste caso assume-se que os actores relacionados têm funções determinantes no processo relacionado, ou seja, têm funções de controlo. Fazendo duplo clique no *link* é possível visualizar a Figura 4.32.

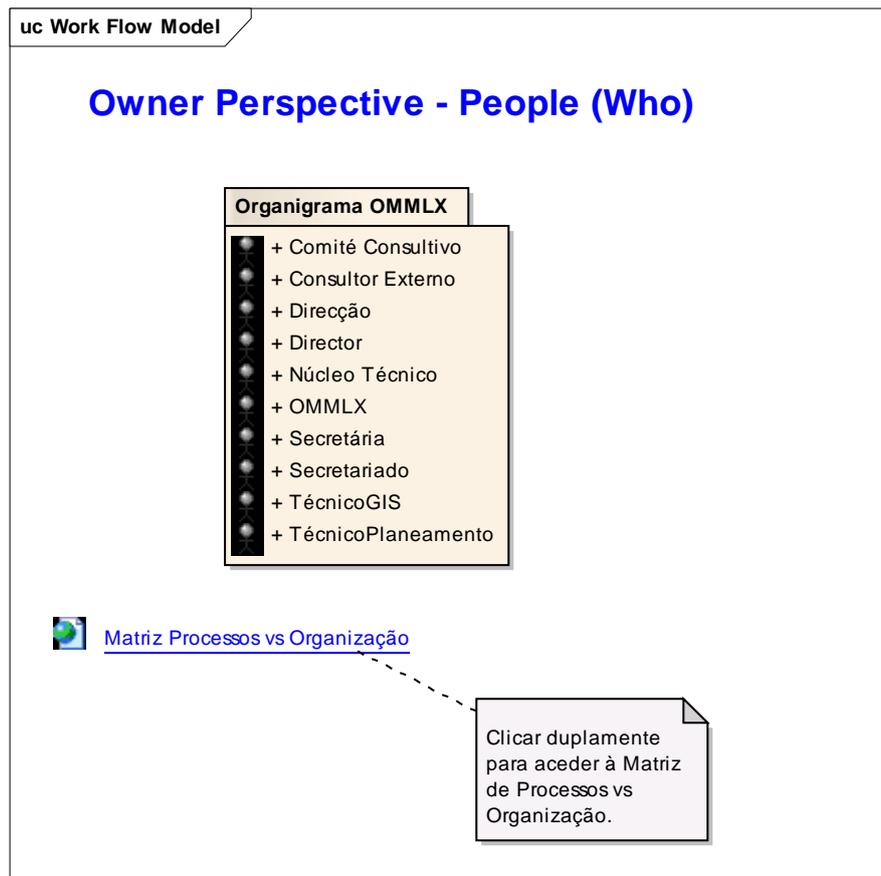


Figura 4.30 - Vista geral da célula J.

#### 4.7.2 Célula 0 – Arquitectura distribuída do sistema

Para representar a arquitectura distribuída do sistema é proposto um *Data Distribution Architecture Diagram*. Trata-se de um diagrama de instalação UML que permite uma representação independente da plataforma em que o sistema será instalado.

Este diagrama pode ser visualizado na Figura 4.33. Chama-se a atenção em particular para o facto da interoperabilidade em termos de indicadores, além da utilização da ontologia já referida, ser assegurada através de um directório onde são publicados os *web services* com os indicadores disponíveis.

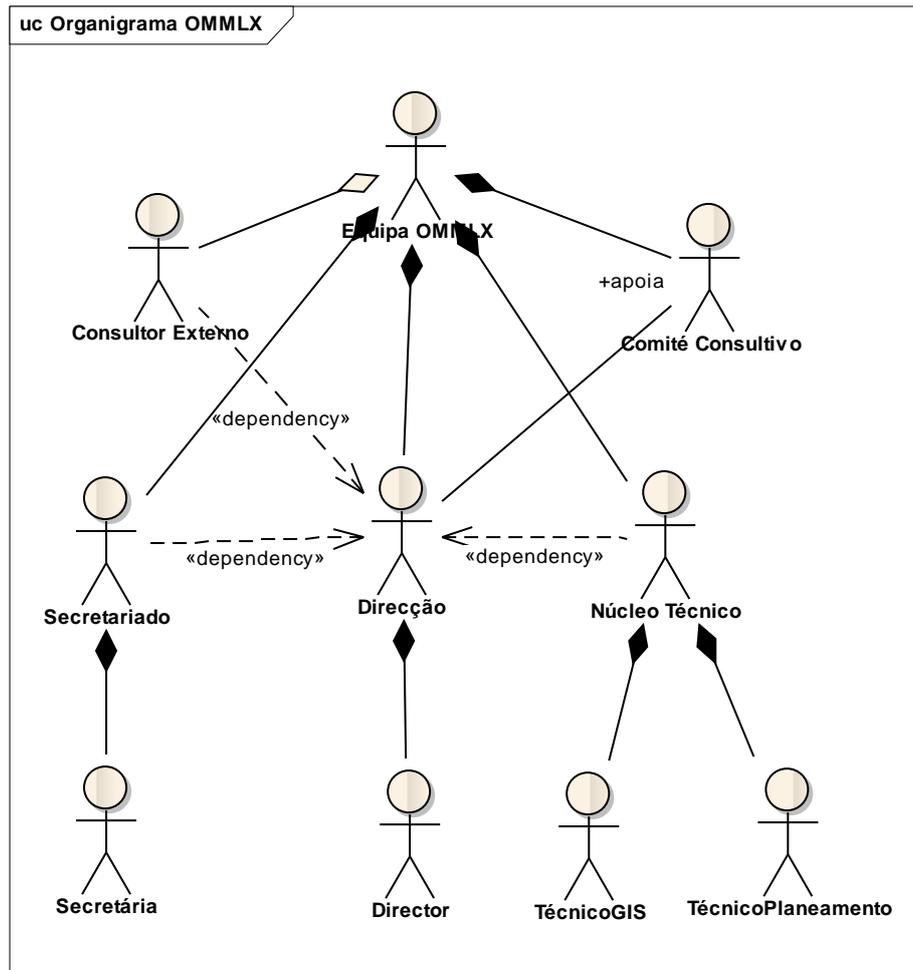


Figura 4.31 - Organigrama do Observatório.

ae\_ommlx - EA Academic

Relationship Matrix: MatrizProcessosOrganizacao

Source: Processos Primários Type: Activity Link Type: Control

Target: Organigrama OMMLX Type: Actor Direction: Target -> Source

	Comité Consultivo	Consultor Externo	Direcção	Director	Núcleo Técnico	OMMLX	Secretária	Secretariado	TécnicoGIS	TécnicoPlaneamento
Disseminar resultados			↑							↑
Exportar indicadores										↑
Importar dados										↑
Importar indicadores										↑
Produzir indicadores										↑
Produzir relatório										↑
Realizar estudo		↑	↑		↑					↑
Realizar seminário		↑								↑

Figura 4.32 - Matriz processos de negócio versus organização.

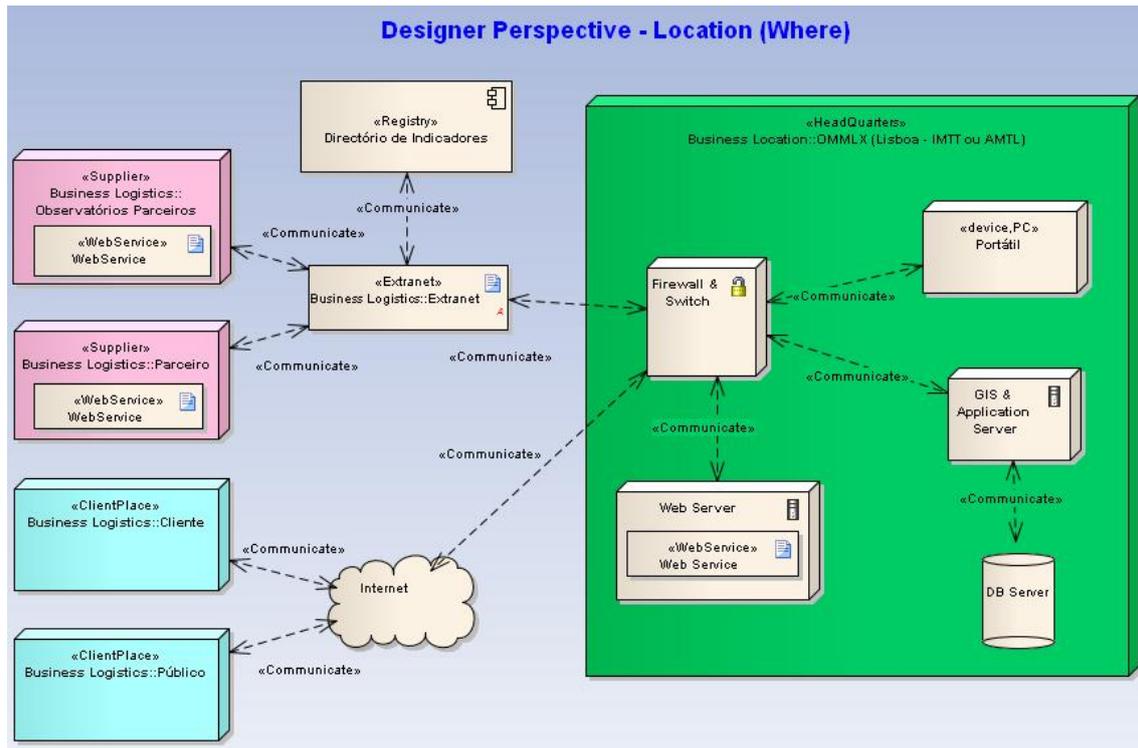


Figura 4.33 - Arquitectura distribuída do sistema.

#### 4.7.3 Célula T – Projecto de sistema

Esta célula captura a arquitectura dos programas e para tal a Sparx propõe um diagrama de componentes com as classes, interfaces e componentes lógicas do(s) sistema(s). No entanto, a estratégia neste caso concreto, passa por usar pacotes de aplicações existentes em vez de desenvolver as aplicações de origem. Claro que haverá trabalho de integração, particularmente ao nível da interoperabilidade e da configuração. Desta forma será usado um diagrama de componentes, representando os sistemas e os pacotes aplicacionais associados.

Para os diversos sistemas serão usados os pacotes aplicacionais seguidamente referidos. Note-se que alguns poderão ser usados em vários.

Sistema de Produção de Indicadores

- Módulo de processamento estatístico - SPSS (Statistical Package for the Social Sciences) – Pacote com várias componentes tipicamente usado em análise estatística. Este pacote pode interligar a bases de dados relacionais externas através de ODBC e SQL. Será usado para produzir indicadores.
- Módulo de processamento espacial – GIS (Geographical Information System) – Pacote informático para visualização, processamento e armazenamento de informação geográfica. Será usado no cálculo e visualização de indicadores baseados em volumes de dados com informação georeferenciada. Exemplo: ArcGIS da ESRI.
- Módulo de interoperabilidade – Combinação de servidor web e de servidor aplicacional suportando JAVA para implementação de web services (ex: APACHE e GERONIMO).

- Módulo de burótica – Pacote de software de escritório electrónico com alguma capacidade de processamento ao nível de folhas de cálculo. Por exemplo, MS-Office e MS-Excel.

#### Sistema de Produção de Relatórios

- Módulo de burótica – Pacote de software de escritório electrónico, focado neste caso para o processamento de texto, figuras e gráficos, com capacidade de produção de documentos em formato PDF. Exemplo MS Office e MS Word.

#### Sistema de Suporte à Realização de Estudos

- Módulo de simulação – Aplicação para simulação de tráfego. Exemplo: EMME (software de previsão de procura de transportes).
- Módulo de processamento estatístico - SPSS (Statistical Package for the Social Sciences) – Pacote com várias componentes tipicamente usado em análise estatística. Este pacote pode interligar a bases de dados relacionais externas através de ODBC e SQL.
- Módulo de processamento espacial – GIS – Pacote informático para visualização, processamento e armazenamento de informação geográfica. Exemplo: ArcGIS da ESRI.
- Módulo de burótica – Pacote de software de escritório electrónico, focado neste caso para o processamento de texto, figuras e gráficos, com capacidade de produção de documentos em formato PDF e com funcionalidade de folha de cálculo. Exemplo MS Office e MS Word/MS Excel.

#### Sistema de Apoio à Realização de Seminários

- Módulo de burótica – Pacote de software de escritório electrónico, focado neste caso para o processamento de texto, figuras e gráficos, com capacidade de produção de documentos em formato PDF e com funcionalidade de folha de cálculo. Exemplo MS Office e MS Word/MS Excel.

#### Sistema de Gestão de Conteúdos de Website

- Módulo de Gestor de Conteúdos – Pacote de software para a criação de website e para a actualização dos conteúdos respectivos, tipicamente intitulado CMS (Content Management Server). Exemplo: PLONE.

Além dos vários pacotes aplicativos referidos, será também usada um sistema de gestão de base de dados. Um exemplo será o MS SQL Server.

Na Figura 4.34 pode ser observado o diagrama de componentes com os pacotes aplicativos que implementam os sistemas.

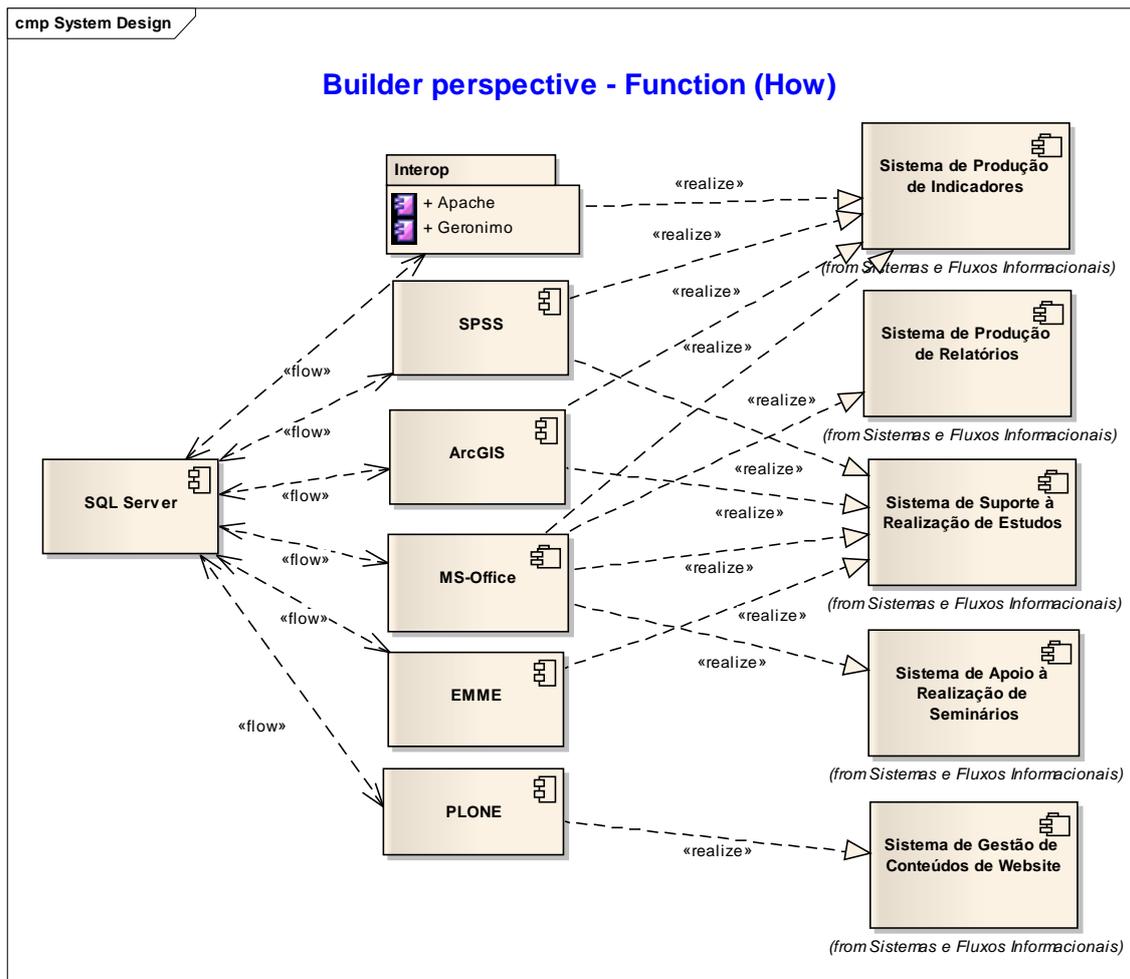


Figura 4.34 - Diagrama de componentes com os pacotes aplicativos que implementam os sistemas.

## 4.8 Fase 6 (célula U)

### 4.8.1 Célula U - Arquitectura da tecnologia

A proposta para esta célula é um diagrama de instalação. De forma a constituir uma referência para uma memória descritiva ou caderno de encargos, neste diagrama são instanciados o número de equipamentos e exemplos dos mesmos. O resultado pode ser observado na Figura 4.35.

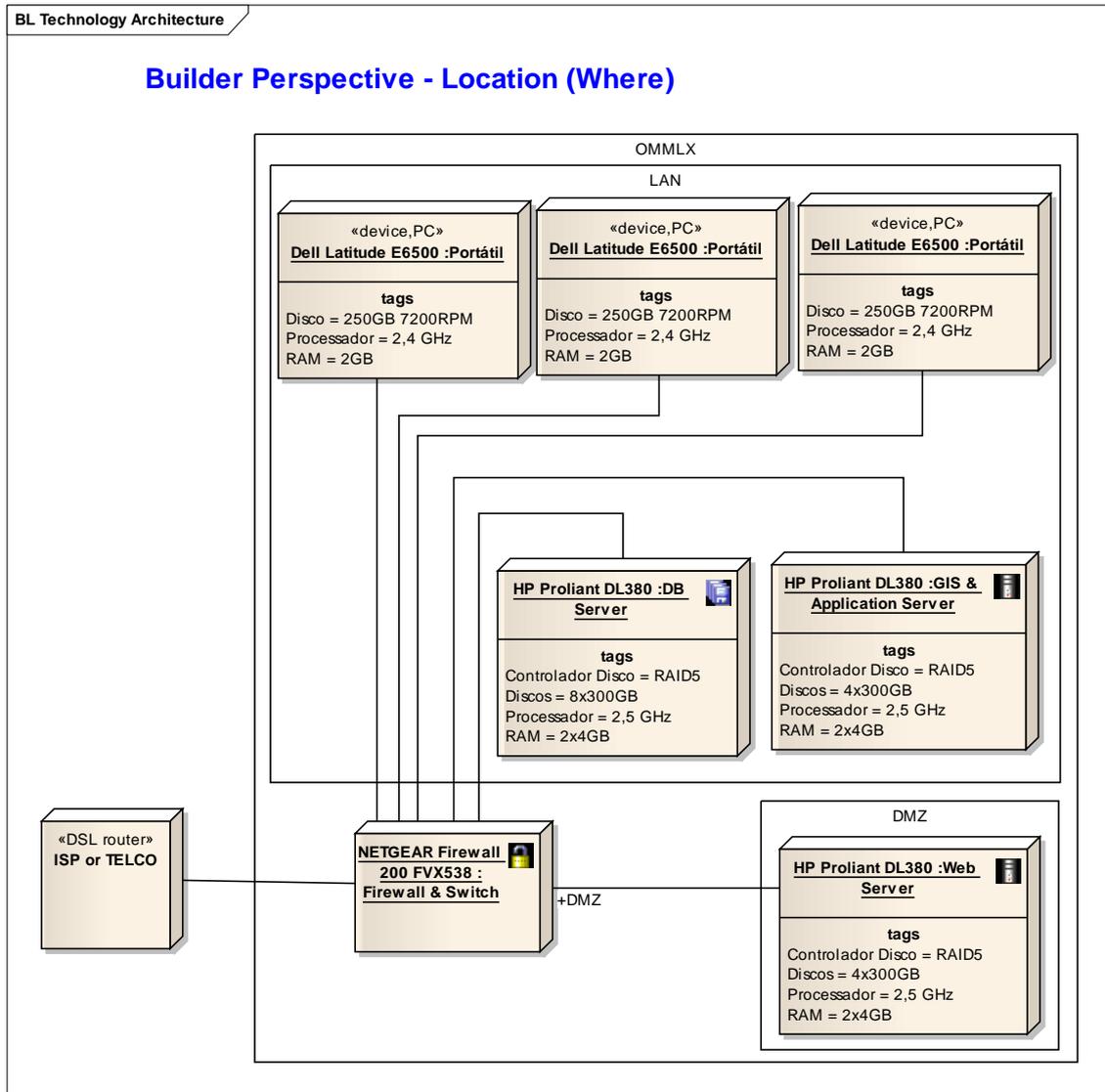


Figura 4.35 – Diagrama de instalação com número de equipamentos e tipo.

## 4.9 Sub-Arquitecturas da Arquitectura Empresarial

Para criarmos as diversas Sub-Arquitecturas da Arquitectura Empresarial, basta reorganizar os artefactos. Para tal basta copiá-los para os pacotes adequados, usando a opção *shallow*, a qual permite a sincronização entre os elementos dos diagramas. A sincronização fica apenas garantida a este nível, pois se existirem alterações nos diagramas estas não serão replicadas. Tal poderá colocar problemas de manutenção.

Foram seleccionados os artefactos considerados relevantes, tendo em conta os artefactos disponíveis e os objectivos de cada Arquitectura. O resultado pode ser observado na Figura 4.36.

## Definição da Arquitectura Empresarial

### O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

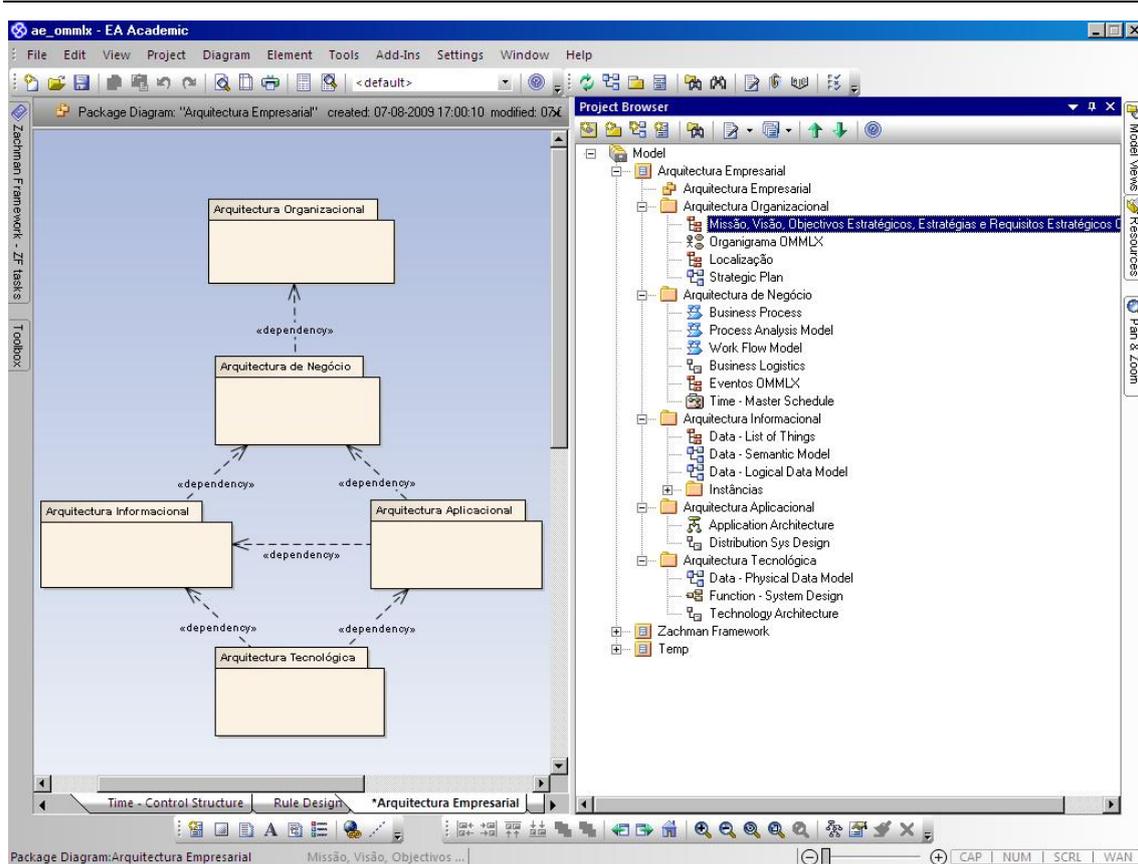


Figura 4.36 - Sub-Arquitecturas da Arquitectura Empresarial e respectivos artefactos.

O passo seguinte é gerar automaticamente um relatório RTF a partir do repositório, descrevendo as diversas Arquitecturas, de forma a fazer parte da memória descritiva do Observatório.

De forma a emular uma situação real de um repositório partilhado, com o objectivo de expôr os conteúdos e obrigar à sua sincronização com o mundo real, podemos elaborar um relatório em HTML e publicar num servidor *web*. Nas figuras seguintes é possível apreciar o aspecto deste tipo de relatório.

O relatório HTML foi publicado na página do aluno na FEUP.

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

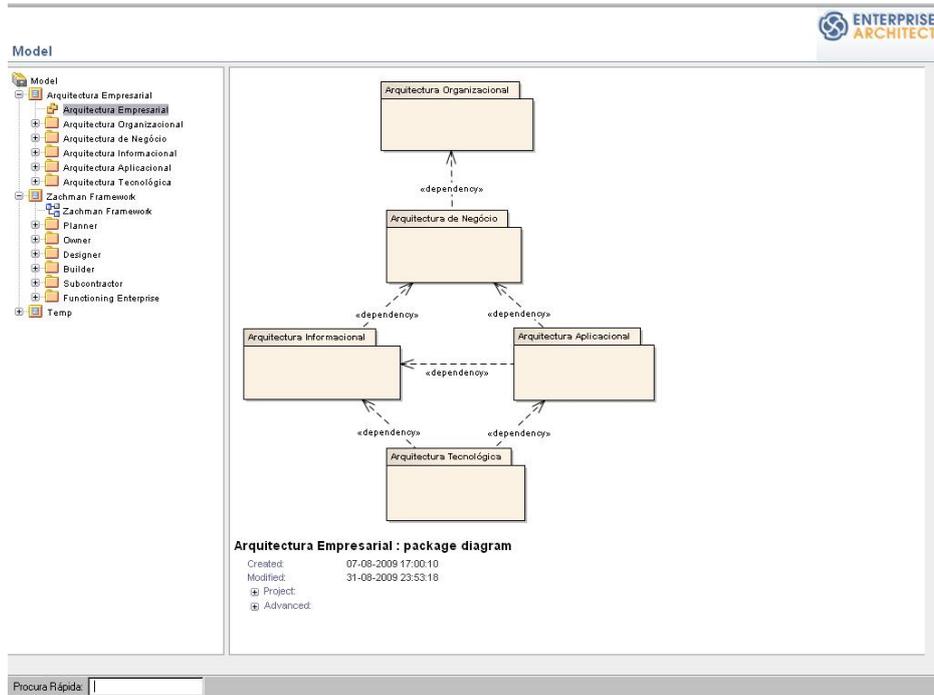


Figura 4.37 - Relatório HTML – Vista Arquitectura Empresarial.

The Zachman Framework	DATA What	FUNCION How	NETWORK Where	PEOPLE Who	TIME When	MOTIVATION Why
<b>SCOPE</b> (Contextual) Planner	Things Important to the Business	Processes the Business Performs	Locations in which the Business Operates	Organizations Important to the Business	Events/Cycles Significant to the Business	Business Goals/Strategies
<b>BUSINESS MODEL</b> (Conceptual) Owner	Conceptual Data Model	Business Process Model	Business Logistics	Work Flow Model	Master Schedule	Business Plan
<b>SYSTEM MODEL</b> (Logical) Designer	Logical Data Model	Application Architecture	Distributed System Architecture	Human Interface Architecture	Processing Structure	Business Rule Model
<b>TECHNOLOGY MODEL</b> (Physical) Builder	Physical Data Model	System Design	Technology Architecture	Presentation Architecture	Control Structure	Rule Design
<b>DETAILED REPRESENTATIONS</b> Sub-Contractor	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Specification
<b>FUNCTIONING ENTERPRISE</b>	Data	Function	Network	Organization Units	Schedule	Strategy

Figura 4.38 - Relatório HTML – Vista *Framework* de Zachman.

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

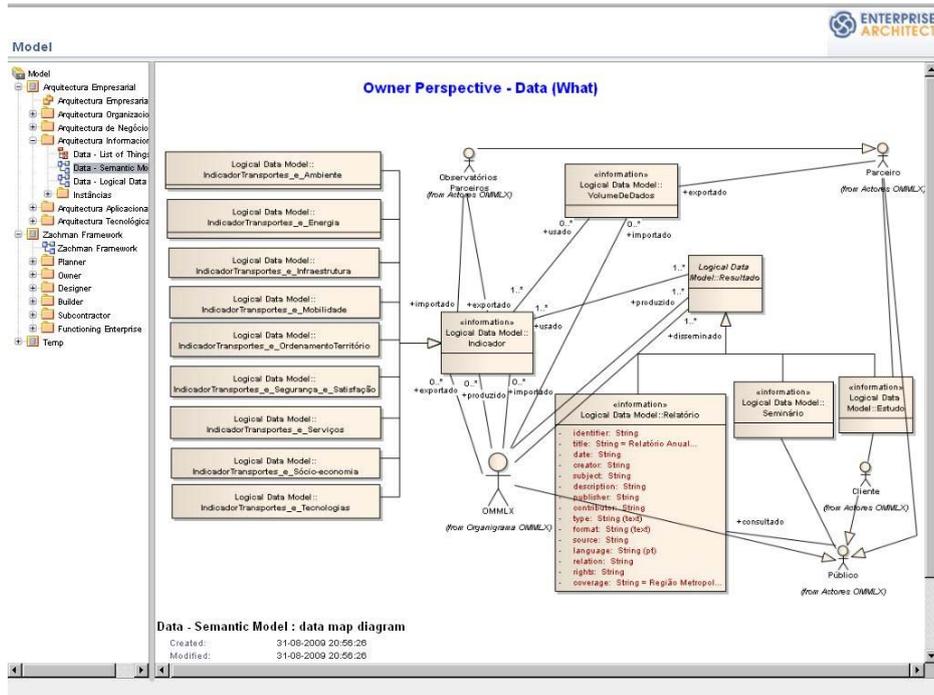


Figura 4.39 - Relatório HTML – Pormenor de navegação na Arquitectura Informacional.

## 5 Conclusões e trabalho futuro

Esta dissertação consistiu na realização de uma Arquitectura Empresarial, aplicada ao caso concreto da criação do Observatório de Mobilidade na Região Metropolitana de Lisboa.

Inserida na realização da Arquitectura Empresarial foi também criada uma ontologia de indicadores de mobilidade, com vista a permitir a operação do Observatório em rede, importando e exportando indicadores para outros Observatórios de natureza complementar, quer da mobilidade, quer de outras áreas.

A construção da Arquitectura Empresarial foi suportada na ferramenta Enterprise Architect.

A utilização deste tipo de ferramentas apresenta várias vantagens:

- Validam os artefactos usados, tendo como base a linguagem associada;
- Produzem um modelo coerente, com base numa linguagem que suporta os vários artefactos, neste caso UML;
- Permitem a edição cooperativa, contribuindo para reduzir tempos ao envolver mais recursos simultaneamente;
- Suportam a geração de código a partir de modelos (MDA), aumentando a utilidade da Arquitectura Empresarial ao integrá-la no processo de produção de software e garantindo o alinhamento entre os modelos e o código associado, o que foi verificado na Arquitectura Informacional, com as transformações de modelo lógico de dados para modelo físico de dados;
- Permitem a rastreabilidade dos objectivos do negócio até ao código, permitindo garantir a utilidade do código produzido;
- Constituem-se num referencial comum para os recursos humanos associados ao negócio e os associados à tecnologia;
- Permitem a publicação em HTML, constituindo-se num repositório “público”, expondo a Arquitectura Empresarial à crítica e à sua constante actualização, à medida que o negócio se adapta à envolvente externa;
- São uma base de conhecimento sobre a organização.

A ontologia de indicadores de mobilidade foi criada usando a ferramenta Protégé.

Foi sentida a necessidade de criação desta ontologia, dado a operação do Observatório em rede ser uma possibilidade e perspectivar-se a troca de indicadores com outros Observatórios. A ontologia permite estruturar o conhecimento do domínio da mobilidade, neste caso, uma estrutura de indicadores que permite avaliar o sucesso das políticas de mobilidade. A vantagem de usar uma ferramenta deste tipo reside no facto de suportar linguagens de ontologias, permitindo, por exemplo, através de repositórios de ontologias, importar/integrar ontologias já existentes e partilhar esta. Infelizmente a interoperabilidade procurada não foi atingida com as ferramentas seleccionadas, o que obrigou a uma importação semi-automática, com muito processamento manual, devido a perder-se muita informação no processo automático.

O método usado para construir a Arquitectura Empresarial consistiu numa abordagem híbrida integrando propostas de várias fontes, nomeadamente a EAP (Spewak & Hill, 1992), a sequência apresentada por Pereira & Sousa (2004) e a proposta da Sparx Systems (Sparks, 2008). Na prática o preenchimento acabou por ser ligeiramente diferente da sequência proposta. Dado que se tratava da modelação de uma nova organização, prévia à sua criação, sentiu-se claramente necessidade de dar prioridade às células associadas ao negócio, particularmente na primeira linha do *Framework* de Zachman. Foi também necessário visitar as células após uma primeira passagem, para melhorar o modelo. Estamos a falar essencialmente de enriquecimento dos elementos, com novos atributos, e também da arrumação destes em pacotes diferentes de forma a facilitar a criação de matrizes de relacionamento ou transformações MDA.

Durante este trabalho ficou bem patente o potencial da utilização de um framework para a criação de uma Arquitectura Empresarial e conseqüentemente para o planeamento estratégico dos sistemas de informação e das tecnologias de suporte, em alinhamento com os objectivos do negócio. As vantagens da sua operacionalização através de uma ferramenta foram também evidentes, mas o ponto que gostaríamos de sublinhar consiste na possibilidade de integrar MDA no processo, permitindo no limite gerar código a partir dos vários artefactos. Tal permite navegar nas colunas preenchendo as células seguintes com artefactos gerados a partir dos artefactos das células anteriores, fazendo as transformações adequadas. Os ganhos de produtividade associados serão ainda maiores.

Em termos de linhas de trabalho futuro foram identificadas algumas oportunidades e interesses, que se descrevem em seguida.

#### **Implementar a validação automática de heurísticas no Enterprise Architect**

Neste trabalho a validação foi feita de forma manual, mas parece-nos que a ferramenta poderá ser programada para validações automáticas. A verificar-se poderá ser um ganho interessante para a operacionalização da criação de uma Arquitectura Empresarial, particularmente envolvendo modelos muito mais complexos.

#### **Integração das ontologias na Arquitectura Empresarial**

Foi com alguma surpresa, particularmente devido à proximidade dos conceitos usados, que não foi possível fazer uma importação completa da ontologia de indicadores na Arquitectura Empresarial. Acreditamos que se trata de uma questão de pequenas divergências na interpretação de normas e que tal ficará resolvido em novas versões das ferramentas usadas. Será um objectivo a perseguir.

#### **Avaliar outros *frameworks***

Como foi referido no capítulo sobre o Estado da Arte existem inúmeros *frameworks*, mais recentes e adaptados a sectores de actividade. Será um exercício interessante avaliar a aplicabilidade de alguns deles. O Enterprise Architect permite programar novos *frameworks*, com os respectivos diagramas e elementos, permitindo automatizar o processo, mantendo as vantagens identificadas neste trabalho.

#### **Agilidade na produção de Arquitecturas Empresariais**

Mesmo com o apoio das ferramentas usadas neste trabalho, sente-se que em realidades mais complexas o processo de criação da Arquitectura Empresarial será demorado. A utilização de agilidade no processo poderia ser um contributo

interessante, particularmente ao nível da manutenção. A combinação da publicação em HTML com a edição usando wikis, poderia ser um contributo interessante nesta direcção, embora seja necessário acautelar a integridade do modelo.

### **Explorar MDA**

O potencial do MDA associado a uma Arquitectura Empresarial parece-nos enorme. Gostaríamos de explorar uma situação concreta quer de produção de células do *Framework* de Zachman a partir de outras, quer de produção de código aplicacional e de bases de dados, usando o Enterprise Architect como IDE ou ligado a um IDE externo.

### **A dimensão temporal**

À medida que o negócio evoluiu é necessário produzir novos artefactos. A gestão do histórico, falamos quer do passado, quer do futuro (TO BE) não está contemplada nesta ferramenta.

### **Testar a organização**

Ao capturarmos o funcionamento da empresa, particularmente ao nível da sua Arquitectura de Negócio, torna-se possível “testá-la”. Estamos a pensar em exercícios de benchmarking contra outras Arquitecturas Empresariais de outras organizações ou mesmo de confrontação com condicionantes externas, como leis e regulamentos. Ao integrarmos BSC é possível também fazer *benchmarking* de desempenho.

## 6 Referências

- Aberdeen (2009). *ONTOSEARCH2*. Retrieved 2009/06/28 from <http://www.ontosearch.org/>.
- Agate (2005). *Framework AGATE*. Retrieved 2009/09/04 from <http://www.achats.defense.gouv.fr/article33349>.
- Beckett, D. & Berners-Lee, T. (2008). *Formato Turtle*. Retrieved 2009/09/06 from <http://www.w3.org/TeamSubmission/turtle/>.
- Berdier, C. & Roussey, C. (2006). *Urban Ontologies: the townology prototype towards case studies*. Retrieved 2009/06/21 from <http://townology.net/Meetings/0611-Geneva/presentations/BerdierRoussey.pdf>.
- C4isr (1997). *Framework C4ISR*. Retrieved 2009/09/04 from <http://fas.org/irp/program/core/fw.pdf>.
- Carvalho, G. & Sousa, P. (2008, June). *Business and Information Systems MisAlignment Model (BISMAM): an Holistic Model Leveraged on Misalignment and Medical Sciences Approaches*, Paper presented at the 3rd International Workshop on Business/IT Alignment and Interoperability, Montpellier, France.
- Chan, Y. E. & Reich, B.H. (2007). IT alignment: an annotated bibliography. *Journal of Information Technology*, 22, pp. 316-396.
- Chan, Y. E. & Reich, B.H. (2007). IT alignment: what have we learned?. *Journal of Information Technology*, 22, pp. 297-315.
- Ciborra, C. (2002). Krisis. In Oxford University Press (Ed.), *The Labyrinths of Information*. New York, USA: Oxford University Press Inc. pp. 11-27.
- Dodaf (2007). *Framework DoDAF*. Retrieved 2009/09/04 from [http://www.defenselink.mil/cio-nii/docs/DoDAF\\_Volume\\_I.pdf](http://www.defenselink.mil/cio-nii/docs/DoDAF_Volume_I.pdf).
- E2af (2005). *Framework E2AF*. Retrieved 2009/09/04 from <http://www.enterprise-architecture.info/Images/Extended%20Enterprise/Extended%20Enterprise%20Architecture.htm#e2af>.
- Ebi (2009). *OLS - Ontology Lookup Service*. Retrieved 2009/06/28 from <http://www.ebi.ac.uk/ontology-lookup/init.do>.
- Eriksson, H. & Penker, M. (2000). *Business Modeling with UML: Business Patterns at Work*. New York, USA: Wiley.

FLogic (2009). *Definição de F-Logic*. Retrieved 2009/09/08 from <http://en.wikipedia.org/wiki/F-logic>.

Feaf (1999). *Framework FEAF*. Retrieved 2009/09/04 from <http://www.enterprise-architecture.info/Images/Documents/Federal%20EA%20Framework.pdf>.

Frankel, D. S., Harmon, P., Mukerji, J., Odell, J., Owen, M., Rivitt, P., Rosen, M. & Soley, R.M. (2003). *The Zachman Framework and the OMG's Model Driven Architecture*. Retrieved 2009/09/05 from <http://www.bptrends.com/publicationfiles/09-03%20WP%20Mapping%20MDA%20to%20Zachman%20Framework.pdf>.

Gaspar, A. (2008,October). *Observatório de Mobilidade numa Região Metropolitana*, Paper presented at the I Congresso Internacional de Ingeniería Civil y Territorio "El Litoral", Vigo, Galicia, Spain.

Gaspar, A., Babo, A., Xavier, A., Barros, R. & Amaro, M. (2008). *Observatório de Mobilidade da Região Metropolitana de Lisboa - Proposta de Observatório e Memória Descritiva*. (Estudo do Observatório de Mobilidade da Região Metropolitana de Lisboa). Lisboa: CCDR-LVT

Gruber, T. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2), pp. 199-220.

Henderson, J. C. & Venkatraman, N. (1999). Strategic alignment: Leveraging information technology for transforming organizations (REPRINT). *IBM Systems Journal*, 38 (2&3), pp. 472-484.

Iaf (1993). *Framework IAF*. Retrieved 2009/09/04 from [http://www.capgemini.com/resources/thought\\_leadership/architecture\\_and\\_the\\_integrated\\_architecture\\_framework/?d=1](http://www.capgemini.com/resources/thought_leadership/architecture_and_the_integrated_architecture_framework/?d=1).

Iana (2009). *Lista de tipos de media (MIME) para a Internet*. Retrieved 2009/08/16 from <http://www.iana.org/assignments/media-types/>.

Ibm (2009). *IBM Rational System Architect*. Retrieved 2009/09/06 from <http://www.telelogic.com/products/systemarchitect/index.cfm>.

Ids Scheer (2009). *ARIS*. Retrieved 2009/09/06 from [http://www.ids-scheer.com/en/ARIS/ARIS\\_Platform/ARIS\\_IT\\_Architect/3741.html](http://www.ids-scheer.com/en/ARIS/ARIS_Platform/ARIS_IT_Architect/3741.html).

Ieee (2000). *Norma IEEE 1471:2000*. Retrieved 2009/09/05 from [http://standards.ieee.org/reading/ieee/std\\_public/description/se/1471-2000\\_desc.html](http://standards.ieee.org/reading/ieee/std_public/description/se/1471-2000_desc.html).

Iif (2006). *Framework IIF*. Retrieved 2009/09/05 from [http://www.finance.gov.au/publications/agimo/docs/Information\\_Interoperability\\_Framework.pdf](http://www.finance.gov.au/publications/agimo/docs/Information_Interoperability_Framework.pdf).

Kaplan, R. S. & Norton, D.P. (1992). The Balanced Scorecard - Measures that Drive Performance. *Harvard Business Review*, , pp. 70-79.

Laudon, K. C. & Laudon, J.P. (2004). Definição de Cadeia de Valor de Porter. In Prentice Hall (Ed.), *Management Information Systems: Managing the Digital Firm*. New Jersey, USA: Prentice Hall. pp. 92-93.

- Laudon, K. C. & Laudon, J.P. (2004). Definição de Sistemas de Informação Estratégicos. In Prentice Hall (Ed.), *Management Information Systems: Managing the Digital Firm*. New Jersey, USA: Prentice Hall. p. 91.
- Luftman, J. (2005). Key Issues for IT Executives 2004. *MIS Quarterly Executive*, 4 (2), pp. 269-285.
- McIlree, R. (2006). *The Enterprise Architecture Definition Collection*. Retrieved 2008/02/15 from [http://enterprisearchitected.typepad.com/ea/2006/11/the\\_enterprise\\_.html](http://enterprisearchitected.typepad.com/ea/2006/11/the_enterprise_.html).
- Microsoft (2009). *Framework Microsoft*. Retrieved 2009/09/05 from <http://msdn.microsoft.com/en-us/architecture/ms978007.aspx>.
- Mitre (2004). *Guide to the (Evolving) Enterprise Architecture Body of Knowledge*. (EABOK). McLean, Virginia, USA: The MITRE Corporation
- Modaf (2008). *Framework MODAF*. Retrieved 2009/09/05 from <http://www.modaf.org.uk/>.
- Nato (2005). *Framework NATO C3*. Retrieved 2009/09/05 from [http://194.7.80.153/website/home\\_volumes.asp?menuid=15](http://194.7.80.153/website/home_volumes.asp?menuid=15).
- NeOn (2009). *Website NeOn Toolkit*. Retrieved 2009/09/08 from [http://neon-toolkit.org/wiki/Main\\_Page](http://neon-toolkit.org/wiki/Main_Page).
- Noy, N. F. & McGuinness, D.L. (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. (Technical Report KSL-01-05). Palo Alto, California, USA: Knowledge Systems Laboratory, Stanford University
- Odm (2009). *Definição de ODM*. Retrieved 2009/09/08 from <http://www.omg.org/docs/ad/05-08-01.pdf>.
- Omg (2009). *Site OMG*. Retrieved 2009/09/07 from [www.omg.org](http://www.omg.org).
- Ostaszadeh, S., Alice, F. S. & Ostaszadeh, A. (2007). A Method for Consistent Modeling of Zachman Framework Cells. In K. Elleithy (Ed.), *Advances and Innovations in Systems, Computing Sciences and Software Engineering*. The Netherlands: Springer. pp. 375-380.
- Owl Editor (2009). *Website OWL Editor*. Retrieved 2009/09/08 from <http://www.modelfutures.com/owl>.
- Palazzo, L. A. M. (2006). *Projeto e Construção de Ontologias*. Retrieved 2009/06/19 from <http://www.inf.pucrs.br/~ai190471/Mestrado/slides/PCO3.ppt>.
- Pereira, C. M. & Sousa, P. (2003,September). *Getting into the misalignment between Business and Information Systems*, Paper presented at the 10th European Conference On Information Technology Evaluation, Madrid, Spain.
- Pereira, C. M. & Sousa, P. (2004,March). *A Method to Define an Enterprise Architecture using the Zachman Framework*, Paper presented at the 2004 ACM Symposium on Applied Computing, Nicosia, Cyprus.

- Promis (2009). *EVA Netmodeler*. Retrieved 2009/09/06 from <http://www.promis.com/eva-netmodeler.html>.
- Protégé (2009). *Website Protégé*. Retrieved 2009/09/08 from <http://protege.stanford.edu/>.
- Ross, J. W. (2007). *Enterprise Architecture as Strategy*. Retrieved 2008/02/15 from [http://colab.cim3.net/file/work/caf/meetings/Jeanne\\_Ross\\_01\\_08\\_2007\\_EA.pdf](http://colab.cim3.net/file/work/caf/meetings/Jeanne_Ross_01_08_2007_EA.pdf).
- SemanticWeb (2009). *Relação entre Web Semântica e Ontologias*. Retrieved 2009/09/08 from <http://semanticweb.org/wiki/Ontology>.
- Silva, A. & Videira, C. (2005). Regras de Mapeamento de Modelos Lógicos UML em Esquemas Relacionais. In Centro Atlântico (Ed.), *UML Metodologias e Ferramentas CASE*. Lisboa: Centro Atlântico. p. 303-.
- Sousa, P. (2005). *Acetatos Aula 6 - Arquitectura Empresarial - Âmbito e Negócio*. (Disciplina de Sistemas de Informação Empresariais). Lisboa: POSI - Pós-Graduação em Sistemas de Informação, Edição 7, 2005-2006, IST.
- Sousa, P., Caetano, A., Vasconcelos, A., Pereira, C. & Tribolet, J. (2007). Enterprise Architecture Modeling with the Unified Modeling Language. In P. Rittgen (Ed.), *Enterprise Modeling and Computing with UML*. USA: Idea Group Publishing. pp. 69-96.
- Sousa, P., Pereira, C. M. & Marques, J.A. (2004). Enterprise architecture alignment heuristics. *Microsoft Architects Journal*, 4, pp. 34-39.
- Sparks, G. (2008). MDG Technology for Zachman Framework User Guide. Sparx Systems Pty Ltd. Creswick, Victoria, Australia
- Sparx (2009). *Enterprise Architect*. Retrieved 2009/09/06 from <http://www.sparxsystems.com/products/ea/index.html>.
- Spewak, S. H. & Hill, S.C. (1992). Enterprise architecture planning : Developing a blueprint for data, applications and technology. USA: Wiley.
- Swoogle (2009). *Swoogle*. Retrieved 2009/09/05 from <http://swoogle.umbc.edu/>.
- Teaf (2000). *Framework TEAF*. Retrieved 2009/09/05 from <http://www.eaframeworks.com/TEAF/teaf.doc>.
- The Dublin Core Metadata Initiative (2009). *Ontologia genérica de metadados Dublin Core v1.1*. Retrieved 2009/08/16 from <http://purl.org/dc/elements/1.1/>.
- Togaf (2009). *Framework TOGAF*. Retrieved 2009/09/05 from <http://www.opengroup.org/architecture/togaf9/downloads.htm>.
- TopBraid (2009). *Website TopBraid Composer*. Retrieved 2009/09/08 from [http://www.topquadrant.com/products/TB\\_Composer.html](http://www.topquadrant.com/products/TB_Composer.html).
- TopQuadrant (2003). *The Ontology Lifecycle*. Retrieved 2009/06/24 from <http://www.coolheads.com/egov/opensource/topicmap/s167/img28.html>.

Umbc (2009). *Swoogle - Semantic Web Search Engine*. Retrieved 2009/06/28 from <http://swoogle.umbc.edu/>.

W3C (2009). *Site W3C*. Retrieved 2009/09/08 from <http://www.w3.org/>.

Ward, J. & Peppard, J. (2002). Definição de Sistemas de Informação Estratégicos. In Wiley Series in Information Systems (Ed.), *Strategic Planning for Information Systems*. England: John Wiley & Sons. p. 23.

Wikipedia/Daml (2009). *Definição de DAML*. Retrieved 2009/06/26 from [http://en.wikipedia.org/wiki/DARPA\\_Agent\\_Markup\\_Language](http://en.wikipedia.org/wiki/DARPA_Agent_Markup_Language).

Wikipedia/Daml+Oil (2009). *Definição de DAML+OIL*. Retrieved 2009/06/26 from <http://en.wikipedia.org/wiki/DAMLplusOIL>.

Wikipedia/IEEE1471 (2009). *Norma IEEE1471:2000*. Retrieved 2009/09/05 from [http://en.wikipedia.org/wiki/IEEE\\_1471](http://en.wikipedia.org/wiki/IEEE_1471).

Wikipedia/Oil (2009). *Definição de OIL*. Retrieved 2009/06/26 from [http://en.wikipedia.org/wiki/Ontology\\_Inference\\_Layer](http://en.wikipedia.org/wiki/Ontology_Inference_Layer).

Wikipedia/PDCA (2009). *Definição de Ciclo de Deming*. Retrieved 2009/08/05 from <http://en.wikipedia.org/wiki/PDCA>.

Wikipedia/Rdf (2009). *Definição RDF*. Retrieved 2009/06/26 from [http://en.wikipedia.org/wiki/Resource\\_Description\\_Framework](http://en.wikipedia.org/wiki/Resource_Description_Framework).

Wikipedia/Rdfs (2009). *Definição RDF Schema*. Retrieved 2009/06/26 from <http://en.wikipedia.org/wiki/RDFS>.

Youn, S. & McLeod, D. (2006). Ontology Development Tools for Ontology -Based Knowledge Management. In Mehdi Khosrow-Pour (Ed.), *Encyclopedia of E-Commerce, E-Government, and Mobile Commerce*. Hershey, PA, USA: Idea Group Reference. pp. 858-864.

Zachman, J. (1987). A framework for information systems architecture. *IBM Systems Journal*, 3, pp. 276-292.

Zachman, J. (2008). *The Zachman Framework: The Official Concise Definition*. Retrieved 2009/09/06 from <http://www.zachmaninternational.com/index.php/the-zachman-framework>.

Zachman, J. (2009). *The Zachman Framework Evolution*. Retrieved 2009/09/06 from <http://www.zachmaninternational.com/index.php/ea-articles/100#maincol>.

e-Gif (2005). *Framework e-GIF*. Retrieved 2009/09/04 from [http://www.govtalk.gov.uk/schemasstandards/egif\\_document.asp?docnum=949](http://www.govtalk.gov.uk/schemasstandards/egif_document.asp?docnum=949).

## Anexo A Importação de ontologias

### A.1 Introdução

Para seleccionar uma ferramenta de criação de ontologias que permitisse a exportação para a ferramenta de Arquitectura Empresarial, foram efectuadas várias experiências com alguns produtos, as quais são relatadas neste Anexo.

### A.2 OWL Editor

O OWL Editor é um editor de OWL gratuito, desenvolvido pela Model Futures (Owl Editor, 2009). Esta empresa é especializada em Arquitectura Empresarial e Ontologias. Esteve envolvida no desenvolvimento do *Framework* MoDAF (Modaf, 2008).

Esta ferramenta é oferecida sem licença, nem garantia. A versão testada foi a 0.2.0.37, uma versão Alfa. As funcionalidades são limitadas, permitindo a construção de hierarquias de classes, comentar cada classe e juntar atributos do Dublin Core (The Dublin Core Metadata Initiative, 2009). Permite a importação de OWL, EXPRESS XML (gerado pela ferramenta Eurostep EXPRESS Parser – EEP) e XMI (XML Metadata Interchange) versão 1.1. É preferido o XMI criado por exportadores baseados no IDE (Integrated Development Environment) NetBeans. Ao nível da exportação permite enviar um ficheiro CSV (Comma Separated Values), mas a parte mais interessante é a possibilidade de exportação para o modelo SPARX EA, usado no Enterprise Architect.

Foi realizado um teste de exportação de uma ontologia em OWL com o OWL Editor Alfa Test Version para o Enterprise Architect (Business and Software Engineering Edition) versão 7.5.845. A ontologia usada intitula-se Wines e é um dos exemplos distribuídos com o Protégé, tendo sido convertida para OWL. Trata-se de uma ontologia com alguma complexidade e com algum peso, permitindo testar processos de importação/exportação.

Na Figura A.1 pode ser observado o resultado da abertura do ficheiro OWL. A ferramenta suporta a hierarquia de classes da ontologia e as instâncias, mas perde outro tipo de associações e os atributos das classes.

Seguidamente a ontologia foi exportada para o Enterprise Architect. O resultado pode ser visto na Figura A.2, onde é visível apenas uma pequena parte da ontologia. Como se previa só a hierarquia das classes foi convertida. As instâncias também passaram, mas a sua associação às respectivas classes perdeu-se. Note-se a classe isolada que corresponde a um objecto.

# Definição da Arquitectura Empresarial

## O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

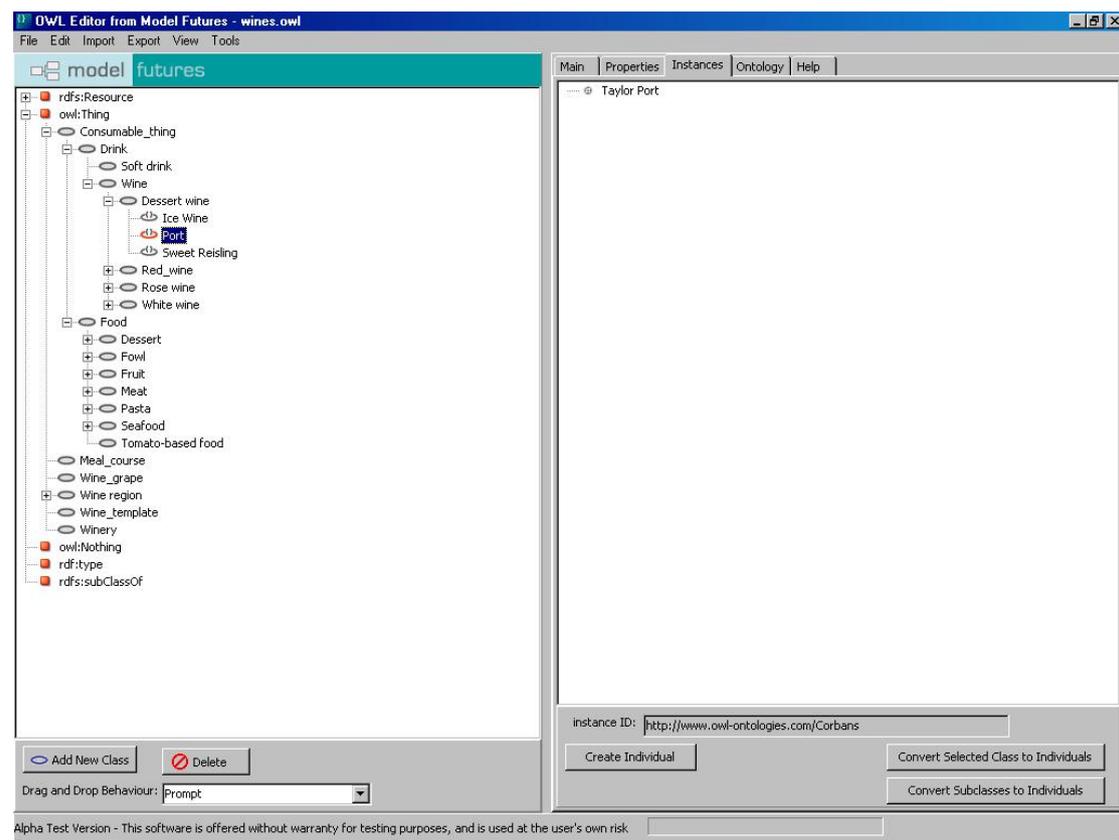


Figura A.1 - Ontologia Wines carregada no OWL Editor.

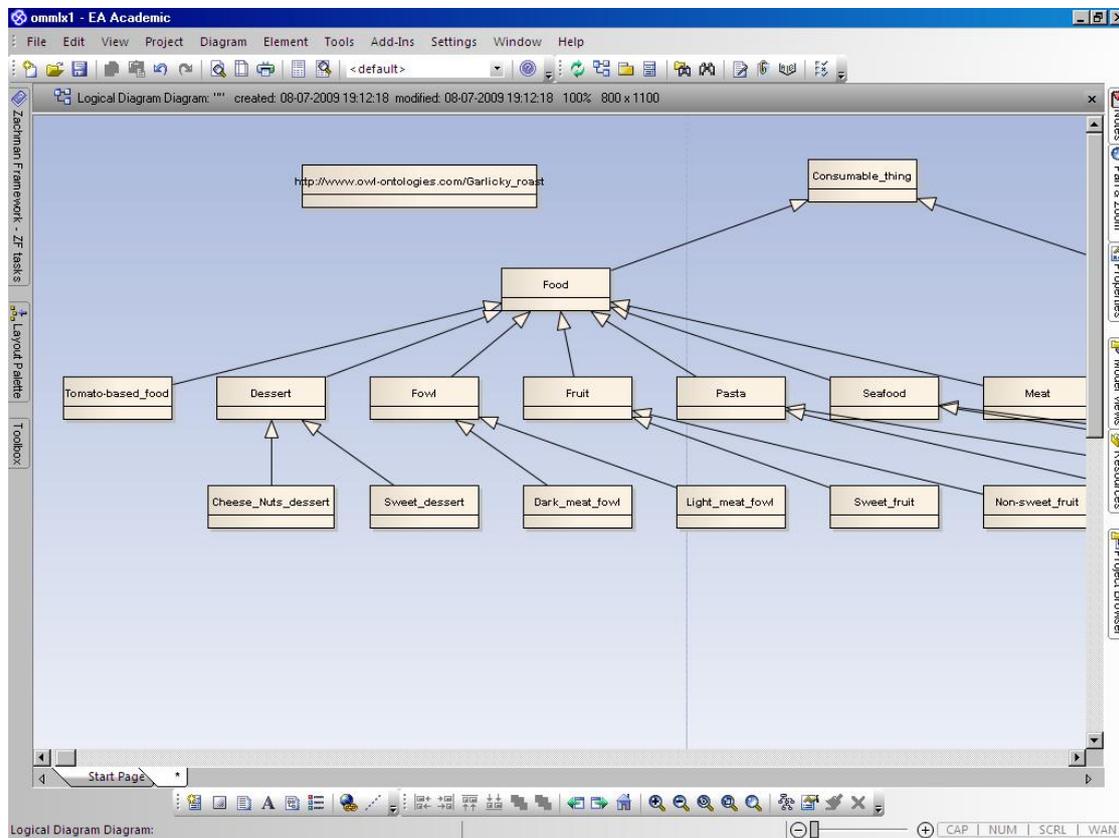


Figura A.2 - Resultado da exportação para Enterprise Architect a partir do OWL Editor.

### A.3 NeOn Toolkit

O NeOn Toolkit surgiu no âmbito do projecto Europeu NeOn IST-2005-027595 (NeOn, 2009) o qual pretende fazer avançar o estado da arte na utilização de ontologias em aplicações semânticas de grande dimensão em organizações distribuídas. O projecto pretende suportar ontologias em rede, criadas colaborativamente e em constante evolução. O NeOn Toolkit é uma ambiente de engenharia de ontologias, construído sobre o IDE Eclipse. Suporta a edição, visualização, consulta e importação/exportação usando as linguagens F-Logic (FLogic, 2009), RDFS e OWL. É complementado por um conjunto de *plug-ins* comerciais que permite expandir a funcionalidade básica. A versão avaliada foi a 1.2.2 build B904.

Foi também usada a mesma ontologia que no caso anterior para testar a exportação. O resultado da edição desta ontologia com esta ferramenta pode ser observado na Figura A.3.

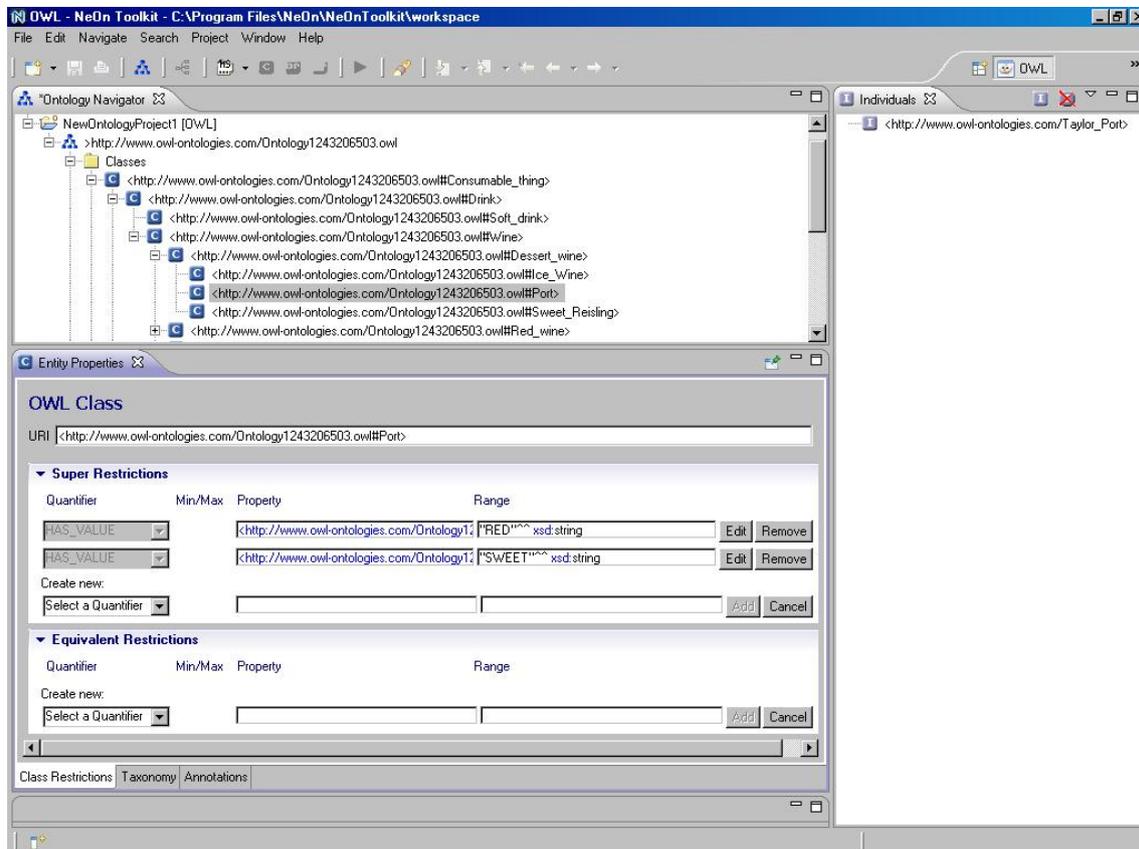


Figura A.3 - NeOn Toolkit mostrando ontologia Wines, classe Port e respectivas propriedades e instâncias.

A exportação para UML só é possível com ontologias que usem F-Logic, pelo que é necessário converter de OWL para F-Logic usando a opção de transformação de ontologias desta aplicação (Transform/Copy Ontology). O resultado desta transformação pode ser observado na Figura A.4.

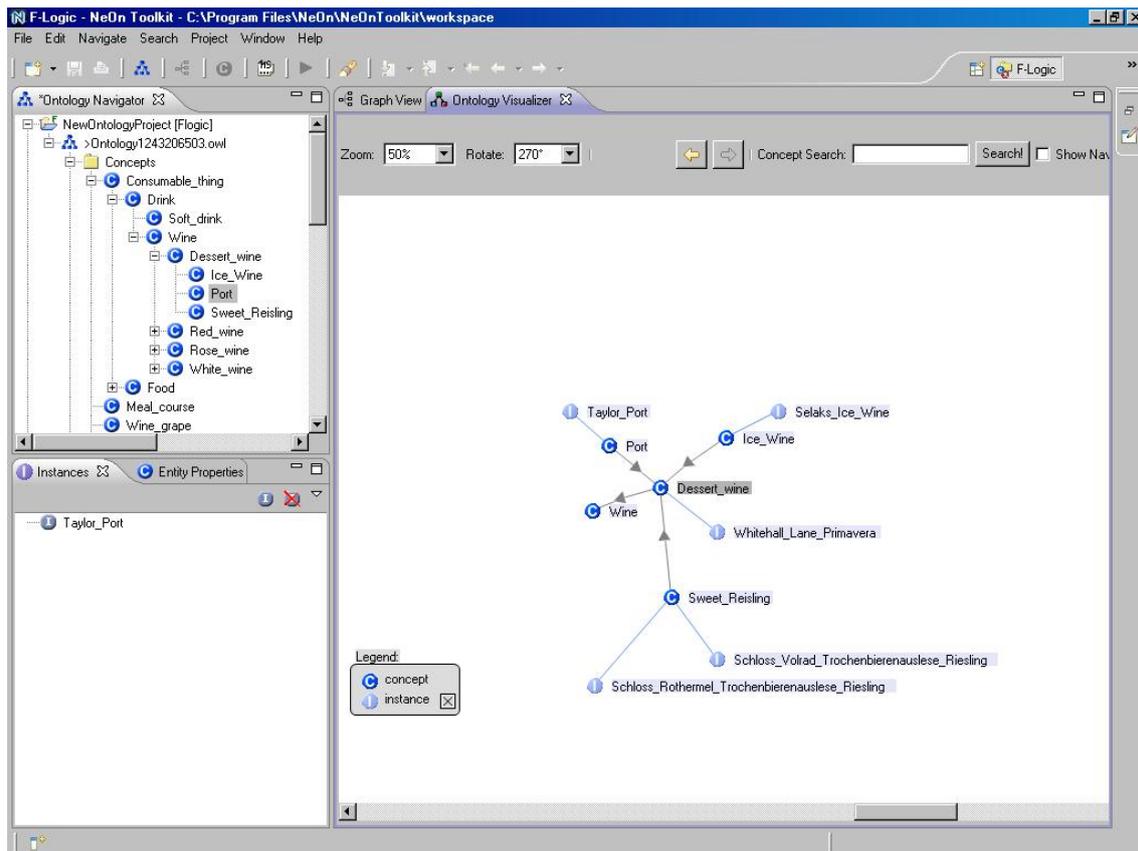


Figura A.4 - NeOn Toolkit mostrando ontologia Wines após conversão de OWL para F-Logic, com visualização de parte da ontologia (vinhos de sobremesa).

Posteriormente foi usada a opção de exportação para UML desta ferramenta. Ao longo do processo perderam-se os atributos das classes e as associações, mantendo-se apenas a estrutura hierárquica da ontologia, graças aos nomes das classes e às suas generalizações. As instâncias são também perdidas. O resultado pode ser observado na Figura A.5, onde podemos ver um diagrama de classes UML com parte da ontologia Wines, após a importação para a ferramenta Enterprise Architect.

Embora na prática não seja representada mais do que a hierarquia de classes da ontologia, esta ferramenta apresenta um potencial interessante devido às suas capacidades de importação/exportação, visualização e número de *plug-ins* disponíveis.

## A.4 Protégé

O Protégé foi desenvolvido pelo Stanford Center for Biomedical Informatics Research da Stanford University School of Medicine (Protégé, 2009). Trata-se de uma plataforma em código aberto gratuita, baseada em Java e extensível, que permite a criação, visualização e manipulação de ontologias.

O Protégé surge actualmente em três variantes: Protégé 3.X, Protégé 4.0 e WebProtégé 0.5 *alpha*. Esta última variante destina-se à elaboração colaborativa de ontologias e ainda está numa versão alfa, sendo muito limitada funcionalmente, particularmente ao nível de exportação. A grande diferença entre as versões 3.X e a 4.0 reside no facto da primeira suportar OWL 1.0, RDFS e Frames, enquanto que a última suporta apenas OWL 2.0, com a agravante de ter muito menos *plug-ins*, particularmente para conversão para UML. Suporta

a possibilidade de representar a ontologia num diagrama UML usando ODM (Ontology Definition Metamodel) (Odm, 2009) do OMG, o que poderá ser interessante em termos de interoperabilidade no futuro. A avaliação focou-se numa primeira fase no Protégé 3.4 e, numa segunda fase, no Protégé 3.4.1, quando este foi disponibilizado.

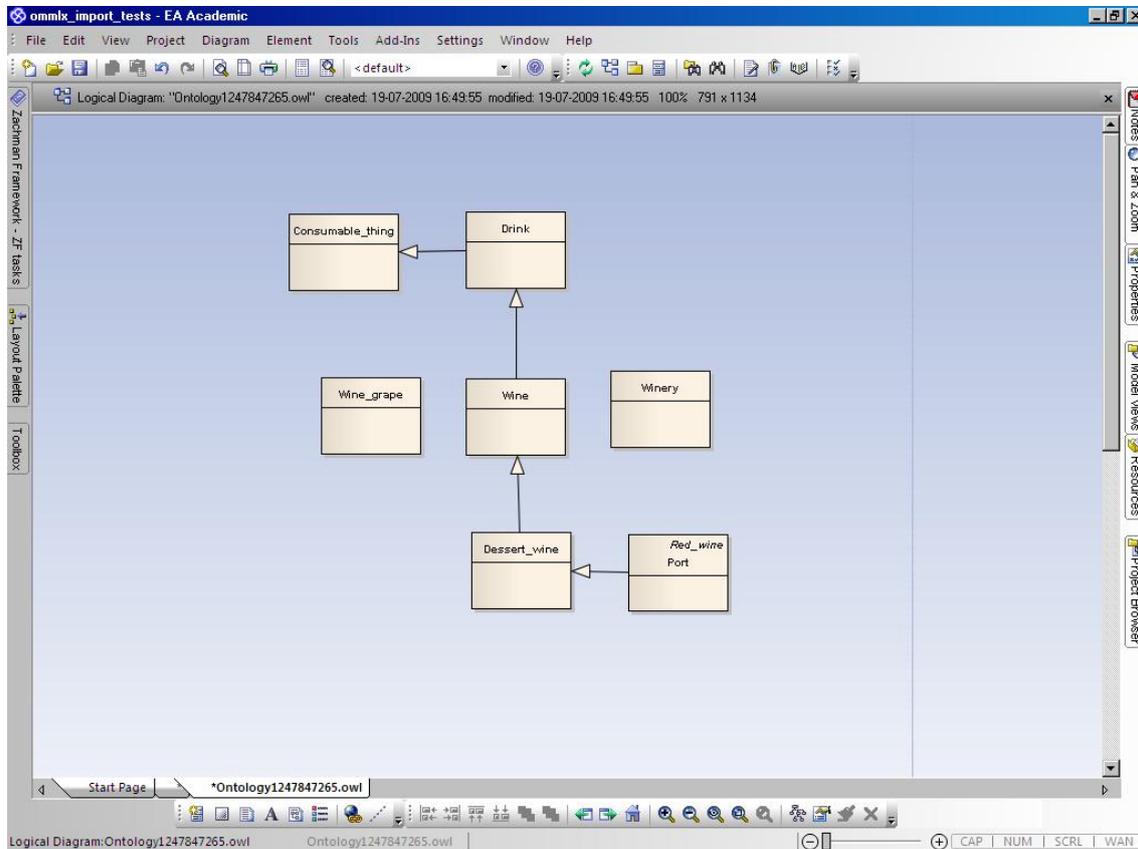


Figura A.5 - Parte da ontologia Wines num diagrama de classes após importação para Enterprise Architect.

Foram realizados vários testes. O primeiro usou o *plug-in* UML Backend, o qual permite exportar para UML 1.4. Na Figura A.6 pode ser observada a ontologia de teste Wines, aberta no Protégé 3.4.1, sendo possível identificar as classes, a hierarquia de classes e os atributos de cada classe. Neste caso a classe Port encontra-se seleccionada e todos os seus atributos (*slots*) encontram-se entre parênteses por serem herdados de superclasses. Para cada atributo, é também possível visualizar a cardinalidade, o tipo e outras facetas, tipicamente restrições, mas também associações.

Na Figura A.7 pode ser observada parte do resultado da exportação em UML 1.4 a partir do Protégé. Como pode ser observado as classes são convertidas, bem como as associações, papéis, cardinalidades, além dos atributos. Infelizmente as generalizações e especializações não são convertidas, perdendo-se a hierarquia de classes da ontologia. Perdem-se também as instâncias.

O segundo teste consistiu na exportação usando RDFS, passando pelo NeOn Toolkit, a partir do qual foi exportado em UML. O resultado pode ser observado na Figura A.8.

## Definição da Arquitectura Empresarial

### O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

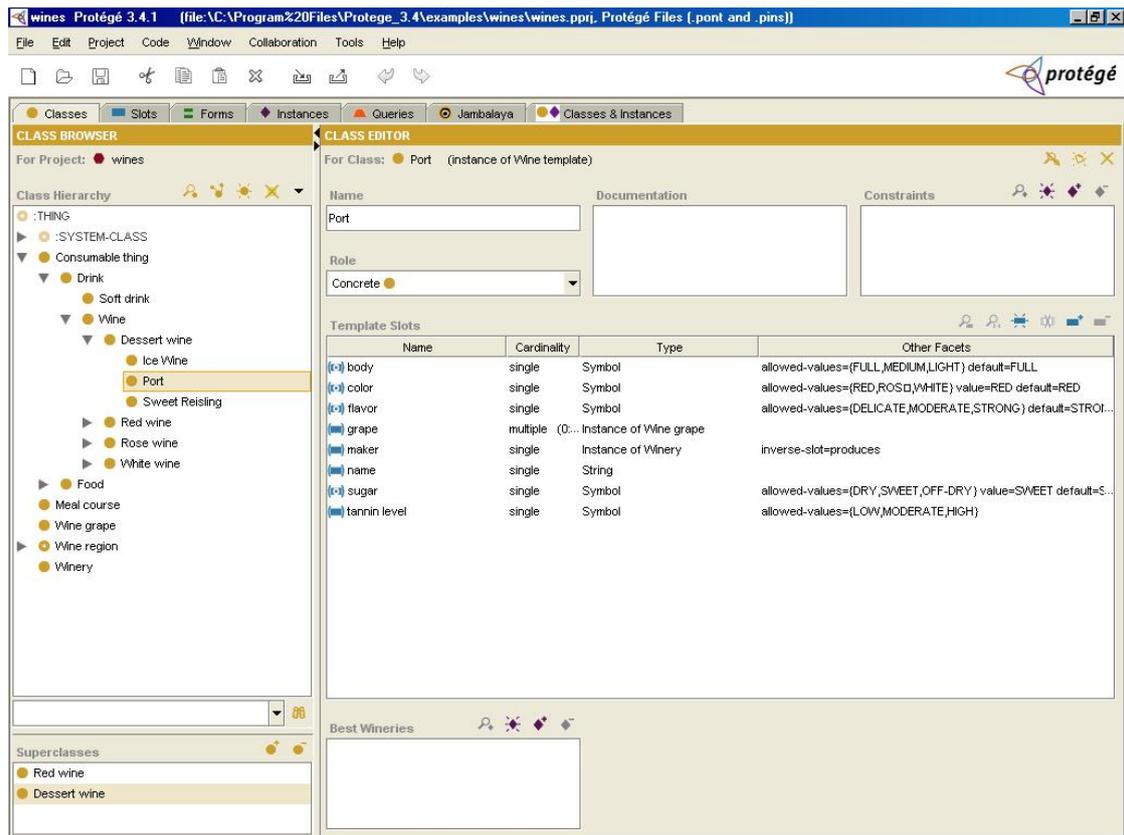


Figura A.6 - Ontologia Wines visualizada no Protégé 3.4.1.

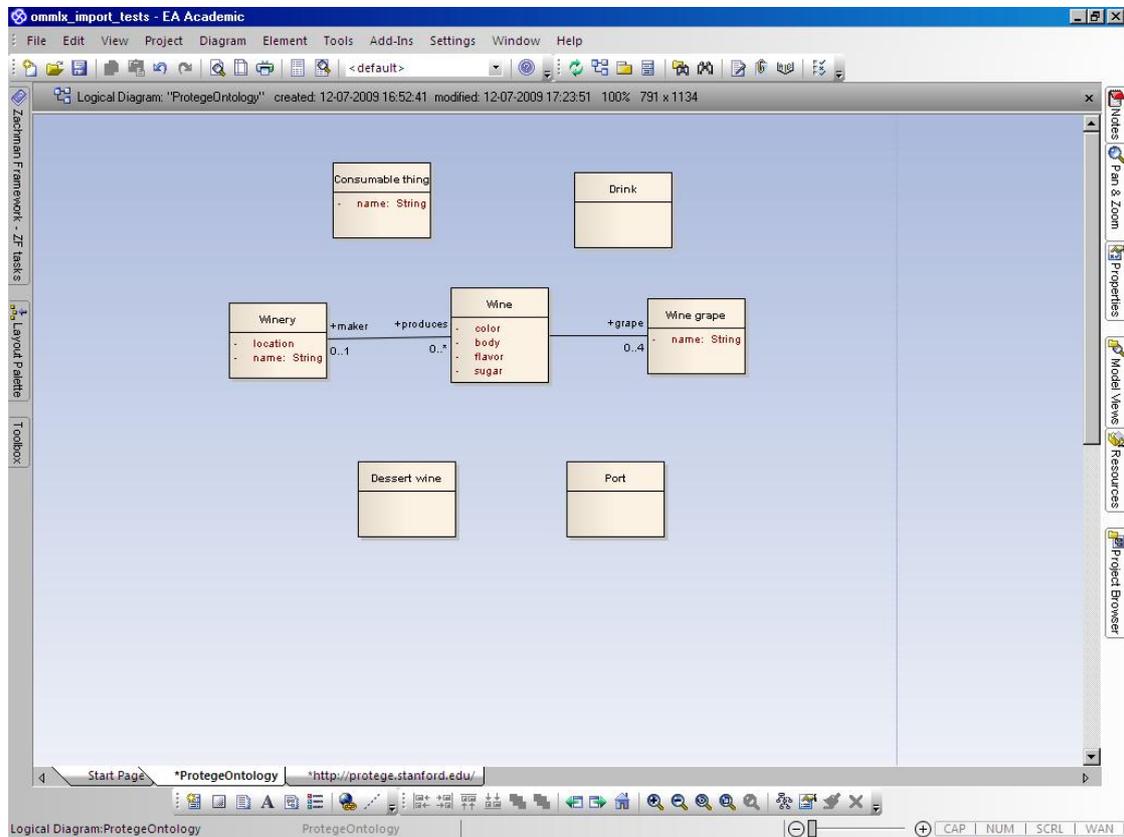


Figura A.7 - Parte do resultado da exportação em UML 1.4 a partir do Protégé.

## Definição da Arquitectura Empresarial

### O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

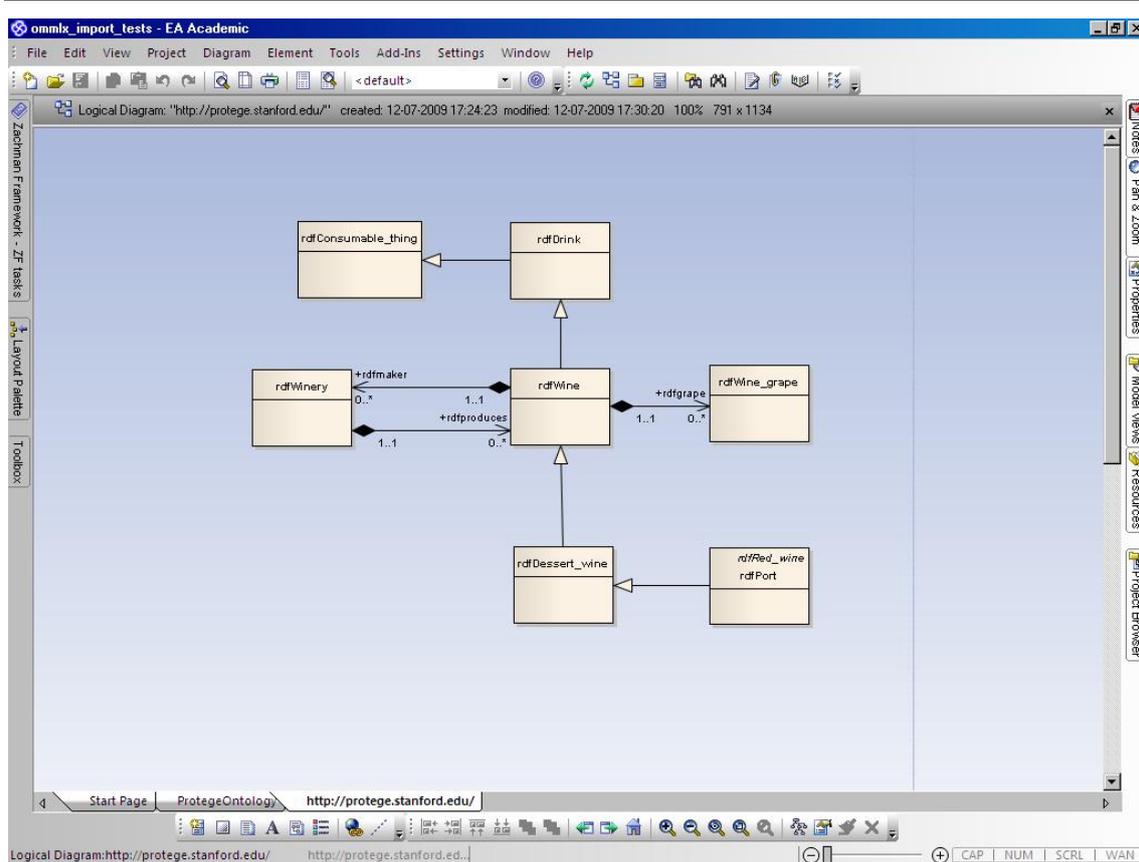


Figura A.8 - Resultado parcial da exportação via NeOn, usando RDFS e UML.

Foi feita também uma importação do ficheiro RDF com as instâncias, mas perdem-se as associações às classes. Como se pode observar na Figura A.8 os atributos não são convertidos, no entanto a hierarquia de classes é convertida e a conversão das associações é bastante rica, sendo convertidas associações duplas entre classes, os respectivos papéis e cardinalidades. Se o *slot* “produtor” e o *slot* inverso “produz” são convertidos em duas associações separadas, a conversão da cardinalidade não está correcta. Com efeito a ontologia indica que um vinho poderá ter no máximo quatro tipos de uva, enquanto que a conversão não especifica limite para o número de tipos de uva. Do mesmo modo é originalmente indicado que um vinho só terá no máximo um fabricante, enquanto que na conversão deixa de haver limite. Note-se que as cardinalidades foram correctamente convertidas no exemplo anterior.

As relações de composição retratadas são fruto do poder das ontologias. Correspondem ao resultado de inferências sobre a ontologia, ou seja, de como podemos extrair conhecimento que não está explícito. Desta forma é possível inferir que as adegas só produzem vinho, que o vinho só é produzido por adegas e que o vinho só tem uvas. Embora estas inferências possam estar logicamente correctas, poderão não corresponder à realidade dos nossos tempos...

O terceiro teste consistiu na passagem da ontologia Wines para OWL e a sua exportação nesse formato para o NeOn toolkit. Seguiu-se a conversão para F-Logic nessa ferramenta e a sua exportação em UML para o Enterprise Architect. O resultado é o descrito no teste associado ao NeOn toolkit. Apenas a hierarquia de classes é convertida.

De forma a verificar se havia algum formato que permitisse uma conversão mais rica, foram testados os restantes formatos disponíveis no Protégé. Em todos estes testes é usado o NeOn como conversor intermédio para UML, implicando uma conversão prévia para F-Logic.

O resultado do teste com o formato TURTLE (Beckett & Berners-Lee, 2008) pode ser observado na Figura A.9. Verifica-se mais uma vez que apenas a estrutura de classes é convertida.

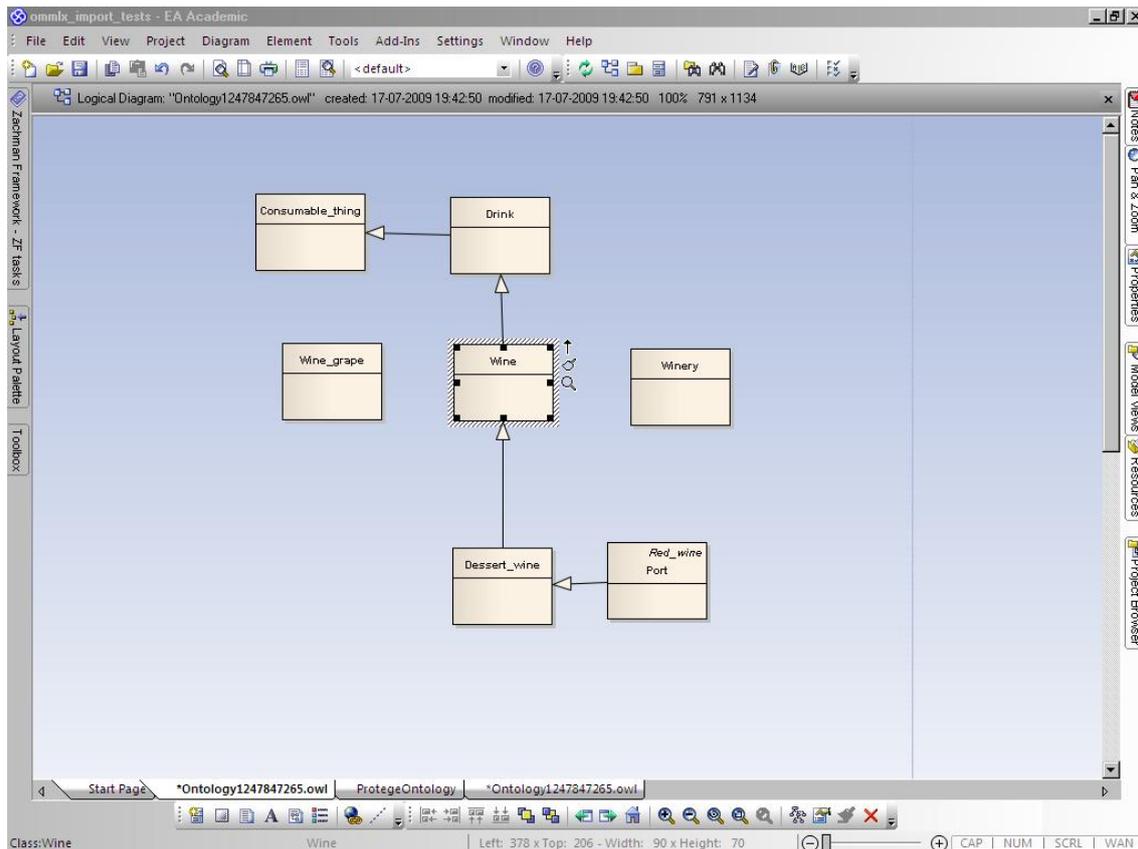


Figura A.9 - Resultado parcial da exportação via NeOn, usando o formato TURTLE.

Os testes seguintes consistiram na utilização dos formatos N3 e NTRIPLE, os quais não tiveram sucesso na exportação final para UML.

Foi também tentado o caminho inverso, ou seja, exportar o diagrama de classes para uma ferramenta de gestão de ontologias, no entanto o XMI exportado pelo Enterprise Architect não foi lido pelo NeOn, o qual seria usado para converter para OWL.

## A.5 Conclusões

Os resultados dos testes são resumidos na Tabela A.1. Analisando as várias alternativas considera-se que esta questão não foi resolvida de forma satisfatória, pois a expectativa era ser obtido um processo de conversão sem intervenção humana. Irá ser tentada uma combinação de duas abordagens, exportando duas vezes e combinando manualmente. A primeira exportação a partir do OWL Editor para o Enterprise Architect e a segunda do

Definição da Arquitectura Empresarial  
O caso do Observatório de Mobilidade na Região Metropolitana de Lisboa

---

Protégé para o Enterprise Architect usando o *plug-in* UML Backend. O ponto de partida é sempre o ficheiro OWL criado no Protégé.

Teste	Diagrama Classes Automático	Classes	Generalizações Especializações Classes (Hierarquia)	Associações	Atributos Classes	Cardinalidade Associações	Instâncias	Papéis Associações
OWL Editor -> EA Model	Sim	Sim	Sim	Não	Não	Não	Erro	Não
NeOn OWL -> Flogic -> UML	Não	Sim	Sim	Não	Não	Não	Não	Não
Protégé OWL -> UML 1.4	Não	Sim	Não	Sim	Sim	Sim	Não	Sim
Protégé OWL -> RDFS -> NeOn -> UML	Não	Sim	Sim	Sim	Não	Erro	Não	Sim
Protégé OWL -> TURPLE -> NeON - Flogic -> UML	Não	Sim	Sim	Não	Não	Não	Não	Não

**Tabela A.1 - Síntese dos testes de importação da ontologia para Enterprise Architect.**

## Anexo B Script da tabela indicador

A título de exemplo anexa-se parte do script criado automaticamente para a geração das tabelas da base de dados, usando transformações MDA. Trata-se da parte correspondente à tabela indicador.

```
DROP TABLE Indicador
```

```
;
```

```
CREATE TABLE Indicador (
```

```
  identifier Memo,
```

```
  title Memo,
```

```
  date Memo,
```

```
  creator Memo,
```

```
  subject Memo,
```

```
  description Memo,
```

```
  publisher Memo,
```

```
  contributor Memo,
```

```
  type Memo,
```

```
  format Memo,
```

```
  source Memo,
```

```
  language Memo,
```

```
  relation Memo,
```

```
  rights Memo,
```

```
  Recolha Memo,
```

```
  coverage Memo,
```

```
  Periodicidade Memo,
```

```
  Unidade Memo,
```

```
  Importação YesNo,
```

```
  Método Memo,
```

```
  indicadorID Integer NOT NULL
```

```
)
```

```
;
```

```
ALTER TABLE Indicador ADD CONSTRAINT PK_Indicador
```

```
  PRIMARY KEY (indicadorID)
```

```
;
```