# REPOX: Obtenção e agregação de dados para Indexação em Bibliotecas Digitais em Rede

Tiago João de Sousa Marques

Instituto Superior Técnico
Av. Rovisco Pais, 1049-1001 Lisboa, Portugal
tiago.marques@ist.utl.pt

**Abstract.** *Com o aparecimento dos Arquivos e das Bibliotecas Digitais surgiu a necessidade de partilhar com toda a comunidade a informação que cada uma destas entidades possui. Esta realidade obrigou à criação de soluções que permitissem a divulgação da informação presente em cada uma destas entidades, de forma a que esta pudesse ser pesquisada de forma simples e rápida. No entanto, devido ao crescimento das novas tecnologias, foram estabelecidos novos objectivos pelas entidades supracitadas, que passam pela possibilidade de pesquisa de informação presente nos textos dos documentos. Assim, este relatório vai-se focar na procura de tecnologias existentes para obtenção e sincronização de dados e de que forma estas podem ser aplicadas para resolver facilitar a recolha dos novos conteúdos de dados, para uma futura indexação num motor de pesquisa.*

**Keywords:** Bibliotecas Digitais, metadados, REPOX, protocolos, obtenção de conteúdos (fulltext), harvest, procura.

# Índice

1. Introdução..... 3
1.1. Motivação..... 3
1.2. Estrutura do documento..... 4
2. O problema e o seu contexto..... 5
2.1. Bibliotecas Digitais..... 5
2.2. Metadados e objectos de digitais..... 5
2.3. O Problema..... 5
2.3.1. Interoperabilidade entre Data Providers e Service Providers..... 6
2.3.2. Interoperabilidade entre Service Providers..... 6
2.3.3. Análise do problema..... 7
2.4. OAI-PMH..... 7
2.5. Questões de Pesquisa..... 9
3. Estado da Arte..... 10
3.1. Sincronizadores de Dados..... 10
3.1.1. SyncML..... 10
3.2. Sincronizadores de ficheiros..... 11
3.2.1. Rsync..... 11
3.2.2. Unison..... 12
3.2.3. SyncToy..... 13
3.2.4. Sync Center..... 14
3.2.5. DropBox..... 15
3.2.6. Live Mesh..... 16
3.2.7. Syncany..... 17
3.3. Software de controlo de versões..... 17
3.3.1. GIT..... 18
3.4. Data-sharing middleware..... 19
3.4.1. IceCube..... 19
3.4.2. Semantic Chunks..... 20
3.4.3. Xmiddle..... 21
3.4.4. Microsoft Sync Framework..... 22
3.5. Análise..... 24
4. Solução a Explorar..... 25
5. Proposta de Validação de Resultados..... 29
6. Planeamento de trabalho..... 30
7. Conclusão..... 30
8. Referências..... 31

# 1 Introdução

## 1.1 Motivação

O aparecimento de bibliotecas e arquivos digitais, gerou um grande interesse na partilha da informação descritiva dos registos existentes, por partes das mesmas em projectos internacionais, como a Europeana[1], TEL[2] e EuDML[3]. Estas entidades têm como objectivo a recolha desta informação descritiva, de forma a facilitar o seu processo de pesquisa, criando as condições necessárias de acesso dos mesmos aos utilizadores, sem que estes necessitassem de analisar todas as fontes de informação separadamente. Estas iniciativas vieram desta forma colmatar uma lacuna correspondente à inexistência de um ponto centralizado de pesquisa, facilitando o acesso à informação por parte de investigadores, alunos ou de qualquer outro utilizador que assim o deseje.
Tendo em conta os cenários descritos, as organizações dispostas a partilhar os dados muitas vezes encontram dificuldades em fazê-lo, devido aos seus sistemas informáticos não suportarem nativamente o protocolo OAI-PMH[1], que é um requisito comum associado aos projectos internacionais supracitados. Esta tecnologia é suportada por várias soluções, tanto comerciais como open-source, mas em muitos casos é difícil fazer os investimentos necessários para a compra de soluções pagas, ao passo que a utilização de software open-source normalmente necessita de alguma adaptação do mesmo a realidade da entidade que o vai utilizar. Na maioria dos casos estas não possuem, nos seus quadros de trabalhadores, as capacidades técnicas para efectuar essas alterações o que poderá novamente implicar investimentos.
Assim o OAI-PMH é um protocolo baseado na arquitectura Cliente-Servidor, usando XML sobre HTTP. Este assenta, como pretendido na conferência de Santa Fé[2], na distinção clara entre o que são **Data Providers** e **Service Providers**.
Data providers são as entidades que possuem informação (metadados) e estão dispostas a partilhá-las com os outros. Estas disponibilizam a informação sem qualquer tipo de custos, sendo que podem mesmo oferecer acesso a outros tipos de conteúdos, como textos ou imagens, mas que não tratados por este protocolo.
Service providers passam por ser as entidades que agregam a informação proveniente dos Data providers e a disponibilizam a toda a comunidade, através da introdução de serviços que permitem visualizar a informação a um nível mais alto (motores de busca ou browsers por exemplo).
Com o intuito de facilitar a partilha de dados entre as bibliotecas digitais e os projectos que promovem a agregação destes mesmos, foi criada uma plataforma open-source de nome REPOX. Esta é uma framework que visa simplificar os processos de

1 Europeana –The European Digital Library -http://dev.europeana.eu/
2 TEL - The European Library- http://www.theeuropeanlibrary.org
3 EuDML– European Digital Mathematics Library- http://www.eudml.eu/

partilha e recolha de dados a qualquer uma das entidades mencionadas anteriormente, pois possui uma instalação e configurações fáceis, diminuindo assim o esforço a níveis técnicos que possa ser requerido, aumentando em muito a simplicidade de iniciação na partilha e recolha dos dados.
Esta framework de gestão de metadados dispõe actualmente de tecnologias para:

- Aquisição e armazenamento de metadados provenientes de diferentes fontes, utilizando os seguintes protocolos HTTP, FTP, o OAI-PMH e o Z39-50;
- Transformação de metadados de acordo com especificações, regras e modelos de cada entidade;
- Apresentação e disponibilização de informação adquirida;

No entanto com o crescimento das capacidades e dos recursos informáticos, têm surgido novos cenários de partilha e recolha de metadados, nomeadamente a transferência não só dos metadados produzidos pelas bibliotecas e arquivos digitais, mas também os conteúdos referenciados por estes mesmos dados, como é o caso de imagens ou documentos dos mais variados tipos (ex: artigos científicos, jornais, revistas, etc), ainda vídeos ou áudio. Estes são novos desafios que abrem um novo conjunto de requisitos que irão impor suporte a novos processos de recolha de agregação de informação.
Assim, para concluir, o que se pretende desta tese passa por encontrar uma solução tecnológica eficiente que permita processar a recolha de conteúdos, dando garantias desta ser eficaz e escalável.

### 1.2 Estrutura do documento

No seguimento da Introdução na secção 1, está O Problema e o seu Contexto, na secção 2, onde é apresentada toda a problemática e serão extraídas as questões que serão analisadas na execução desta tese. Na secção 3 é apresentado o Estado da Arte, onde são investigadas as tecnologias e soluções que de alguma forma podem ser aplicadas para a resolução dos problemas que são apresentados na secção 2. Seguidamente na secção 4 é apresentada uma proposta de solução que se pretende analisar durante a execução da tese. Nas secções seguintes (5 e 6), serão apresentados a Proposta de Validação, onde será descrito de que forma se pretende validar os dados obtidos e o Planeamento de Trabalho onde será apresentada a forma como se pretende que o trabalho seja desenvolvido e os tempos que serão despendidos em cada fase. Para finalizar será apresentada uma Conclusão onde será feita um resumo de todo o trabalho que já foi efectuado e as dificuldades que ainda são esperadas na execução do resto da tese.

# 2 O problema e o seu contexto

Neste capítulo vai ser analisada toda a problemática da tese, começando por analisar o que são bibliotecas digitais seguindo-se uma demonstração dos cenários previstos para a implementação do novo paradigma de recolha de dados. Logo após será feita uma análise mais profunda ao OAI-PMH, para assim se perceber a tecnologia de transferência de dados mais utilizada no âmbito da disponibilização de dados pelas bibliotecas digitais. Para terminar irão ser descritas quais as principais questões resultantes da análise do problema.

## 2.1 Bibliotecas Digitais

As bibliotecas digitais podem ser definidas como "uma colecção de objectos digitais, que inclui texto, vídeo e áudio, bem como métodos de acesso e obtenção de dados, e ainda para organização e manutenção de colecções de dados."[3] Esta definição permite perceber que as bibliotecas são entidades, que passaram de meros aglomeradores e organizadores de informação a "criadores" de objectos digitais, partindo dos registos que estas possuam no seu espólio, de forma a possibilitar não só uma nova forma de preservação dos mesmos, mas ao mesmo tempo uma nova forma de partilha destes conteúdos.

## 2.2 Metadados e objectos de digitais

A explosão do World-Wide Web possibilitou a múltiplos agentes, a disponibilização dos seus dados na Internet, de forma a estes serem pesquisáveis por todos os tipos de utilizadores. No entanto a tarefa de pesquisa de recursos que possuam relevância para a resolução de determinado problema pode ser morosa e muitas vezes pode implicar interacções entre várias fontes de informação, de forma a responder ao problema proposto. Para facilitar as tarefas supracitadas foi criado o conceito de **metadados**, em que os primeiros servem para caracterizar de forma explícita os últimos. No âmbito das bibliotecas e arquivos digitas este conceito é utilizado como descritor de informação dos recursos disponíveis, sendo que normalmente descreve o título, a data de criação, o autor, etc do documento preservado por estas instâncias[1].

## 2.3 O Problema

Actualmente existem várias entidades dispostas a partilhar os seus recursos catalogados através de metadados, sendo exemplos disso o projecto Europena

1 http://dublincore.org/documents/2001/04/12/usageguide/generic.shtml

initiative[1], bibliotecas, museus e arquivos. O objectivos principais que levam ao interesse na partilha de recursos passam pela preservação dos mesmos e criação de estruturas de pesquisa de dados.
Em termos da preservação dos dados é importante ter mais do que uma cópia dos dados, daí surge a necessidade de um sistema de sincronização de dados entre as entidades que irão fazer a sua manutenção e preservação e as fontes de dados, de modo a tornar este processo mais simples e automatizado.
Relativamente à criação de estruturas de pesquisa de dados, estas têm por objectivo a utilização dos conteúdos de forma a torná-los pesquisáveis, sejam estes provenientes dos metadados, ou dos objectos referenciados pelos metadados (especialmente fulltext), passando este ultimo tópico a ser uma nova realidade, que introduz desafios os quais se pretende estudar ao longo desta tese.
Tendo em conta os objectivos supracitados, é possível prever alguns cenários de interoperabilidade por parte das entidades que partilham a informação e as que fazem a recolha e agregação da mesma. É nesta aplicação que se conseguem vislumbrar os problemas que são a base para esta tese.
Assim os principais caso de uso que foram antecipados são a interoperabilidade entre Data Provider e Service Provider e a interoperabilidade entre Service Providers.

### 2.3.1 Interoperabilidade entre Data Providers e Service Providers

O normal funcionamento do sistema de agregação de dados, passa por um Data Provider publicar metadados para que o Service Provider equipado com a tecnologia REPOX possa fazer a recolha dos mesmos, e tendo em conta o novo paradigma previsto de recolha de conteúdos. existem três cenários a ter em conta para esta secção. São eles:

- Interoperabilidade entre um Service Provider e um Data Provider que publica os metadados por OAI-PMH normal e os conteúdos por HTTP ou FTP;
- Interoperabilidade entre um Service Provider e um Data Provider que publica os metadados por OAI-PMH normal e os conteúdos por tecnologia recomendada pelo projecto REPOX;
- Interoperabilidade entre um Service Provider e um Data Provider que publica os metadados e os conteúdos por tecnologia recomendada pelo projecto REPOX.

### 2.3.2 Interoperabilidade entre Service Providers

Considerando desta vez cenários de interoperabilidade entre dois Service Providers em que um deles utiliza a tecnologia REPOX, passando a funcionar como agregador

1 Europeana - http://www.europeana.org/

de dados, ou seja um Data Provider, existem novamente três cenários a ter em conta. São eles:

- Interoperabilidade entre um Service Provider e outro Service Provider que recolhe os metadados por OAI-PMH normal e os conteúdos por HTTP ou FTP;
- Interoperabilidade entre um Service Provider e outro Service Provider que recolhe os metadados por OAI-PMH normal e os conteúdos por tecnologia recomendada pelo projecto REPOX;
- Interoperabilidade entre um Service Provider e outro Service Provider que recolhe os metadados e os conteúdos por tecnologia recomendada pelo projecto REPOX;

### 2.3.3 Análise do problema

Analisando os cenários anteriores, é possível verificar que só em alguns deles é possível controlar a forma como como os dados são publicados e recolhidos, o que restringe de alguma forma o âmbito do problema, pois nesses casos é apenas possível fazer optimizações nas entidades que utilizem tecnologias recomendadas pelo projecto. Ainda assim é possível retirar alguns desafios interessantes, nomeadamente de como efectuar o processo de recolha de conteúdos de forma a evitar o download de colecções inteiras de ficheiros, quando existirem apenas pequenas actualizações.
Outra questão de valor pode passar pela optimização do processo de recolha dos metadados, mas para isso vai ser analisado um pouco mais a fundo o protocolos OAI-PMH.

## 2.4 OAI-PMH

O protocolo OAI-PMH é o resultado de um projecto desenvolvido pelo Open Archives Initiative (OAI), cuja sua principal actividade é centrada na resolução de problemas de interoperabilidade entre entidades, nomeadamente bibliotecas digitais e arquivos, pela criação e desenvolvimento de protocolos e standards para a disseminação de conteúdos[1].
A criação do OAI teve início em Outubro de 1999 na Convenção de Santa Fé[2] e as motivações existentes para esta foram :

- O rápido crescimento da Internet e a adopção das entidades escolares como meio de partilha de resultados;
- A morosidade tradicional da publicação do avanços feitos nos vários âmbitos escolares;
- A problemática da transferência de direitos dos autores para as editoras, introduzindo problemas de divulgação de resultados;

1 http://www.openarchives.org/OAI/OAI-organization.php

- Os atrasos ou a supressão de novas ideias provocados pelas revisões de pares, favorecendo em muitos casos publicações provenientes de instituições de maior prestígio, em detrimento de outras;
- Os preços das assinaturas por parte das bibliotecas tornou-se demasiado dispendioso.

Assim o objectivo inicial da OAI passava por alcançar repositórios de arquivos digitais de e-print que contivessem trabalhos de pesquisa guardados, tendo definido como propósito da convenção as seguintes predisposições:

- Criação de uma framework que permitisse a descoberta e partilha rápida dos conteúdos supracitados;
- Fornecimento de recomendações técnicas para a criação de arquivos que correspondessem as características do ponto anterior;
- Distinção clara entre o que são Data Providers e Service Providers.

Tendo observado a complexidade da problemática em análise, foi decidido que da convenção apenas iriam ser apresentadas soluções para a recolha de metadados, sendo estas as seguintes:

- Definição de um conjunto de elementos de metadados – Open Archives Metadata Set (OAMS), sendo alguns deles qualificados, para permitir a pesquisa de documentos nos arquivos;
- Uso de XML como sintaxe de transporte e representação;
- Utilização de um protocolo de transferência de dados comum.

Destes pressupostos nasce o OAI-PMH (Open Archives Initiative – Protocol for Metadata Harvesting), cujo o principal objectivo era ter a capacidade de transferir metadados dos arquivos fonte para os arquivos destino. Este protocolo nasce de uma prova de conceito feita com o protocolo Dienst[2], tendo evoluído até a versão usada actualmente, a versão 2.0 lançada em 2006. Nesta versão houve uma revisão do protocolo, sendo que a maior diferença passa pela forma como se aborda a informação, deixando de ser apenas documentos para passarem a ser recursos.

Como já foi mencionado anteriormente este protocolo tem como principal função a obtenção de metadados, sendo que não suporta qualquer tipo de pesquisa directamente sobre a informação obtida. A obtenção ou recolha de dados feita pelo OAI-PMH pode ser total ou selectiva, sendo que se pode obter informação como um todo ou apenas podem ser obtidos pequenas porções de informação, mediante as necessidades do service provider. Este processo é baseado em dois critérios, **sets** e **datestamps**, que podem ser aplicados em conjunto ou individualmente, sendo que no primeiro caso todos os dados pertencentes a um conjunto ou set serão obtidos, ao contrário do que acontece com a recolha por datestamp onde a é apenas recolhida informação que tenha sido alterada durante um período de tempo específico.

Os pedidos são tratados através do métodos GET e POST do HTTP, sendo que as respostas a estes são sempre devolvidas em formato XML codificadas em UTF-8,

2 http://www.cs.cornell.edu/cdlrg/dienst/protocols/DienstProtocol.htm

tendo em atenção o esquema escolhido pela entidade para a codificação dos dados. Este protocolo tem a capacidade de tratar múltiplos tidos de metadados, mas o Dublin Core[1] passa por ser o imperativo para manter o propósito da interoperabilidade que este protocolo apresenta.
Para finalizar, em termos de estruturação de dados ao nível dos repositórios, o OAI-PMH lida com três tipos de estruturas de informação:

- Resources (recursos) – informação a partir da qual são gerados os metadados utilizados pelo OAI-PMH;
- Item – entidade mais abstracta do OAI-PMH, sendo o ponto de entrada para a disseminação da informação de um resource;
- Record (registo) – contém os metadados de um item, codificados num formato específico de XML (Dublin Core, MARCXML[2], METS[3]).

### 2.5 Questões de Pesquisa

Com base na análise feita anteriormente ao problema é possível inferir as seguintes questões, que serão a base para esta tese.

1. **Que técnicas são mais eficientes para a recolha e sincronização de objectos de conteúdos referenciados pelos metadados (especialmente fulltext), considerando tanto os cenários de uma primeira recolha como os de sincronização futura em caso de alterações?**
2. **Será possível optimizar o processo de recolha e sincronização de metadados, passando este a ser mais rápido e eficiente que o processo omissão com os servidores e clientes OAI-PMH actuais?**

## 3 Estado da Arte

Tendo em conta os objectivos definidos no âmbito das bibliotecas digitais e dos projectos de agregação de dados e as questões levantadas na secção anterior, é possível definir quais as soluções existentes que possam ajudar a solucionar as mesmas. Assim para as questões relativas à possibilidade de optimizar o processo de recolha e sincronização de metadados e para a questão sobre as técnicas de recolha e sincronização de objectos (fulltext) pretende-se apresentar algumas soluções focando

1 Dublin Core Metadata Initiative(DCMI), http://dublincore.org/
2 MARC 21 XML Schema, http://www.loc.gov/standards/marcxml/
3 Metadata Encoding and Transmission Standard (METS) Official Web Site, http://www.loc.gov/standards/mets/

os seguintes temas: Sincronização de Dados[1] Sincronização de Ficheiros[2] e Data-sharing Middleware.

### 3.1 Sincronizadores de Dados

Nesta secção irão ser abordadas as tecnologias relativas à sincronização de dados e que possam relevantes no âmbito do REPOX.

#### 3.1.1 SyncML

O SyncML[4][3] (Synchronization Markup Language) é um protocolo de sincronização de informação independente da plataforma ou dispositivo utilizado, definido pela Open Mobile Alliance[4]. O principal objectivo deste protocolo passa por ser a facilidade e eficiência da sincronização de dados, sendo principalmente utilizado para sincronizar email, calendários, contactos, etc. entre diversos tipos de dispositivos, móveis (PDA, computador portátil) ou fixos (PC ou servidor por exemplo). Esta iniciativa está associada a algumas das maiores empresas do ramo das telecomunicações e software como a Ericsson, Nokia, IBM, Motorola e a Symbian, assim como algumas empresas de comunicações wireless.
A arquitectura do SyncML baseia-se em dois objectivos :

- Sincronização de dados presentes em rede com os dados de qualquer dispositivo móvel;
- Sincronização de dados presentes em dispositivos móveis com dados existentes em rede.

Estes objectivos obrigaram a que este protocolo tivesse características muito bem definidas, sendo estas claramente as mais valias presentes no SyncML:

- Funcionamento através de redes wireless e redes físicas;
- Suporte a múltiplos protocolos de transporte de dados, como o HTTP, WAP, SMTP, etc;
- Capacidade de transporte de dados independente do formato;
- Permissão de acesso de acesso aos dados por diferentes tipos de aplicações;

---

1 http://en.wikipedia.org/wiki/Data_synchronization
2 http://en.wikipedia.org/wiki/File_synchronization
3 The SyncML Initiative – http://xml.coverpages.org/syncML.html

Synchronization Markup Language – http://wiki.horde.org/SyncML/

http://www.openmobilealliance.org/tech/affiliates/syncml/syncmlindex.html

http://developers.sun.com/mobility/midp/articles/syncml/

4 http://www.openmobilealliance.org/AboutOMA/Default.aspx

- Cuidado com as limitações dos dispositivos móveis, nomeadamente ao nível da quantidade de dados que estes podem albergar.

Em conclusão o SyncML é direccionado especialmente para os requisitos presentes no "mundo wireless", pois este minimiza o uso de largura de banda necessária para a transferência de dados e consegue lidar com desafios de sincronização associados a redes de baixa qualidade e de alta latência.

## 3.2 Sincronizadores de ficheiros

Nesta secção irão ser abordadas as tecnologias relativas à sincronização de ficheiros, que possam ser relevantes para o tópico em estudo ao longo desta tese.

### 3.2.1 Rsync

O Rsync[5] é um programa open source, desenhado para funcionar em qualquer plataforma, seja ela Windows, Linux ou Mac. Esta aplicação tem como principal funcionalidade a sincronização de pastas e de ficheiros, que estejam presentes em localizações diferentes. Esta tecnologia foi desenvolvida com o seu foco nas redes de baixa capacidade, onde a largura de banda é reduzida e existe uma elevada latência na transmissão dos ficheiros. Para isso utiliza formas para reduzir a quantidade de informação que é transmitida, cumprindo assim os requisitos a que se propõe.
O funcionamento deste sistema começa pela subdivisão de um ou mais ficheiros em pedaços que não se sobreponham de tamanho fixo (que podem ir de 500 a 1000 bytes), sendo que os últimos blocos poderão ter um tamanho inferior ao tamanho dos blocos utilizados. Estes blocos passam por uma fase onde são submetidos ao cálculo de checksums. De cada um destes blocos é obtido um checksum mais fraco de 32 bits, também conhecido por Rolling Checksum e um segundo, mais forte, de 128 bits o MD5 checksum. Em fases iniciais do projecto foi utilizado o MD4, tendo sido substituído posteriormente pelo MD5. Terminada a fase obtenção dos valores do checksums de cada bloco este são enviados a outro computador que contem informação que necessita de ser actualizada, onde estes são comparados com os checksums fracos e fortes dos blocos dos ficheiros existentes nesta localização. Desta comparação será recolhida informação que irá servir para a fonte produzir uma solução (conjunto de instruções) que permita a actualização da informação no local onde se encontram as réplicas. As instruções podem ser constituídas ou por referencias a blocos de informação ou por blocos de informação mesmo, sendo esta última apenas enviada quando é necessário inserir novos dados nos ficheiros.
É possível ainda através do Rsync fazer pequenas configurações no processo de sincronização de dados para que possam melhorar o seu funcionamento. Algumas destas passam pela possibilidade de comprimir e descomprimir blocos que irão ser enviados e ainda de cifrar a informação enviada com uso de protocolos como o SSH. É possível através das hipóteses de configuração do rsync de limitar a largura de

banda utilizada pelo programa, controlando assim todo o fluxo de dados e permitindo adaptar o funcionamento desta software as condições que as ligações entre os utilizadores possam apresentar.
Em conclusão pode ser verificado que este sistema de sincronização de informação apresenta grandes benefícios para os seus utilizadores pois, como já foi demonstrado anteriormente. Este apresenta uma grande flexibilidade e adaptação a vários meios e condições de partilha de informação, devido as características que o definem, sendo a principal a baixa quantidade de informação que este necessita de enviar aquando de uma actualização de dados. Devido a estas particularidades esta tecnologia é muito utilizada em vários contextos, tendo sido reaproveitada para vários outros projectos como poderá ser visto ao longo deste documento.
De referir ainda a existência de algumas variações da implementação do rsync, sendo feitas algumas alterações tendo em vista casos mais específicos de utilização do mesmo sistema, mas sempre com o objectivo de o tornar mais eficiente em todos os aspectos. Assim algumas destas são o In-Place Rsync [10] e Multiround Rsync [11].

### 3.2.2 Unison

Unison [6] é um sincronizador de ficheiros open source, que um pouco à semelhança do SyncML, tem como principais objectivos ser portável, estável e robusto e não menos importante, a sua utilização ser transversal em vários sistemas operativos e arquitecturas, como são casos o Unix e Windows. Este facto permite que seja possível que num servidor com Windows esteja sincronizado com um computador portátil com um sistema Unix. A sincronização é feita através do algoritmo de rsync, definido na secção anterior, sendo este usado para evitar desperdícios no envio de informação. Este apenas envia partes dos ficheiros que necessitam de ser actualizados e não todo o seu conteúdo, aumentando assim a velocidade de sincronização.
Esta ferramenta apresenta um conjunto de características muito interessantes que a tornam única:

- Sincronização de ficheiros entre de diferentes plataformas. Problema com a utilização de nomes. que em diferentes sistemas podem ser considerados ilegais;
- Capacidade de restauro de documentos, por utilização de um sistema de cópias. O número de cópias é limitado para poupar espaço de armazenamento;
- Detecção de conflitos, em casos de alterações do mesmo ficheiro em fontes diferentes, sendo o utilizador informado do acontecimento;
- Possibilidade de resolução de conflitos através da chamada de aplicações externas;
- Sincronização entre duas ou mais máquinas feita por TCP/IP, o que permite a utilização de sockets ou o protocolo SSH, sendo o segundo um fonte de comunicação mais segura e por a isso a aconselhada para o efeito;

- robustez e resistência a falhas de comunicação ou crash de sistema.

### 3.2.3 SyncToy

O SyncToy [1] é uma ferramenta desenvolvida pela Microsoft, para os seus sistemas operativos Windows XP, Vista e 7, com o intuito de automatizar a sincronização de ficheiros e pastas num mesmo computador, num dispositivo externo (pen usb por exemplo) ou ainda numa outra máquina existente na mesma rede. Uma utilização típica desta aplicação passa pela partilha de ficheiros, como fotografias ou músicas e criação de cópias de segurança de dados presentes num computador.
O funcionamento deste sistema passa normalmente pela sincronização de duas pastas (pasta da esquerda e da direita) que como foi dito anteriormente podem estar alocadas em diferentes dispositivos e localizações de rede. Assim são disponibilizadas três hipóteses de métodos de sincronização:

- Synchronize: este método certifica-se que ambas as pastas têm os mesmos ficheiros, o que pode implicar cópias, remoções e alterações de nomes a ficheiros em ambas as pastas, garantindo que um estado de igualdade entre estas seja atingido;
- Echo: este método preocupa-se em procurar as diferenças existentes entre a pasta da direita e a da esquerda, alterando posteriormente a pasta do lado direito para garantir a sincronização das mesmas;
- Contribute: este processo é em tudo similar ao método anterior, sendo que se distinguem por este não propagar remoções de ficheiros para a pasta da direita, preocupando-se apenas com a adição de novos ficheiros e de alterações de nomes de ficheiros existentes.

Este sistema de sincronização permite ainda fazer uma previsão do estado final por aplicação da sincronização, sem realmente ter feito a sincronização, permitindo verificar as modificações propostas, dando a possibilidade de alterar as acções que estas iriam ser executadas previamente à execução das mesmas, diminuindo assim o risco de perda de informação. De referir ainda que aquando de uma sincronização com remoção de ficheiros, é possível configurar o SyncToy para que este envie os ficheiros removidos para a Reciclagem do sistema, aumentando assim a segurança no tratamento dos dados e sendo esta mais uma técnica importante para a prevenção de perdas de informação.
Em termos de resolução de conflitos, a forma como estes são resolvidos depende das acções definidas no momento da criação do par de pastas que é necessário sincronizar, sendo passíveis alguns cenários de resolução por estas acções a alteração do nome de um ficheiro em ambas as pastas, a remoção do ficheiro numa pasta e a

1 http://www.microsoft.com/download/en/details.aspx?DisplayLang=en&id=8358
http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=15155

alteração do seu nome noutra ou ainda a mudança de nome de um ficheiro numa pasta e a sua alteração noutra.
Para finalizar o processo de sincronização entre duas pastas este sistema promove ainda a captura de uma imagem final das pastas (snapshot), que irá conter informação relativa aos ficheiros que integram as pastas, os seus tamanhos, datas de alteração, etc, o que irá permitir uma maior facilidade na previsão por parte do SyncToy aquando da análise das diferenças e de como lidar com as sincronizações que são necessárias.
Em conclusão este sistema é muito direccionado para os requisitos ambicionados por utilizadores que trabalham com imagens, mas podendo ser usado para a gestão de ficheiros em pastas. É um sistema com algumas limitações no que respeita à resolução de conflitos e não permite a passagem de ficheiros entre partilhas o que o torna de alguma forma limitado quando comparado com outras softwares concorrentes.

### 3.2.4 Sync Center

O Sync Center[1] é um sistema de sincronização de ficheiros presente no sistema operativo desenvolvido pela Microsoft, estando integrado nos dois mais recentes projectos desta empresa o Windows Vista e o Windows 7.
Este sistema tem como principal objectivo a sincronização de ficheiros e pastas com dispositivos móveis, pastas presentes numa rede ou ainda com outros programas, sendo este um agregador de informação que gere todos dispositivos que se sincronizam com o dispositivo que albergue este software.
O funcionamento do Sync Center tenta ser o mais user-friendly possível, visto este estar incorporado num sistema operativo utilizado pelas massas, apresentando apenas ao utilizador a escolha da localização ou dispositivo onde se encontram os dados, quais os dados que se deseja sincronizar e os períodos em que se deve efectuar essa mesma sincronização. Este sistema oferece assim uma plataforma onde os utilizadores podem aceder a todos os dispositivos e aos ficheiros presentes nos mesmos, controlar a sincronização dos mesmos e resolver conflitos que possam advir do processo anterior.
Uma das principais inovações presentes neste sistema são os “Offline Files”. Estes permitem que sejam designadas pastas e ficheiros as quais vão estar disponíveis para ser acedidas e alteradas mesmo quando não existe uma ligação entre os dispositivos que as mantêm. Isto permite a alteração dos dados em qualquer altura, sendo que quando existe uma ligação com a versão presente online esta será sincronizada e actualizando ambas as versões para o mesmo estado. Este processo pode levar a existência de conflitos que normalmente são resolvidos com a escolha da versão mais recente dos dados gerados. Caso os dados tenham sido alterados em ambas as

1 http://windows.microsoft.com/en-US/windows-vista/How-to-keep-your-information-in-sync
http://msdn.microsoft.com/en-us/library/aa369140%28v=vs.85%29.aspx

localizações é dada ao utilizador a responsabilidade da escolha da versão que se deseja manter, sendo possível mesmo guardar as duas versões.
Para concluir o Sync Center é uma boa plataforma para unificar todos os processos de sincronização que possam advir dos mais variados dispositivos, indo desde uma simples pen usb ou um leitor de música portátil a uma rede onde se encontram os ficheiros partilhados. Este sistema apresenta algumas lacunas nomeadamente ao nível da informação que transmite ao utilizador sobre as actualizações que são feitas sobre os dados, sendo que guarda apenas uma versão de cada documento, o que conduzir a problemas de perda de informação proveniente de uma resolução de conflitos que possa não ter sido tão satisfatória.

### 3.2.5 DropBox

Dropbox[1] é um sistema de armazenamento online de ficheiros, acessível a partir de qualquer computador, independentemente do sistema operativo que possua. Actualmente este é suportado em sistemas como Windows, Linux, Max OS X, e ainda em dispositivos móveis que possuam Android, Windows Phone e ainda em Iphones, Ipad ou BlackBerrys. Este sistema inicialmente fornece aos utilizadores uma capacidade de 2GB de capacidade de armazenamento que pode chegar aos 8 GB. Para utilizadores que escolham pagar para tirar um maior proveito deste serviço este pode chegar até aos 100GB.
O funcionamento deste sistema passa por ser bastante simples e fácil de perceber por parte do utilizador. Existem duas formas de funcionar com esta tecnologia, sendo que a mais recorrente passa por instalar num computador ou dispositivo móvel a aplicação da dropbox. Esta cria uma pasta no sistema onde está instalada e os utilizadores apenas têm de ir colocando nessa pasta os ficheiros que desejem fazer uma cópia dos mesmos. Estes serão automaticamente guardados nos servidores deste serviço e replicados em todos os dispositivos que tenham a mesma conta associada, sendo possível ter múltiplos sincronizados com a mesma dropbox. No caso de ficheiros serem adicionados, removidos ou alterados nesta pasta, todas estas alterações são propagadas para o Servidores deste serviço e o utilizador é avisado das alterações que estão a proceder, sendo todos os dispositivos automaticamente ligados pela mesma conta actualizados. A outra forma é através de um qualquer browser, onde a partir de qualquer computador com acesso à Internet é possível fazer todas as acções supracitadas, pois basta aceder à pagina da Dropbox fazer o login e todos os dados estão disponíveis ao utilizador, podendo este proceder as alterações que desejar, que a semelhança do que foi dito anteriormente todas serão propagadas pelos seus dispositivos, caso estes estejam conectados à Internet.
Outra possibilidade interessante fornecida por este serviço é a capacidade de partilhar pastas com outros utilizadores, permitindo assim trabalho colaborativo entre um ou mais utilizadores, sem requerer a presença física dos mesmos. Este sistema apresenta

1 https://www.dropbox.com/

ainda um sistema de versões associado a todos os ficheiros presentes na Dropbox. Este sistema permite o retrocesso a versões antigas de ficheiros em caso de necessidade, sendo que o histórico de cada ficheiros apenas tem uma duração de 30 dias para as versões livres, enquanto nas contas com versões pagas a duração pode ser ilimitada. Outra limitação das contas livres encontra-se no tamanho máximo que cada ficheiro pode ter, sendo que cada um não pode ultrapassar os 300 MB, o que novamente não acontece nas versões pagas do mesmo serviço.
As maiores desvantagens que podem ser apontadas a este serviço passam por ser:

- Em caso de alteração simultânea do mesmo ficheiro a Dropbox não possui a capacidade para resolução de conflitos, dando apenas um aviso ao utilizador da presença do mesmo e em que ficheiro, sendo criadas duas versões do mesmo ficheiro com as respectivas diferenças e deixando à responsabilidade do utilizador a resolução do problema;
- Existência de algumas questões relativas à privacidade dos dados que os utilizadores guardam nos servidores da Dropbox, pois as técnicas de conservação e compressão dos dados utilizadas por esta empresa têm sido postas em causa, por poderem descodificar a informação que os utilizadores guardam invadindo assim a sua privacidade.

### 3.2.6 Live Mesh

Live Mesh[1] é um sistema desenvolvido pela Microsoft, sendo em múltiplos aspectos similar à DropBox, sendo que o seu objectivo principal acaba por ser guardar ficheiros online e garantir a sua sincronização. À semelhança da DropBox também é disponibilizado espaço para livre utilização sendo o valor inicial deste de 5GB.
Actualmente as semelhanças entre este serviço são bastantes, sendo que algumas das diferenças se baseiam no facto de este serviço apesar ser utilizável em várias plataformas como o Windows ou o Mac OS X, deixa de fora o mundo open source sendo que os utilizadores de Linux ou sistemas como o Android não poderam beneficiar da utilização de um sistema desta índole, o que pode estar implícito em algum desconhecimento da existência desta tecnologia. Nestes casos é oferecida apenas a possibilidade de acesso aos ficheiros via web.
Esta tecnologia acaba por ser base de algumas das funcionalidades apresentadas nos dias de hoje na Dropbox, especialmente a utilização do browser como ferramenta de aquisição, alteração ou remoção de ficheiros.
Em suma este tecnologia é em múltiplos aspectos similar à Dropox, acabando por se diferenciar na capacidade de armazenamento disponibilizada no serviço livre de pagamentos e na inexistência do seu suporte em tecnologias open-source.

1 http://explore.live.com/windows-live-mesh-devices-sync-upgrade-ui

http://en.wikipedia.org/wiki/Windows_Live_Mesh

### 3.2.7 Syncany

Syncany[1] é um novo software de sincronização de ficheiros. Este em comparação com o Live Mesh da Microsoft ou a Dropbox apresenta algumas vantagens relativamente aos seus concorrentes, nomeadamente o facto de ser um produto open source, logo possibilitando a facilidade de obtenção do código do mesmo e a sua reutilização. Outra vantagem passa pela capacidade de encriptar os ficheiros presentes, na máquina em que estão a ser partilhados, garantindo desta forma a segurança dos mesmos e a privacidade que normalmente os utilizadores desejam num serviço desta natureza. Este serviço tem o objectivo de ser extensível através de introdução de plugins, para que possam ser introduzidos novos protocolos para sincronização de dados. Actualmente os protocolos suportados são FTP, Box.net, Amazon S3, Google Storage, Imap, etc sendo objectivo dos criadores desta tecnologia introduzir no futuro suporte a outros protocolos como o Windows Share.
Apesar de este sistema apresentar algumas ideias revolucionárias, acaba por ser muito imberbe, especialmente por apenas funcionar em sistemas Linux e se encontrar em pleno estado de desenvolvimento. Infelizmente um dos objectivos presentes passa por nunca ser suportado por plataformas como o Windows o Mac, o que lhe pode vir a retirar alguma visibilidade e reconhecimento no mundo da informática.

## 3.3 Software de controlo de versões

Este tipo de software não se enquadra no tipo de utilizadores que normalmente utilizam sincronizadores de ficheiros, visto que um software de controlo de versões é utilizado com mais regularidade para o desenvolvimento de projectos onde existe a partilha de ficheiros entre vários responsáveis pela evolução do mesmo. Assim sendo, o objectivo do estudo deste tópico passa pela análise de uma tecnologia chamada Git, que tem como principais características velocidade, eficiência e escalabilidade. Estas são alguns dos objectivos que se pretendem melhorar no REPOX com a execução desta tese.

### 3.3.1 GIT

Git[2] é um sistema open source de controlo de versões distribuído. Este, ao contrário do que acontece noutros tipos de sistemas de controlo de versões, não utiliza um repositório central como acontece por exemplo no CVS[3], onde são guardados todos os ficheiros e a sua evolução ao longo do desenvolvimento do projecto e os utilizadores acedem para efectuarem operações sobre o mesmo. Com o sistema Git cada projecto é

1 http://www.webupd8.org/2011/05/syncany-great-dropbox-alternative-which.html

2 http://git-scm.com/

3 http://cvs.nongnu.org/

um repositório com todas as capacidades de controlo de versões e de histórico asseguradas. Este sistema permite ainda uma fácil ramificação dos projectos, podendo estes ser locais ou remotos podendo ser mesmo inseridos noutros projectos, sendo cada ramo uma cópia exacta do repositório de onde deriva. Este processo permite que não hajam perdas de informação pois deixa de existir um repositório para passar a haver N.
Uma grande vantagem inerente ao facto de cada utilizador ter um repositório próprio onde efectua o desenvolvimento é a inexistência de necessidade de conexão à Internet para que este registe alterações no projecto ou permita a procura de alterações antigas ou ainda permita a fusão de código proveniente de ramos locais ao sistema. A conexão ao exterior é apenas necessária para partilhar ou obter dados de ramos remotos de um projecto.
O sistema Git é transversal aos protocolos de Internet principais, sendo possível que os repositórios sejam partilhados tanto por HTTP, FTP, rsync ou ainda por ssh, o que facilita a sua utilização e aplicação nos mais diversos contextos. Visto a sua implementação ser maioritariamente em C permitiu que esta plataforma seja também transversal ao sistema operativo em que se trabalhe, sendo mesmo utilizado como ferramenta complementar a sistemas de IDE como o Eclipse, IntelliJ ou NetBeans.
Como sistema de armazenamento, o Git usa um sistema de snapshots do estado de toda a árvore de ficheiros. Isto passo é efectuado a cada nova alteração efectuada no projecto. Numa fase inicial do projecto, este utilizava um sistema de deltas que consistia em guardar as diferenças existentes entre os ficheiros existentes aquando da submissão de alterações. Este processo revelou-se muito dispendioso ao nível do espaço necessário para o armazenamento dos mesmos, sendo necessário passar para um sistema de snapshots do estado de todos os ficheiros, obtendo-se assim melhorias ao nível tanto do armazenamento como da eficiência da obtenção das diferenças submetidas e ainda na pesquisa e partilha das mesmas com outros ramos do projecto.
Tendo já mencionado como característica deste sistema a facilidade da partilha de dados entre os diversos ramos do projecto, este processo torna-se mais cómodo devido ao Git possuir um sistema de detecção de conflitos, providenciando ferramentas que permitem aos utilizadores a visualização das diferenças e ajuda para a sua resolução. É ainda possível alterar as ferramentas usadas para este efeito, visto este sistema permitir a utilização de outras que sejam mais familiares aos utilizadores.
Assim e para concluir, podemos verificar que este sistema apresenta características que permitem facilitar principalmente equipas de desenvolvimento de projectos, pois reduz a complexidade na sincronização da informação gerada por diferentes intervenientes.

### 3.4 Data-sharing middleware

Como o próprio termo indica, middleware representa uma camada de abstracção entre dois sistemas utilizados. Neste caso representam sistemas que retiram a

responsabilidade da tecnologia de obtenção dados a forma como estes são replicados para o sistema. A finalidade principal deste tipo de software é a de serem facilitadores de desenvolvimento de aplicações pois permitem a sua utilização sem grandes mudanças ao código das mesmas.

### 3.4.1 IceCube

O objectivo do projecto IceCube [9] passa por ser o fornecimento de uma plataforma que funcione como um conciliador de cópias do mesmo trabalho que sofreram alterações, para que a sua integração seja o mais pacífica possível. Este sistema está parametrizado para ter em atenção a semântica do tipo de dados que se está a trabalhar a aplicação dos mesmos ou do utilizador que gerou as alterações.
Este sistema tem como unidades base para o seu funcionamento o estado inicial de um ficheiro que se está a analisar e os registos (logs) gerados pelas acções aplicadas em cada réplica deste. Os registos (logs) fornecem ao sistema o histórico das acções tomadas por cada utilizador, permitindo assim aumentar as capacidades do programa, pois este fica a perceber as intenções de cada utilizador aquando da aplicação das alterações, facilitando a sua integração com as alterações de outros utilizadores. Ao contrário de outros sistemas o IceCube pretende reordenar as operações latentes nos registos (logs), para além da simples ordenação temporal, para assim procurar formas de minimizar os conflitos. Esta solução levanta uma questão que se prende com uma possível explosão das combinações possíveis entre os registos (logs), tendo sido para isso aplicadas ao sistema sistemas de restrições (estáticas e dinâmicas) para controlar esta lacuna de desenho do sistema.
Restrições estáticas estão directamente relacionadas com a ordem de como são aplicadas as operações para chegar a um estado final. Apenas têm em atenção se estas são aplicadas de forma segura não se preocupando com estado actual dos objectos que estão a ser tratados.
Restrições dinâmicas pode ser ou operações ou pré-condições. Uma operação é um método que pode ou não modificar os objectos que estão a ser partilhados, indicando o sucesso ou insucesso da mesma. Uma pré-condição apenas verifica se estado actual de um objecto é válido.
O processo de funcionamento do IceCube passa por duas fases. Estas são:

- Execução isolada: nesta fase são aplicados os objectos partilhados um conjunto de actualizações produzidas pelo utilizador, sendo gerado um registo deste procedimento;
- Fase de reconciliação: nesta fase é feita a tentativa de sincronização de duas ou mais réplicas do mesmo objecto partilhado sendo subdividida em três novas fases;
    - Fase do escalonamento: nesta fase são criados escalonamentos a partir das várias combinações possíveis das actualizações. Estes escalonamentos são conjuntos de acções que irão ser aplicadas nos

objectos partilhados respeitando as restrições estáticas para gerar um estado considerado correcto. Se o estado não for considerado correcto esse escalonamento é descartado, controlando assim a explosão que possa haver de combinações;

- Fase da simulação: nesta fase são aplicados os escalonamentos de actualizações considerados válidos na fase anterior, verificando se as restrições dinâmicas são respeitadas, descartando novamente os conjuntos de actualizações que não respeitem as considerações definidas;
- Fase da selecção: nesta última fase todas os escalonamentos que foram considerados válidos sendo estes comparados e classificados, sendo escolhido a solução que apresente o melhor resultado final. O resultado originário é posteriormente partilhado entre todos as réplicas do objecto para aplicação da solução obtida.

Em conclusão o IceCube é sistema de reconciliação de objectos partilhados, baseado nos registos gerados das actualizações produzidas pelos utilizadores. Esta solução tem como principal objectivo a diminuição de conflitos, sendo totalmente orientada para soluções de trabalho partilhado.

### 3.4.2 Semantic Chunks

Semantic chunks [10] são um conceito desenvolvido com o intuito de tentar resolver alguns problemas apresentados pelas técnicas (update-based e operational-based) utilizadas na base do trabalho cooperativo, nomeadamente na ajuda ao nível de fornecer garantias de consistência e de diminuição de conflitos aquando da sincronização de ficheiros.
A ideia por de trás deste conceito nasce de tentar adoptar as vantagens de cada uma das técnicas anteriormente mencionas e da utilização de chunks em LBFS e Haddock-FS, tendo dado origem a poupanças tanto de armazenamento de dados como de largura de banda utilizada na propagação das actualizações efectuadas nos projectos. O funcionamento deste sistema baseia-se na divisão de documentos em regiões semanticamente relevantes, que podem ser diferentes consoante do tipo ficheiro e a sua aplicação semântica. Estes pedaços de informação extraídos podem ir de um simples parágrafo num texto, a uma célula de uma folha da cálculo ou até mesmo a uma página de uma apresentação. Este facto promove a consistência de ficheiros num sistema, visto que as divisões dos mesmos ser feita semanticamente, tendo em conta o tipo de ficheiro, e não por blocos de tamanho constante como em outras soluções, reduzindo assim o número de conflitos provenientes da sincronização de ficheiros e aumentando a concorrência e a frequência de actualizações provenientes de múltiplas fontes.

Visto este ser um sistema direccionado para o trabalho cooperativo e apesar da redução de conflitos conseguida, estes continuam a existir e assim são utilizadas várias esquemas para que os utilizadores os consigam resolver. Estes são :

- Votações para sabes quais as actualizações que devem utilizadas;
- Actualizações adoptadas por decisão de utilizadores com maiores privilégios no desenvolvimento do projecto;
- Definição de períodos para a introdução de actualizações por parte de algum utilizador (lease);
- Partilha de informação com outros utilizadores sobre a alguma actualização que se pretende inserir no projecto.

Como conclusão pode-se observar que este sistema consegue obter as maiores virtudes das técnicas update-based e operational-based, reduzindo o número de conflitos produzidos por actualizações da informação e aumento da concorrência e frequência das actualizações. Existe ainda uma redução de largura de banda e da capacidade necessária para armazenamento devido a se lidar com fragmentos de informação facilitando o controlo das actualizações de ficheiros.

### 3.4.3 Xmiddle

O Xmiddle [11] é um sistema que tem como objectivo principal a partilha de informação entre dispositivos de computação móveis, como é o caso telemóveis, PDAs ou computadores portáteis, sem a existência de qualquer tipo de rede de comunicações fixa. Este sistema aborda principalmente cenários em que é são utilizadas redes ad-hoc, mais especificamente comunicação entre apenas dois intervenientes.

Em termos de estruturação dos dados, este sistema guarda os mesmos em árvores de estruturas organizadas hierarquicamente, facilitando desta forma o acesso e manipulação dos mesmos. Toda a informação é representada internamente em XML para uma maior flexibilidade e facilidade de associação com o sistema supracitado.

Quanto ao funcionamento do Xmiddle, este pretende a sincronização de ramos de árvores que sejam comuns a dois dispositivos, criando assim duas novas árvores semelhantes e com a mesma versão nos dois dispositivos. Este começa por, quando há ligação entre dois dispositivos, verificar a existência de ramos partilhados entre estes. Confirmado este requisito o dispositivo que promove a conexão, que iremos chamar de D2, envia ao outro (D1) o histórico das alterações promovidas no ramo. Este verifica as diferenças entre os dois históricos e envia-as para D2 onde este irá gerar uma nova árvore baseada nas diferenças, promovendo assim uma fusão das duas árvores. Tendo terminado este processo D2 envia para D1o conjunto de modificações efectuadas para que o segundo possa gerar um nova árvore semelhante à de D2 e assim terminar o processo de sincronização entre ambos. Caso este tenha terminado com sucesso ambas as árvores irão apresentar a mesma versão, caso tenham havido conflitos aquando da fusão dos ramos o Xmiddle permite a definição de políticas de

reconciliação de dados através de esquemas XML, que resolvam conflitos automaticamente quando estes existirem.
No caso de haver algum tipo de quebra da ligação, que impeça a continuação da comunicação das mudanças aplicadas nos dados, os dispositivos retêm as réplicas da última árvore estável que estava a ser partilhada, permitindo assim a evolução do desenvolvimento do trabalho apesar do acontecimento, sendo o processo de sincronização reinicializado aquando da existência de uma nova ligação entre os dispositivos em questão.
Para concluir pode-se afirmar que o desenvolvimento desta tecnologia tem como objectivo promover a descoberta de estratégias que lidem com os problemas associados à computação móvel como, perdas de conexão, baixa capacidade de largura de banda ou problemas energéticos.

### 3.4.4 Microsoft Sync Framework

O Microsoft Sync Framework[1] é uma plataforma desenvolvida pela Microsoft com o intuito de permitir uma fácil sincronização de qualquer tipo de documentos independentemente da aplicação, do tipo de dados ou de qualquer protocolo utilizado no desenvolvimento do projecto.
Para que seja possível a partilha de qualquer tipo de informação esta tem de estar armazenada em algum lugar. Nesta óptica a Microsoft decidiu definir actores a quem deu o nome de Participantes. Estes são os locais de onde se consegue obter a informação proveniente das fontes de dados, podendo estes serem um computador, uma pen drive, um PDA ou até mesmo um web service. Estes participantes podem ser de vários tipos, que variam conforme as suas capacidades de armazenamento e manipulação da informação tanto de forma local como remota e ainda da sua capacidade de executar aplicações para sincronização de informação directamente no dispositivo. Assim os participantes existentes são:

- Participantes Totais: são definidos por permitirem a criação de aplicações directamente nestes dispositivos bem como a definição da localização dos dados a serem guardados;
- Participantes Parciais: são definidos pela sua capacidade de armazenamento de dados como exemplo disso pen drives ou SD cards;
- Participantes Simples: são definidos por apenas partilharem informação quando esta lhe é pedida, sendo exemplos disto os RSS feeds.

Para que os participantes possam partilhar dados entre si é necessária a existência de outra entidade. Esta é a chave de todo os sistema utilizado pelo Microsoft Sync Framework e dá pelo nome de Provider (fornecedor). Estes podem ser definidos pelos utilizadores, visto que os dados que serão partilhados podem ser de algum tipo não suportado por esta estrutura. No entanto são disponibilizados vários providers, sendo exemplo de alguns deles, providers para sincronização de bases de dados, de ficheiros

[1] http://msdn.microsoft.com/pt-pt/sync/default(en-us).aspx

e pastas, etc. Para cada fornecedor é especificado o tipo de dados que este irá sincronizar, sendo este responsável pela sua manutenção e garantia de consistência dos mesmos. O funcionamento destes vai para além das funções supracitadas, sendo que estes guardam informação relativa as alterações aplicadas aos dados, nomeadamente as mudanças ocorridas e o estado em que os dados se encontram. Esta informação é guardada num repositório de metadados que pode ser definido ou pelo criador do provider ou então utilizar o fornecido pelo próprio Microsoft Sync Framework.
O processo de sincronização deste sistema está directamente ligado a três módulos muito específicos. O Sync Provider, que fornece toda a comunicação entre todas as réplicas do projecto e ainda outros providers com se deseje comunicar com este e ainda o Data Source e o Metadata Store que armazenam respectivamente os dados e os metadados relativos aos dados.
Finalmente este sistema apresenta ainda um sistema de detecção e resolução de conflitos com mecanismos pré-definidos para a automática resolução dos mesmos. À semelhança do que acontecia com os providers, também é possível criar regras para resolução deste tipo de conflitos.
Em conclusão o Microsoft Sync Framework apresenta uma grande flexibilidade para a sua utilização, fornecendo assim uma plataforma avançada de utilização, sendo esta direccionada para utilizadores com conhecimentos mais avançados conseguindo assim partilhar e gerir a evolução dos seus projectos.

### 3.5 Análise

Tendo em conta as soluções que foram apresentadas nesta secção, é possível verificar que qualquer uma delas faz a replicação de ficheiros e pastas entre diferentes dispositivos garantindo a sua consistência. Assim neste sub-capítulo iremos avaliar, e com base na tabela 1, como estas soluções podem ser utilizadas para resolver as questões definidas no capítulo 2.
Analisando a tabela anterior podemos verificar que muitas das soluções apenas funcionam em determinados sistemas operativos, sejam eles proprietários ou livres, o que pode vir a ser um problema pois, é impossível, prever que sistemas operativos tanto os Data Providers como os Service Provider, o que limita automaticamente as escolhas para possíveis soluções dos problemas estudados. Outro factor de exclusão para o tipo de tecnologias que podem ser utilizadas como solução para os problemas desta tese é o tipo de licença que existe para reutilização das tecnologias. Visto o REPOX não utilizar qualquer tipo de software proprietário esta terá de ser uma máxima que terá de ser mantida. Assim para concluir as tecnologias que são passíveis de estudo para esta tese, com base nos factores de exclusão supracitados são o SyncML, o Rsync, o Unison, os Semantic-Chunks, o IceCube e o Xmiddle.

**Tabela 1.** Comparação de tecnologias de recolha e sincronização

| | **Plataforma** | **Tipo de Sincronização** | **Serviço de sinconização** | **Agendamento** |
|---|---|---|---|---|
| **SyncML** | Independente | Online | Dados entre diversos dispositivos | Sim |
| **Rsync** | Windows, Mac OS X, Linux | Online | Ficheiros em redes de baixa capacidade | Usa OS |
| **Unison** | Windows, Mac OS X, Linux | Online | Algoritmo de Rsync | Usa OS |
| **Synctoy** | Windows | Offline | Pastas locais | Usa OS |
| **Sync Center** | Windows | Offline | Pastas locais | Não |
| **DropBox** | Windows, Mac OS X, Linux. Android, Windows Mobile, Iphone | Online | Pastas e ficheiros em múltiplos dispositivos | Não |
| **Live Mesh** | Windows | Online | Pastas e ficheiros em múltiplos dispositivos | Não |
| **Syncany** | Linux | Online | Pastas e ficheiros em múltiplos dispositivos | Não |
| **Semantic-Chunks** | Independente | Online | Ficheiros e redução de conflitos em redes de baixa capacidade | Sim |
| **IceCube** | Independente | Online | Reconciliador de actualizações de objectos partilhados | Sim |
| **Xmiddle** | Independente | Offline | Informação entre dispositivos móveis | Sim |
| **Microsoft Sync Center** | Microsoft | Online | Qualquer tipo de documentos independente da sua génese | Sim |

# 4 Solução a Explorar

Tendo em conta os desafios apresentados ao longo deste projecto de dissertação e as tecnologias analisadas ao longo do estado da arte presente neste documento, o primeiro passo para a procura de soluções passa por, uma análise clara aos conteúdos que são produzidos pelas bibliotecas e arquivos digitais, que são posteriormente disponibilizados para recolha. Estes dados podem ser de vários tipos (texto (fulltext), imagens, áudio e vídeo), sendo que no âmbito desta tese pretendo apenas tratar do caso específico do fulltext. Como se pode observar na figura 1 que representa um possível processo de recolha de fulltext este pode vir armazenado de diversas formas.

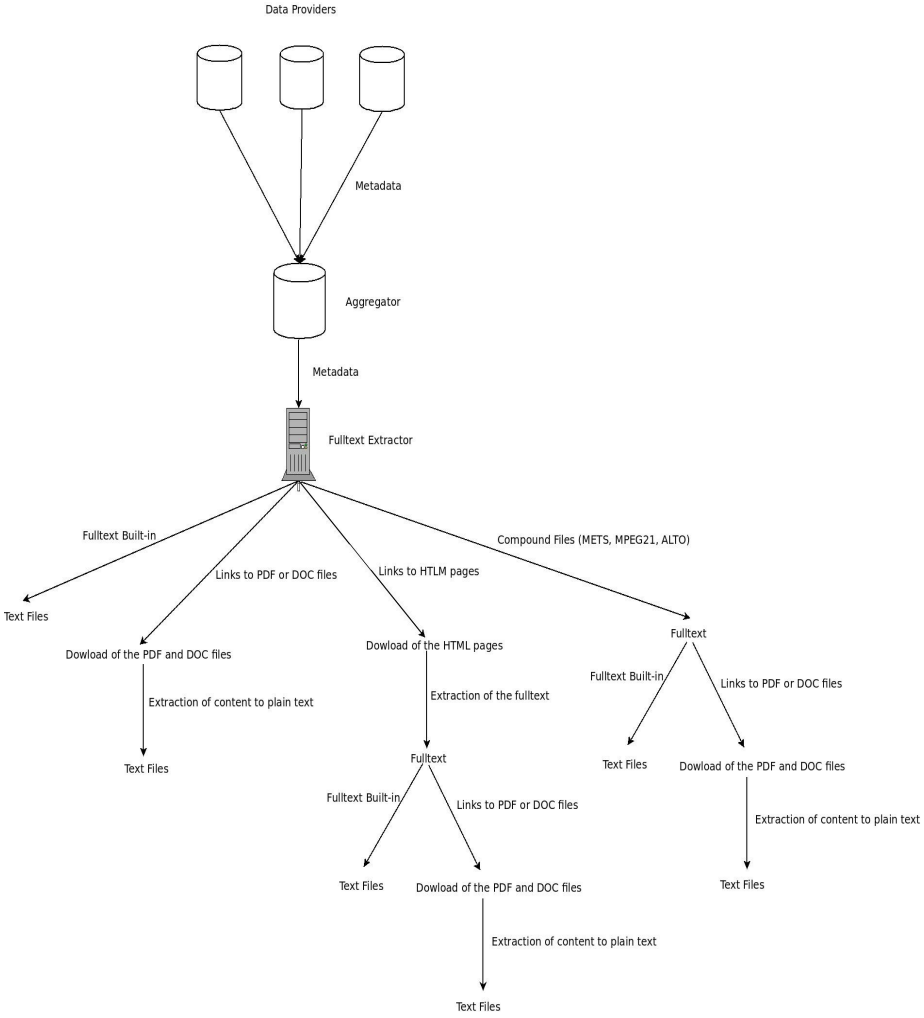


**Figura 1.** Processo de recolha de fulltext

Este facto é de extrema relevância, pois para diferentes tipos de dados os seus processos de recolha e a sua sincronização podem ser afectados.
Tendo concluído este processo pretende-se analisar as tecnologias que melhores garantias tragam ao nível do uso da largura de banda, da quantidade de processamento de CPU do computador, para os dados analisados e assim desta forma conseguir optimizar o seu funcionamento de forma a tentar superar as tecnologias existentes. Esta optimização tenderá a passar por uma tentativa de cumprimento dos seguintes requisitos:

- diminuição da largura de banda necessária para a recolha e /ou sincronização dos conteúdos;
- diminuição da capacidade de processamento necessária para o tratamento dos conteúdos recebidos;
- escalabilidade da tecnologia de forma a permitir o contínuo crescimento de informação agregada, do número de Data Providers e Service Providers agregados ao projecto.

Esta solução pretende ser estudada e implementada no âmbito do projecto REPOX, pois esta é uma tecnologia amplamente utilizada e direccionada para lidar com os problemas que envolvem as bibliotecas e arquivos digitais.
Assim "o REPOX é uma implementação do Conceito de Repositório de Metadados"[12], que fornece uma plataforma de funcionamento aberta, sem o recurso a qualquer tipo de tecnologia proprietária no seu processo de preservação de dados.
Este sistema faz a gestão de várias colecções de metadados, provenientes de entidades diversas, sendo que cada uma delas está ligada a uma interface de fonte de dados, que serão responsáveis pela obtenção dos registos gerados e verificação dos mesmos antes sua integração no sistema REPOX. Estas verificações passam essencialmente por codificar os registos obtidos no esquema XML que tenha sido adoptado pela entidade que o gera, sendo que a obtenção dos mesmos pode ou não ter uma periodicidade obrigatória definida.
Arquitecturalmente toda a estrutura do REPOX[13] pode ser observada na imagem em baixo, sendo que o principal destaque de interesse desta infra-estrutura é o gestor de repositório. Este é implementado em Java EE[1] e tem como principal funcionalidade a recolha e gestão de dados provenientes das diferentes interfaces.
Como pode ser observado na figura 2, o REPOX possui várias interfaces para a recolha de dados sendo elas o HTTP, FTP, o OAI-PMH e o Z39-50. De todas estas interfaces a mais utilizada é o OAI-PMH, visto este protocolo ser um requisito comum requerido para a partilha de dados por parte das organizações que assim o desejem.
Considerando, no entanto a alteração de paradigma, o que se pretende implementar no REPOX, consiste na introdução de um sistema de recolha de conteúdos para além dos metadados, o que irá introduzir algumas alterações na arquitectura desta framework.

1 http://www.oracle.com/technetwork/java/javaee/overview/index.html

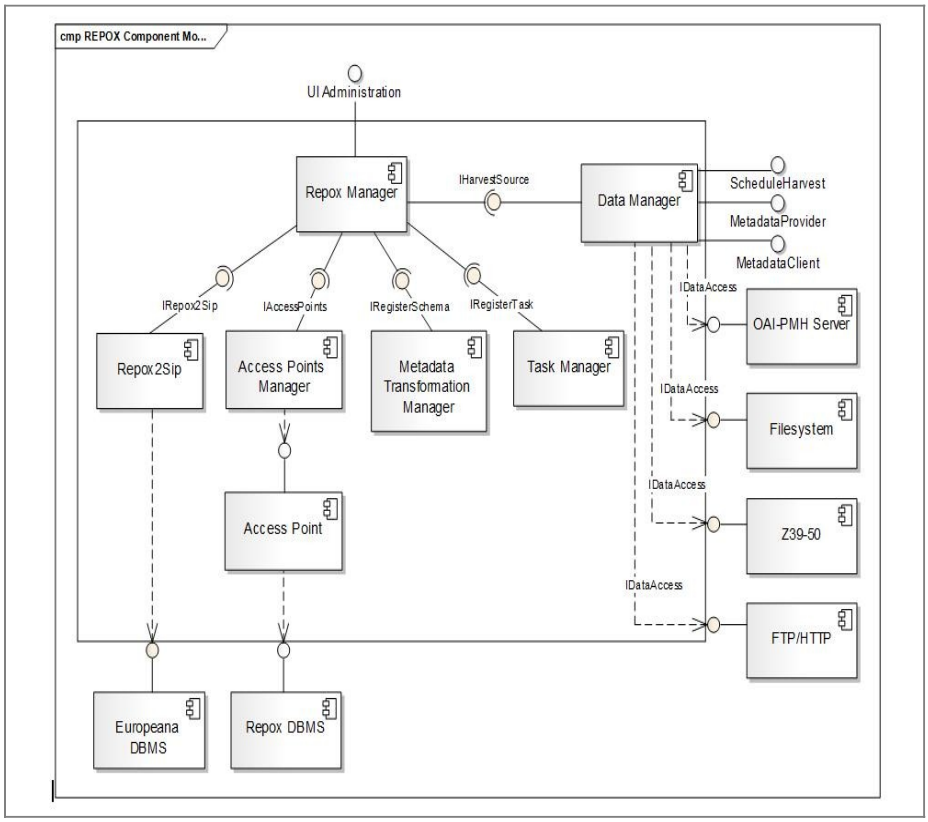


**Figura 2.** Arquitectura de componentes da framework REPOX

Estas alterações consistem basicamente na introdução de um Fulltext Manager que vai complementar o funcionamento do Data Manager, permitindo a este a possibilidade de poder interpretar os metadados recebidos e extrair destes os conteúdos que estes referenciem. Estes serão guardados para um futura publicação dos mesmos, tanto para efeitos de pesquisa por um Resource Explorer, como por outra entidade autorizada que deseje fazer uma recolha dos mesmos, como pode ser observado na figura 3.

Assim, e como forma de resumo, durante a execução desta tese pretendo atacar os seguintes requisitos funcionais:

- Criação de um serviço de recolha de conteúdos, principalmente focado no fulltext, com base nas referências extraídas dos metadados;
- Introdução de um sistema para sincronização dos conteúdos;
- Gestão do armazenamento dos conteúdos recolhidos.
- Criação de um serviço de publicação e partilha de conteúdos.

Estes requisitos estão na origem de algumas das interfaces e componentes que se podem observar na figura 3, nomeadamente a introdução de uma nova interface para a recolha dos conteúdos, que corresponde ao Optimal Fulltext client, estendendo assim as funcionalidades do Harvest Manager, a criação do novo componente FullText Manager para gestão e sincronização. No FullText repository dos conteúdos recolhidos e finalmente uma interface para permitir a publicação e partilha dos dados supracitados.

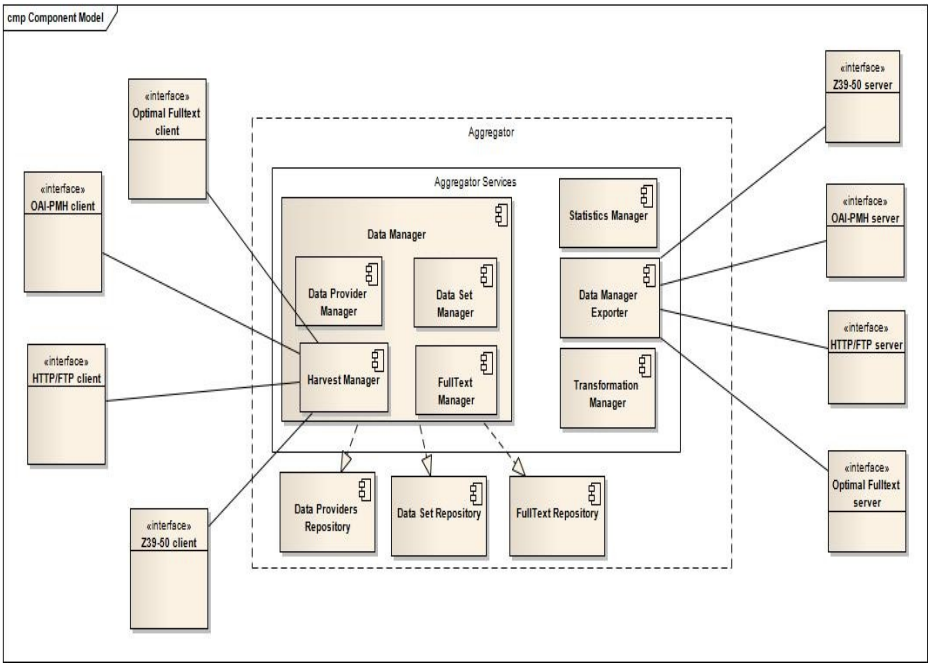


**Figura 3.** Introdução sistema de recolha de fulltex na arquitectura do REPOX

# 5 Proposta de Validação de Resultados

Para validar os resultados provenientes da proposta de solução do capítulo anterior proponho a utilização dos registos recolhidos nos âmbitos dos projectos Europeana e EuDML. Estes projectos apresentam uma enorme quantidade de registos sendo que no caso do projecto da Europena o total de páginas ascende aos 3 milhões de páginas.
Com esta quantidade de registos é possível assim criar um perfil dos dados que são recolhidos dos projectos e desta forma poder avaliar convenientemente qual deve ser a técnica estudada a utilizar por forma a cumprir os objectivos definidos ao longo deste projecto.

De forma a avaliar convenientemente a tecnologia que deve ser utilizada para resolver as questões enunciadas ao longo deste projecto pretendem-se fazer os seguintes testes:

- Duração de uma recolha total dos conteúdos provenientes de várias entidades;
- Quantidade de dados armazenados durante o processo de recolha dos conteúdos;
- Quantidade de dados transferidos num processo de actualização dos conteúdos anteriormente recolhidos;
- Duração do processo de actualização supracitado;
- Recursos necessários para o cumprimento da tarefas anteriores.

Como já foi referido no capítulo anterior irá ser utilizada a framework REPOX para desenvolvimento da solução para os problemas analisados na secção dois deste projecto. Esta tecnologia acaba por ser também por si um validador dos estudos que irão ser feitos, pois este permite a realização de casos de teste reais, com projectos como a Europeana ou o EuDML.
Resumindo pretende-se aferir de forma inequívoca se a solução que se pretende introduzir ,os custos em termos da largura de banda necessária, de capacidade de processamento necessária para realizar as tarefas e se com o escalamento dos dados tratados se estas directrizes são eficientes.

# 6 Planeamento de trabalho

Em termos de planeamento de trabalho prevejo que o desenvolvimento e escrita da tese tenha os seguintes períodos de trabalho descritos:

1. Análise dos conteúdos recolhidos (fulltext) e criação de um perfil dos mesmos – 3 semanas;
2. Análise das tecnologias estudadas para resolução das questões abordadas ao longo da escrita deste documento, com base no perfil feito anteriormente – 2 semanas;
3. Integração e optimização das tecnlogias escolhidas – 2 meses;
4. Escrita da dissertação – 1 mês;

É possível ver de forma mais detalhada os tarefas e sub tarefas que se pretendem desenvolver no Apêndice A.

# 7 Conclusão

O aparecimento das bibliotecas e arquivos digitais criou a possibilidade de se partilhar com a comunidade a informação que cada uma destas possui, para que esta possa ser

pesquisada e analisada para os mais variados fins e desejos. Com este objectivo em vista foi criado o REPOX que funciona como aglomerador de metadados descritivos das informações que as bibliotecas e os arquivos desejem disponibilizar. Estas são publicadas de forma a facilitar a sua pesquisa por parte dos utilizadores, permitindo a estes encontrar as informações que desejem para depois se dirigirem as entidades responsáveis pela partilha dos dados para obtenção de informações mais detalhadas do que se deseje.

Com o objectivo de melhorar as pesquisas por parte dos utilizadores das bibliotecas e arquivos digitais foi proposto um novo desafio. Este passa pela obtenção e partilha de conteúdos, principalmente fulltext, ou seja, para além dos metadados descritivos obtidos na primeira fase deste projecto, nesta fase serão obtidos conjuntamente todos os conteúdos produzidos para que possam ser pesquisáveis e assim permitir uma maior e melhor qualidade de pesquisa.

Com este novo paradigma em vista deixa de ser possível manter apenas os protocolos de obtenção de dados que se encontram instituídos nas partilhas de dados entre bibliotecas digitais, sendo caso disso o OAI-PMH, que se revelou um sucesso em termos da obtenção dos metadados em diferentes localizações, mas que devido as suas características acaba por não ser suficiente para cumprir os novos objectivos apresentados.

Assim o objectivo desta dissertação passa pela pela procura da melhor solução que permita resolver o problema introduzido neste documento. Esta terá de saber lidar vários tipos de dados e promover a recolha dos mesmos, sendo que se pretende reduzir custos de banda larga, processamento e tempo despendido aquando da actualização dos mesmos e ainda que seja escalável. Outro objectivo que se pretende alcançar durante a execução desta tese passa pela tentativa de optimização do protocolo OAI-PMH, onde se procurará melhorar o processo de sincronização dos metadados em Bibliotecas Digitais em Rede, de forma a tornar este procedimento mais eficiente.

# 8 Referências

[1]Carl Lagoze and Herbert Van de Sompel. The open archives initiative: building a low-barrier interoperability framework. In JCDL ’01: Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries, pages 54–62, New York, NY, USA, 2001. ACM.

[2]Van de Sompel, Lagoze. D-Lib Magazine February. 2000, Vol. 6 Number 2.Consultado em: http://www.dlib.org/dlib/february00/vandesompel-oai/02vandesompel-oai.html

[3]Ian H. Witten, David Bainbridge, David M. Nichols How to Build a Digital Library, pages 7-8, Second Edition.

[4]SyncML White Paper:Building an Industry-Wide Mobile Data Synchronization Protocol

[5]Andrew Tridgell and Paul Mackerras. The rsync algorithm. The Australian National University

[6]David Rasch and Randal Burns. In-Place Rsync. File Synchronization for Mobile and Wireless Devices. Department of Computer Science Johns Hopkins University

[7]John Langford. Multiround Rsync. 2001

[8]Benjamin C. Pierce and Jerome Vouillon. What's in Unison? A Formal Specification and ReferenceImplementation of a File Synchronizer. Department of Computer & Information Science. 2004

[9]A.-M. Kermarrec, A. Rowstron, M. Shapiro, and P. Druschel. The icecube approach to the recon-ciliation of divergent replicas.

[10]L. Veiga and P. Ferreira. Semantic-Chunks a middleware for ubiquitous cooperative work. In Proceedings of the 4th workshop on Reflective and adaptive middleware systems, page 6. ACM, 2005.

[11]S. Zachariadis, L. Capra, C. Mascolo, and W. Emmerich. XMIDDLE: A Data-Sharing Middleware for Mobile Computing. 2002

[12]Freire, Manguinhas, Borbinha, REPOX: Uma infra-estrutura XML para a PORBASE, Lisboa

[13]Freire, Manguinhas, Borbinha. Metadata Spaces: the concept and a case with REPOX, International Conference on Asian Digital Libraries. 2006

# Apêndice A – Planeamento tarefas a executar durante a tese

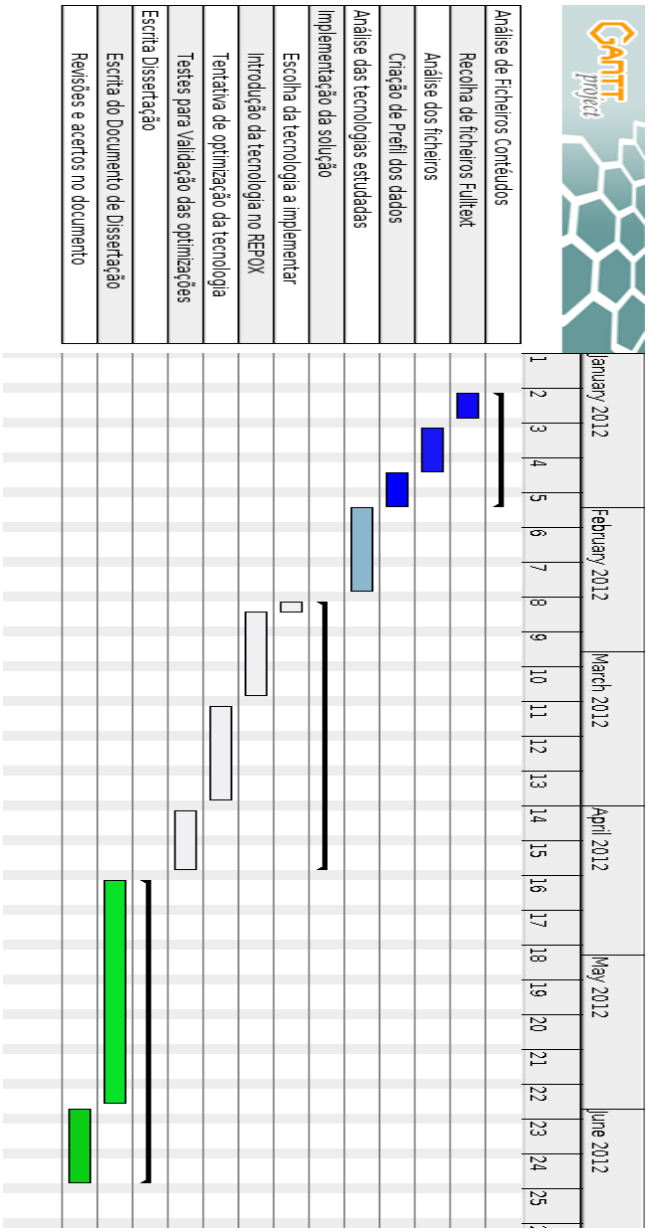


**Figura 4.** Planeamento de Trabalho para dissertação