

RODA

Repositório de Objectos Digitais Autênticos

Documentação técnica do sistema

Projecto 613/2006 POAP

Identificador	41012-???
Versão	Draft
Autor	Luís Faria e Rui Castro
Data publicação	2009-01-01
Acesso	Público
Datas extremas	2006-04-01/2008-12-31
Início de projecto	2006-04-01
Equipa	Francisco Barbedo, José Carlos Ramalho, Luís Corujo, Luís Faria, Miguel Ferreira, Rui Castro

© Direcção Geral de Arquivos - Universidade do Minho
2007



Conteúdo

1	Introdução	2
2	Pré-requisitos	3
2.1	RODA-Core	3
2.2	RODA-Migrator	4
2.3	RODA-Migrator-Win	5
2.4	RODA-in-installer	5
2.5	RODA-WUI	5
3	Instalação	6
4	Configuração	7
4.1	RODA-Core	8
4.1.1	RODA-Core Plugins	10
4.2	Munin	11
4.3	Segurança do JBoss	13
4.4	RODA WUI	15
4.5	RODA In	19
4.6	RODA In Installer	21
4.7	RODA-Migrator	22
4.8	RODA-Migrator-win	22
5	Desenvolvimento	23
5.1	Código fonte	25
5.1.1	Obter	25
5.1.2	Estrutura	25
5.1.3	Eclipse	27
5.2	Compilação e implementação	31
5.3	Componentes do RODA e projectos associados	32
5.3.1	RODA Common	32
5.3.2	RODA Core	34
5.3.3	RODA Core Plugins	36
5.3.4	RODA Client	37
5.3.5	RODA In	38
5.3.6	RODA In Installer	40
5.3.7	RODA WUI	41
5.3.8	RODA Migrator	45
5.3.9	RODA Handle	46

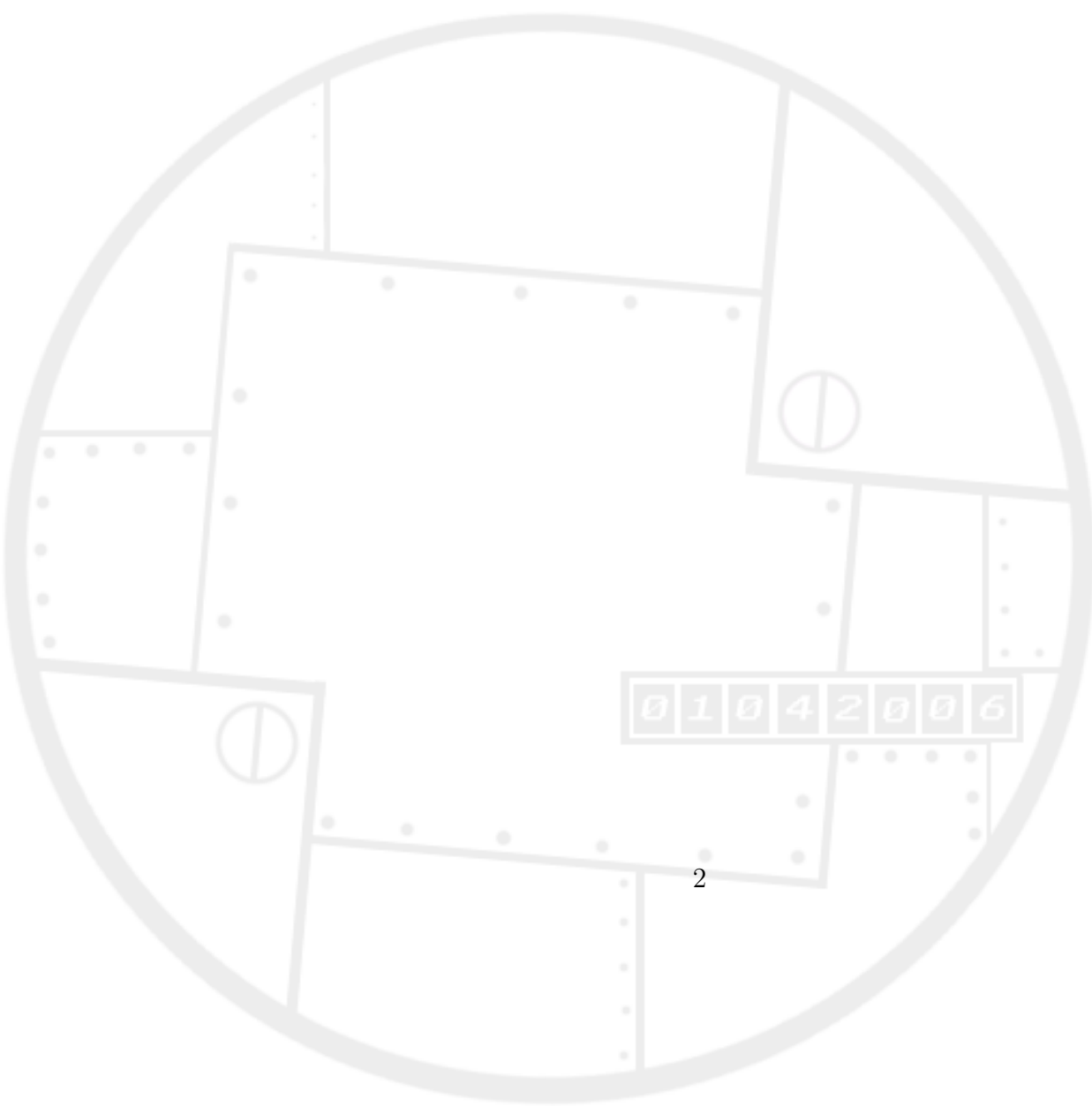
1 Introdução

Este documento descreve o sistema RODA do ponto de vista técnico, com detalhes sobre a implementação, pré-requisitos e configuração dos vários componentes.

No capítulo Pré-requisitos (Secção 2) são descritos os pré-requisitos de cada componente do RODA.

No capítulo Configuração (Secção 4) são descritos todos os ficheiros de configuração para cada componente do RODA.

O capítulo Desenvolvimento (Secção 5) contém informações sobre o desenvolvimento dos componentes do RODA e como configurar as ferramentas para desenvolver o RODA a partir do código fonte.



2 Pré-requisitos

O RODA foi desenvolvido majoritariamente em sistemas GNU/Linux, nomeadamente na distribuição Ubuntu, e nunca foi testado em outros sistemas ou distribuições. Sendo o RODA um sistema com uma arquitectura baseada em serviços, pode ser distribuído por várias máquinas, para melhorar a sua escalabilidade. Os componentes mais importantes são o RODA-Core, RODA-WUI, RODA-Migrator e RODA-in-installer.

Este capítulo descreve as dependências dos vários componentes em termos das ferramentas que necessitam de ser instaladas no sistema operativo para suportar o RODA. Junto com as descrições estão os comandos para instalar essas ferramentas num sistema Ubuntu (8.04).

2.1 RODA-Core

O RODA-Core é o componente central do RODA. Para instalar num sistema Ubuntu (8.04) é necessário instalar os seguintes componentes:

- Ant

```
$ sudo apt-get install ant ant-optional
```

- OpenLDAP

```
$ sudo apt-get install slapd
```

- Clam AntiVirus¹

```
$ sudo apt-get install clamav
```

- Java 5

```
$ sudo apt-get install sun-java5-bin
```

ou Java 6

```
$ sudo apt-get install sun-java6-bin
```

- VsFTPD (The Very Secure FTP Daemon) e o módulo LDAP para PAM (Pluggable Authentication Modules)

¹<http://www.clamav.net>

```
$ sudo apt-get install vsftpd libpam-ldap
```

- Um servidor SMTP (como o postfix no Ubuntu).

```
$ sudo apt-get install postfix
```

2.2 RODA-Migrator

O RODA-Migrator é uma aplicação independente do RODA-Core e do RODA-WUI, no entanto estes necessitam do RODA-Migrator. O RODA-Migrator disponibiliza uma série de serviços de conversão baseados nas seguintes ferramentas:

- ImageMagick² - converte imagens entre variados formatos de imagens.
- Ghostscript - converte documentos PDF em imagens.
- OpenOffice - converte vários formatos de texto.
- Mencoder e w32codecs/w64codecs - converte vários formatos de vídeo.
- Soundconverter, gstreamer0.10-ffmpeg, gstreamer-plugins-ugly e gstreamer0.10-plugins-ugly-multiverse - converte vários formatos de áudio.

Para instalar estes componentes no Ubuntu 8.04 é necessário activar um repositório extra (Medibuntu)³ e depois executar os comandos:

```
$ sudo apt-get install imagemagick ghostscript mencoder \
openoffice.org openoffice.org-headless openoffice.org-java-common \
soundconverter gstreamer0.10-plugins-base gstreamer0.10-ffmpeg \
gstreamer0.10-plugins-ugly gstreamer0.10-plugins-ugly-multiverse \
gstreamer0.10-pitfdll
```

Para sistemas 32 bits:

```
$ sudo apt-get install w32codecs
```

Para sistemas 64 bits:

```
$ sudo apt-get install w64codecs
```

²<http://www.imagemagick.org>

³<https://help.ubuntu.com/community/Medibuntu>

2.3 RODA-Migrator-Win

O RODA-Migrator-Win é em tudo igual ao RODA-Migrator, mas disponibiliza serviços baseados em ferramentas para Windows (eg. Microsoft Office).

- *Microsoft Office 2007*⁴.
- *2007 Microsoft Office Add-in: Microsoft Save as PDF or XPS*⁵ - converte documentos de texto em formato .doc e .docx para PDF/A.

2.4 RODA-in-installer

O RODA-in-installer permite descarregar o instalador do RODA-in, podendo até gerar instaladores com o plano de classificação embebido. É uma aplicação web que corre dentro de um J2EE web server container, como Apache Tomcat⁶ e precisa de aceder ao RODA-Core. Em sistemas de 64 bits também terá de ser instalado:

- MinGW 32 binary utilities

No ubuntu:

```
$ sudo apt-get install mingw32-binutils
```

2.5 RODA-WUI

O RODA-WUI é uma aplicação web que contém a interface com o utilizador. Esta aplicação web corre dentro de um J2EE web server container, como Apache Tomcat⁷, precisa de aceder ao RODA-Core e ao RODA-in-installer, e também:

- Apache, Apache PHP module, PHP, PHP MySQL module, MySQL.
- A “Mail Transport Agent” instalado no computador local (“localhost”) (eg. Postfix no Ubuntu).

No Ubuntu:

```
$ sudo apt-get install postfix libmm14 libmcrypt4 libltdl3
```

⁴<http://office.microsoft.com/>

⁵<http://www.microsoft.com/downloads/details.aspx?FamilyID=4d951911-3e7e-4ae6-b059-a2e79ed87041&displaylang=en>

⁶<http://tomcat.apache.org>

⁷<http://tomcat.apache.org>

3 Instalação

Consultar o documento “RODA - Manual de instalação”.



4 Configuração



4.1 RODA-Core

Configuração básica

A configuração básica situa-se no ficheiro:

```
$RODA_HOME/core/config/roda-core.properties
```

Aqui é possível alterar:

- A localização do serviço Fedora
- Nome e palavra-passe do administrador do Fedora (*fedoraAdmin*)
- Dados de acesso à base de dados do serviço **Logger**
- Nome e palavra-passe do utilizador visitante **guest**
- Nome e palavra-passe do utilizador administrador **admin**
- Dados de acesso ao LDAP
- Configuração do prefixo do serviço *Handle*
- Valores por omissão para o código do país e código do repositório em registos de descrição.

Existem também outras propriedades neste ficheiro de configuração que não devem ser alteradas para efeitos de configuração do sistema.

- Identificadores de *datastreams* do Fedora
- Nomes de utilizadores e grupos com permissões fixas e obrigatórias.

Configuração do serviço “Logger”

A configuração básica do serviço **Logger** situa-se no ficheiro:

```
$RODA_HOME/core/config/logger.properties
```

Aqui é possível alterar:

- Dados de acesso à base de dados do serviço **Logger**

Configuração do serviço “Statistics”

A configuração básica do serviço `Statistics` situa-se no ficheiro:

```
$RODA_HOME/core/config/statistics.properties
```

Aqui é possível alterar:

- Dados de acesso à base de dados do serviço `Statistics`

Configuração do serviço “Scheduler”

A configuração básica do serviço `Scheduler` situam-se os ficheiros:

```
$RODA_HOME/core/config/scheduler.properties  
$RODA_HOME/core/config/quartz.properties
```

Aqui é possível alterar:

- Dados de acesso à base de dados do serviço `Scheduler`
- Dados de acesso à base de dados do componente `Quartz`.

Configuração do serviço “Reports”

A configuração básica do serviço `Reports` situa-se no ficheiro:

```
$RODA_HOME/core/config/reports.properties
```

Aqui é possível alterar:

- Dados de acesso à base de dados do serviço `Reports`

Configuração dos “plugins”

A configuração básica dos *plugins* situa-se no ficheiro:

```
$RODA_HOME/core/config/plugins.properties
```

Aqui é possível alterar:

- A directoria onde o `RODA-Core` deve procurar os *plugins*.

4.1.1 RODA-Core Plugins

Tarefas de ingestão

A configuração básica das tarefas de ingestão situa-se no ficheiro:

```
$RODA_HOME/core/config/ingest.properties
```

Aqui é possível alterar:

- Dados de acesso à base de dados do serviço de ingestão.
- A directoria onde são guardados os SIPs durante o processo de ingestão.
- A lista de estados de um SIP durante o processo de ingestão.
- A lista dos estados iniciais e finais do processo de ingestão.
- As propriedades de cara tarefa de ingestão. Estas propriedades incluem o estado inicial e final dos SIPs para cada tarefa de ingestão, isto determina a sequência das tarefas no processo de ingestão.

Tarefa de normalização e conversores

A tarefa de normalização de representações durante a ingestão usa os *plugins* de conversão e cada *plugin* de conversão usa um conversor.

As configurações da tarefa de normalização estão no ficheiro:

```
$RODA_DEV/roda-plugins/src/pt/gov/dgarq/roda/plugins/ingest/normalization.properties
```

Aqui é possível alterar:

- Que plugin de conversão deve ser executado para cada tipo de representação.

As configurações dos *plugins* de conversão estão nos ficheiros `.properties` dentro da directoria:

```
$RODA_DEV/roda-plugins/src/pt/gov/dgarq/roda/plugins/converters
```

Para cada um destes ficheiros é possível alterar:

- O endereço do serviço de migração que o conversor deve usar.
- O subtipo das representações a que o conversor se aplica (`image/mets+jpeg`, `audio/flac`, etc).

4.2 Munin

Munin é um serviço de monitorização, composto por duas partes, o `munin-node`, que recolhe a informação de cada máquina segundo os plugins instalados, e o `munin`, que recolhe informação dos vários `munin-nodes` definidos e gera os gráficos e html. Em todos os servidores a monitorizar deve ser instalado um `munin-node`.

O `munin-node` para Windows pode ser adquirido em <http://www.jory.info/serendipity/index.php?/categories/4-Munin-Node-for-Windows>. Nesta página é possível descarregar a versão “MSI Installer” que instala um serviço que fica a correr no Windows a recolher informação.

Caso a categoria de monitorização dos discos “HDD” não funcione, esta opção deve ser desactivada no ficheiro

```
C:\Program Files\Munin Node for Windows\munin-node.ini
```

alterando a linha com a opção `hd` para

```
hd=0
```

O `munin-node` para Ubuntu 8.10 pode ser instalado com:

```
$ sudo apt-get install munin-node munin-plugins-extra \
libxml-simple-perl mailutils
```

Após instalação, é possível definir os plugins utilizados no `munin-node` criando links na pasta `/etc/munin/plugins` para os plugins que se encontram nas pastas `/usr/share/munin/plugins` e `$RODA_HOME/ui/munin/plugins`. Todos os plugins instalados pelo sistema estão na pasta `/usr/share/munin/plugins`, o plugin para `jboss` desenvolvido pelo RODA está em:

Seguidamente é necessário criar as configurações necessárias, mediante os plugins instalados, em:

```
/etc/munin/plugin-conf.d/munin-node
```

Estas configuração definem os parametros dos plugins:

- Estado do armazenamento, como percentagem de utilização do disco, percentagem de utilização dos inodes e estado de I/O

- Estado do servidor JBoss, como número de acessos, utilização da memória da maquina virtual Java, número de threads e volume
- Estado do servidor MySQL, como throughput, número e distribuição de queries e slow queries, e threads utilizadas
- Estado da rede, como erros e tráfico nas várias interfaces e o netstat
- Estado do servidor de mail, como a fila de espera e o throughput
- Estado do processos, como o rácio de fork, o número de processos e o VMstat
- Estado do sistema, como a taxa de utilização do processador, a entropia, interrupções, carga do sistema, utilização da memória, utilização da tabela de ficheiros, utilização da tabela de inodes, utilização da Swap
- Tempo desde a última reinicialização

Plugin Munin para MySQL

As configurações do plugin do MySQL, para uma máquina de 32 bits, devem ser:

```
[mysql*]
user root
env.mysqladmin RODA_HOME/database/mysql-5.1.30-linux-i686-glibc23/bin/mysqladmin
env.mysqlopts --defaults-extra-file=/root/.my.cnf
```

Para uma máquina de 64 bits:

```
[mysql*]
user root
env.mysqladmin RODA_HOME/database/mysql-5.1.30-linux-x86_64-glibc23/bin/mysqladmin
env.mysqlopts --defaults-extra-file=/root/.my.cnf
```

RODA_HOME deve ser substituído o caminho real para a directoria onde o RODA está instalado.

Em ambas as situações, deve ser criado um ficheiro em /root/.my.cnf, com permissões de leitura apenas para root, com o conteúdo:

```
[client]
host      = localhost
user      = USERNAME
password  = PASSWORD
socket    = SOCKET_PATH
```

Em que `USERNAME` é o nome de utilizador a utilizar para aceder à base de dados, `PASSWORD` é a palavra-passe, e `SOCKET_PATH` é o caminho para o socket de MySQL a utilizar. Caso o Munin esteja a monitorizar o MySQL da layer de dados, o caminho será `/tmp/mysql.sock`. Caso a monitorização esteja a ser feita no RODA-WUI, o caminho será `/tmp/wui-mysql.sock`.

Plugin Munin para JBoss

O plugin para JBoss não precisa de configuração no JBoss instalado por omissão. No entanto, se adicionar autenticação ou encriptação à aplicação web-console do JBoss, necessita de seguir as configurações explicadas em 4.3.

4.3 Segurança do JBoss

A configuração por omissão do JBoss deixa aplicação de gestão do JBoss (web-console e jmx-console) abertas ao público sem autenticação nem encriptação. Para resolver este ponto deve ser seguido o manual em <http://www.jboss.org/community/docs/DOC-12190>.

No manual é explicado como adicionar autenticação às aplicações e apenas permitir ligações seguras a estas. No entanto, não se pode esquecer de configurar devidamente o plugin de JBoss do Munin, para apontar para estas novas definições. Em `/etc/munin/plugin-conf.d/munin-node` deve ler-se⁸:

```
[jboss*]
env.url https://%s:%s@127.0.0.1:%d/web-console/status?XML=true
env.ports 8443
env.user USERNAME
env.password PASSWORD
```

Um pré-requisito deste plugin para aceder a HTTPS é o pacote `libcrypt-ssleay-perl`:

```
$ sudo apt-get install libcrypt-ssleay-perl
```

Para ainda maior segurança no JBoss, devem ser impedidas conexões SSL com cifras fracas. Para garantir esta propriedade é necessário editar o ficheiro:

```
$_RODA_HOME/jboss/server/default/deploy/jboss-web.deployer/server.xml
```

⁸`USERNAME` e `PASSWORD` devem ser substituídos pelos valores definidos ao seguir o manual

E adicionar ao conector de SSL o atributo:

```
ciphers="SSL_RSA_WITH_RC4_128_MD5, SSL_RSA_WITH_RC4_128_SHA,  
TLS_RSA_WITH_AES_128_CBC_SHA, TLS_DHE_RSA_WITH_AES_128_CBC_SHA,  
TLS_DHE_DSS_WITH_AES_128_CBC_SHA, SSL_RSA_WITH_3DES_EDE_CBC_SHA,  
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA, SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA"
```

No JBoss integrado no instalador do RODA estas optimizações de segurança já estão implementadas, mas é crucial **mudar as passwords de acesso às consolas**.



4.4 RODA WUI

Configuração básica

A configuração básica situa-se no ficheiro:

```
$RODA_DEV/roda-wui/src/config/roda-wui.properties
```

Aqui é possível alterar:

- A localização do serviço RODA-Core
- A localização do serviço RODA-in-installer
- A localização do serviço de monitorização Munin
- O nome de utilizador e palavra-passe a utilizar pelo RODA-WUI em acções de controlo
- As permissões de utilizador reconhecidas
- As permissões necessárias para aceder a cada um dos menus
- Definição de disseminadores
- Mapeamento dos disseminadores às classes e formatos de representação
- A directoria de cache das derivadas de disseminação
- Vários parâmetros para cada um dos disseminadores integrados no RODA-WUI

Modificar valores por omissão da metainformação descritiva

Ao criar o plano de classificação no RODA-WUI, os objectos criados são inicializados metainformação descritiva definível no ficheiro:

```
$RODA_DEV/roda-wui/src/config/roda-element-defaults.properties
```

Modificar frases nas páginas web

Todas as frases usadas nas páginas web do RODA-WUI estão definidas em ficheiros de propriedades (extensão .properties) sob o directório:

```
$RODA_DEV/roda-wui/src/config/i18n/client
```

Estas estão divididas em vários ficheiros, dependendo a que sub-componente pertencem.

Modificar frases nos relatório em PDF

No componente de listas dinâmicas do RODA-WUI é possível descarregar um PDF (ou CSV) com os mesmos parâmetros das lista. Este PDF contém algumas frases traduzidas, que estão definidas em vários ficheiros, de acordo ao tipo de dados a que a lista se refere:

```
$RODA_DEV/roda-wui/src/config/i18n/server/ContentAdapterHelperMessages_pt_PT.properties  
$RODA_DEV/roda-wui/src/config/i18n/server/EventManagerMessages_pt_PT.properties  
$RODA_DEV/roda-wui/src/config/i18n/server/IngestListReportMessages_pt_PT.properties  
$RODA_DEV/roda-wui/src/config/i18n/server/StatisticsListReportMessages_pt_PT.properties
```

Modificar a estrutura dos relatório em PDF

Existem dois tipos de relatórios, um baseado no Content-Adapter, usado nas listas dinâmicas explicadas acima, e outro baseado no Report (resultado de uma tarefa). Ambos relatórios PDF estão estruturados numa série de templates XML-FO, relativos a cada parte diferenciada do relatório. Estes templates podem ser encontrados em:

```
$RODA_DEV/roda-wui/src/config/reports/xml-fo
```

Modificar mensagens de rejeição padrão

Ao rejeitar um pacote no RODA-WUI, é possível escolher uma mensagem padrão. Estas mensagens padrão são definidas no ficheiro:

```
$RODA_DEV/roda-wui/src/config/i18n/server/IngestListRejectMessages_pt_PT.properties
```

A chave da mensagem padrão corresponde ao nome da mensagem de rejeição, que aparece na combo box. O valor corresponde à própria mensagem que será enviada ao produtor⁹.

Modificar email de verificação de email

Quando um utilizador se regista no RODA-WUI, é enviado um email para o endereço electrónico definido, para que este possa ser confirmado. O conteúdo desse email pode ser alterado no ficheiro:

```
$RODA_DEV/roda-wui/src/config/mail/templates/emailverification_html.vm
```

Outras propriedades, como o assunto deste email, podem ser alteradas no ficheiro:

```
$RODA_DEV/roda-wui/src/config/mail/velocity.mail.properties
```

⁹O utilizador pode sempre modificar a mensagem padrão antes de enviar

Modificar email de recuperação da palavra-passe

Caso um utilizador esqueça o nome de utilizador ou palavra-passe, é possível recuperá-los. O sistema envia um email ao endereço electrónico definido pelo utilizador, com um link para mudar a palavra-passe. O conteúdo deste email pode ser alterado no ficheiro:

```
$RODA_DEV/roda-wui/src/config/mail/templates/recoverlogin_html.vm
```

Outras propriedades, como o assunto deste email, podem ser alteradas no ficheiro:

```
$RODA_DEV/roda-wui/src/config/mail/velocity.mail.properties
```

Modificar email de notificação ao produtor de rejeição de pacote

Ao rejeitar manualmente um pacote, é dada a oportunidade de notificar imediatamente o produtor sobre esta rejeição. Um email é enviado ao endereço electrónico definido pelo produtor, no qual o SIP é identificado e a razão da rejeição manual. O conteúdo deste email pode ser alterado no ficheiro:

```
$RODA_DEV/roda-wui/src/config/mail/templates/notifyproducer_html.vm
```

Outras propriedades, como o assunto deste email, podem ser alteradas no ficheiro:

```
$RODA_DEV/roda-wui/src/config/mail/velocity.mail.properties
```

Modificar conteúdo das páginas estáticas

Algumas das páginas do RODA-WUI são estáticas e estão definidas em ficheiros HTML:

- Página inicial

```
$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/home/public/Home.html
```

- Sobre o RODA

```
$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/about/public/About.html
```

- Sobre o RODA > Serviços

```
$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/about/public/Services.html
```

- Políticas e documentos técnicos

`$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/about/public/Policies.html`

- I&D

`$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/about/public/ResearchDevelopment.html`

- Contactos

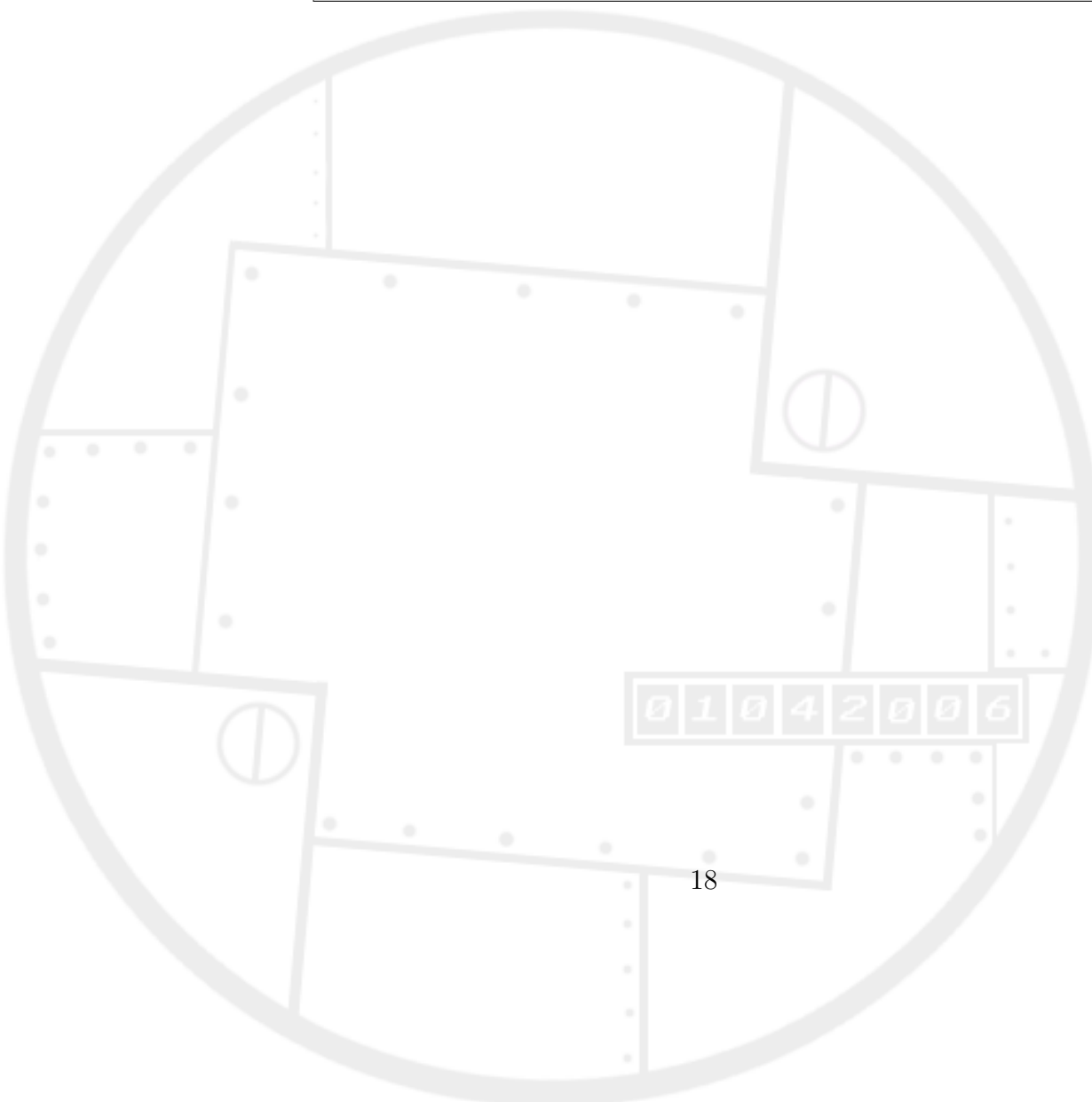
`$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/about/public/Contacts.html`

- Sobre o registo

`$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/about/public/AboutRegister.html`

- Ajuda

`$RODA_DEV/ui/roda-wui/src/pt/gov/dgarq/roda/wui/about/public/AboutHelp.html`



4.5 RODA In

Modificar configuração básica

A configuração básica situa-se no ficheiro:

```
$RODA_DEV/roda-in/config/roda-sip-creator.properties
```

Aqui é possível alterar:

- A localização do serviço RODA-Core
- A localização do indicador da versão do RODA-in
- A localização do serviço de actualização do RODA-in
- As directorias onde são guardados os dados do RODA-in

Modificar frases na aplicação

Todas as frases usadas na aplicação RODA-in estão definidas no ficheiro:

```
$RODA_DEV/roda-in/config/messages.properties
```

Modificar instruções de envio offline

As instruções para envio offline dos SIPs criados estão em formato HTML no ficheiro:

```
$RODA_DEV/roda-in/config/OfflineSendInstructions.html
```

Modificar o tema (Look&Feel)

O RODA-in usa o Look&Feel NimRod para melhorar a aparência da aplicação. É possível alterar as cores usadas no tema com o ficheiro:

```
$RODA_DEV/roda-in/config/NimRODThemeFile.theme
```

Um editor de temas NimRod pode ser usado para mais facilmente alterar o tema. Para abrir o editor, use:

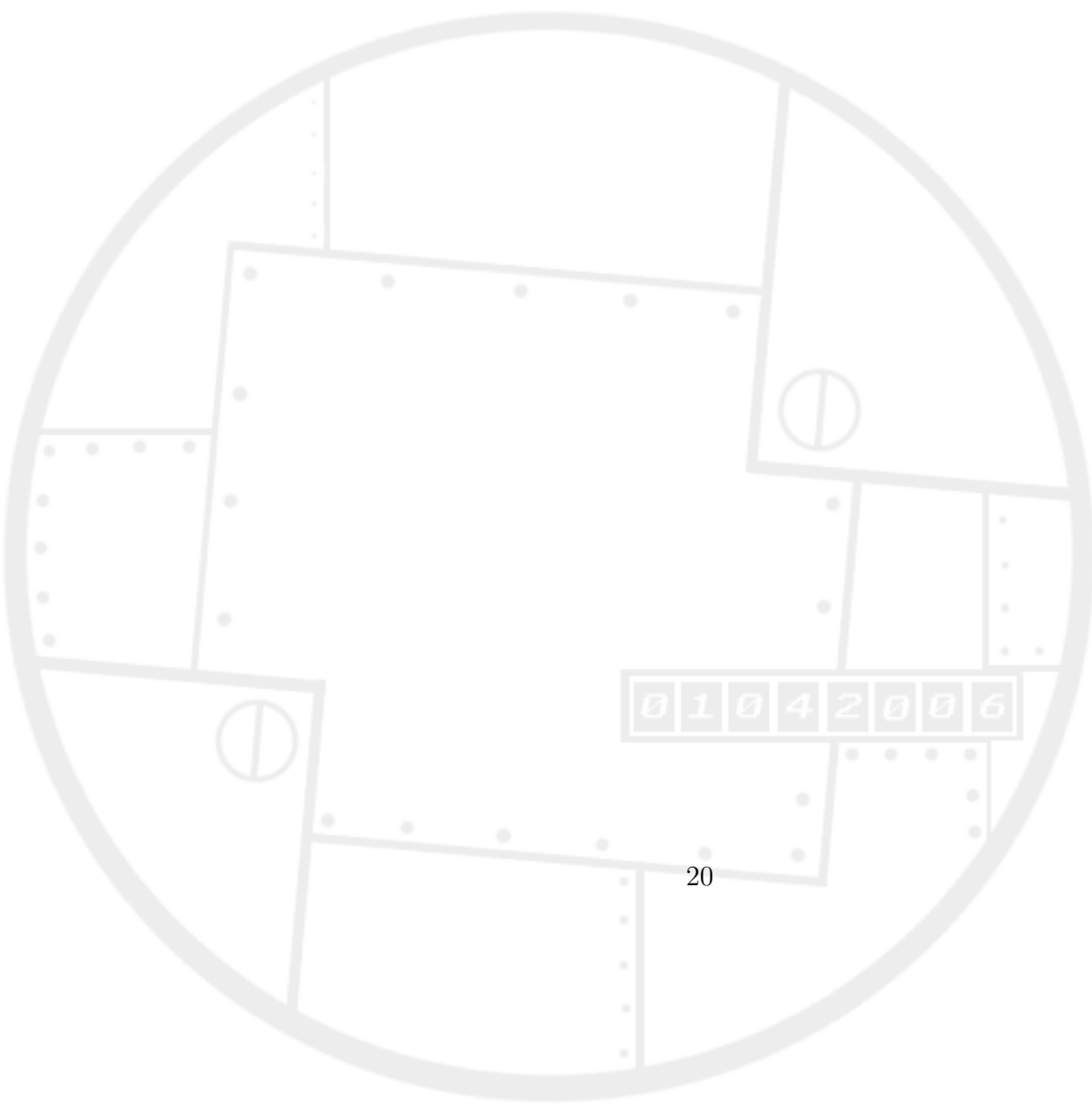
```
java -jar $RODA_DEV/roda-in/lib/nimrod1f-1.0e.jar
```

E depois abra o ficheiro de tema NimRod definido acima.

Modificar a versão da aplicação

O RODA-in detecta automaticamente actualizações à versão disponível on-line. Para que esta detecção seja possível, um ficheiro é usado para manter a versão da aplicação. Ao alterar a aplicação, e antes do deploy, a versão deve ser incrementada para que um aviso apareça em todas as versões instaladas pelos produtores. O ficheiro que mantém a versão situa-se em:

```
$RODA_DEV/roda-in/config/roda-in-version.properties
```



4.6 RODA In Installer

Modificar configuração básica

A configuração básica situa-se no ficheiro:

```
$RODA_DEV/roda-in-installer/src/config/roda-in-installer.properties
```

Aqui é possível alterar a localização do serviço RODA-Core, usado para gerar instaladores com o plano de classificação embebido.

Modificar frases do instalador

O instalador suporta Português e Inglês. Algumas frases que são usadas pelo instalador podem ser modificadas nos ficheiros:

```
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/resources/packsLang.xml_por  
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/resources/packsLang.xml_eng
```

Modificar as informações (Readme)

As informações, ou Readme, apresentadas na instalação, estão formatadas em HTML e podem ser alteradas nos ficheiros:

```
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/resources/Readme.por.html  
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/resources/Readme.eng.html
```

Modificar a licença

A licença, apresentada na instalação, está formatada em HTML e pode ser alterada nos ficheiros:

```
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/resources/Licence.por.html  
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/resources/Licence.eng.html
```

A licença em Português deve estar codificada com HTML entities.

4.7 RODA-Migrator

A configuração básica situa-se no ficheiro:

```
$RODA_DEV/roda-migrator/config/roda-migrator.properties
```

Aqui é possível alterar:

- A directoria onde são guardadas as representações convertidas.
- O endereço do serviço RODA-Core.
- Configuração de acesso à base de dados do disseminador PhpMyAdmin.

4.8 RODA-Migrator-win

A configuração básica situa-se no ficheiro:

```
$RODA_DEV/roda-migrator-win/config/roda-migrator.properties
```

Aqui é possível alterar:

- A directoria onde são guardadas as representações convertidas.
- O endereço do serviço RODA-Core.
- Localização do executável `doc2pdf.exe` para o serviço de conversão de documentos Microsoft Word.

5 Desenvolvimento

Nesta secção é descrito como continuar o desenvolvimento do projecto, desde como obter o código fonte, a sua estrutura, a melhor maneira de editar o projecto, compilar e implantar (deploy) o resultado.

Para desenvolver o RODA, são necessários alguns pré-requisitos:

- Cliente Subversion¹⁰ (eg. SmartSVN¹¹)
- Java Development Kit¹² (versão 5 ou superior)
- Apache Ant¹³
- Google Web Toolkit¹⁴ (para RODA-WUI)
- J2EE Container (e.g. Apache Tomcat 6¹⁵, JBoss 4.2.3¹⁶)

Aconselha-se também a instalação de:

- Eclipse IDE¹⁷
- Plugin de subversion para o Eclipse: Subclipse¹⁸
- Plugin de Google Web Toolkit para o Eclipse: Cypal Studio¹⁹
- Apache Directory Studio plugins for Eclipse Plugin (opcional)²⁰

Devem também ser definidas algumas variáveis de ambiente anteriormente à instalação:

\$RODA_DEV

Directoria onde se situará o código fonte (`dev`)

\$GWT_HOME

Directoria onde está instalado o Google Web Toolkit. Esta só é necessária para desenvolver o RODA-WUI

¹⁰<http://subversion.tigris.org/>

¹¹<http://www.syntevo.com/smartsvn/>

¹²<http://java.sun.com/javase/downloads/>

¹³<http://ant.apache.org/>

¹⁴<http://code.google.com/webtoolkit/>

¹⁵<http://tomcat.apache.org/>

¹⁶<http://www.jboss.org/jbossas/downloads/>

¹⁷<http://www.eclipse.org>

¹⁸<http://subclipse.tigris.org>

¹⁹<http://www.cypal.in/studio>

²⁰<http://directory.apache.org/studio/>

\$J2EE_CONTAINER_LIBS

Directoria onde estão instaladas as bibliotecas do container J2EE. No caso do Apache Tomcat, será a pasta `common/lib`.



5.1 Código fonte

5.1.1 Obter

O código fonte está disponível pelo controlador de versões Subversion. Para obter a sua cópia do código fonte, execute²¹:

```
svn co svn://portal.roda.dgarq.gov.pt/roda2/trunk roda-trunk
```

Em trunk está disponível não só o código fonte do projecto, mas também todos os documentos do projecto, incluindo relatórios, apresentações, esquemas do desenho das interfaces, documentação sobre os esquemas de metainformação utilizados e instaladores. No entanto, se só estiver interessado no código fonte para desenvolvimento, pode usar:

```
svn co svn://portal.roda.dgarq.gov.pt/roda2/trunk/dev roda-trunk/dev
```

5.1.2 Estrutura

A directoria base (trunk) é composta por:

Design

Esquemas e recursos usados no desenho das interfaces

dev

Código fonte para desenvolvimento

Docs

Relatórios, apresentações, documentação, livros, reuniões, manuais, etc.

installer

Instalador do RODA

NonFree

Recursos proprietários adquiridos pela DGARQ

Schemas

Recursos e documentação sobre os esquemas de metainformação

Os componentes mais importantes do código fonte dentro da directoria *dev* são:

²¹Posteriormente, \$RODA_DEV deverá apontar para `roda-trunk/dev`

core/roda-core

Projecto com a lógica de negócio, RODA-Core.

core/fedora-client-utility

Projecto com o componente de ligação aos serviços do Fedora.

core/roda-core-plugins

Projecto com o gestor de *plugins* do RODA-Core.

core/roda-fedora-security

Projecto com o filtro para os serviços do Fedora.

roda-plugins

Projecto com os *plugins* do RODA-Core

roda-client

Cliente Java para aceder aos serviços de RODA-Core

ui/roda-wui

Projecto de Interface Web para o Utilizador, RODA-WUI

roda-handle

Projecto de integração com o Handle System

roda-migrator

Projecto com migradores/conversores baseado em Linux

roda-migrator-win

Projecto com migradores/conversores baseado em Windows

roda-migrator-client

Cliente para o serviço de migração/conversão

roda-in

Projecto para a aplicação desktop de auxílio ao produtor RODA-in

roda-in-installer

Projecto de criação de um instalador para o RODA-in

roda-common/roda-common-certification

Projecto de auxílio a criação de assinaturas digitais e validação das mesmas

roda-common/roda-common-content-adapter

Projecto de auxílio para acesso genérico a dados de diversas fontes: MySQL, LDAP, Fedora Commons, etc.

roda-common/roda-common-convert-db

Projecto com utilitários para converter entre vários formatos de bases de dados.

roda-common/roda-common-data

Projecto com as classes de dados partilhadas entre o RODA-Core, todos os componentes deste e pelo RODA-Client.

roda-common/roda-common-db

Projecto com componentes básicos para utilizar bases de dados.

roda-common/roda-common-format-utility

Projecto com utilitários para detecção de formatos e extracção de propriedades de ficheiros.

roda-common/roda-common-ingest

Projecto com componentes partilhados entre o RODA-Core e os *plugins* de ingestão.

roda-common/roda-common-metadata

Projecto com utilitários para transformar dados em formatos XML (METS/EAD/PREMIS/XACML).

roda-common/roda-common-migrator

Projecto com as partes comuns entre *roda-migrator* e *roda-migrator-win*.

roda-common/roda-common-servlet-security

Projecto com os componentes de segurança das aplicações.

roda-common/roda-common-sip-utility

Projecto com utilitários para ler e escrever SIPs.

shared-libs

Conjunto de bibliotecas utilizadas. Como boa prática, é mantida a documentação, e todas as bibliotecas permanecem nesta directoria. *Links* simbólicos são utilizados onde apropriado.

5.1.3 Eclipse

O método aconselhado para editar o código fonte é usando o Eclipse IDE. É também aconselhado a instalação de um plugin para Subversion (eg. Subclipse) e para editar o RODA-WUI, um plugin para o Google Web Toolkit (Cypal Studio).

Aconselha-se também que seja definido um *J2EE Runtime Library* com a API de referência em `$RODA_DEV/shared-libs/servlet-api-2.4` (Figura 1).

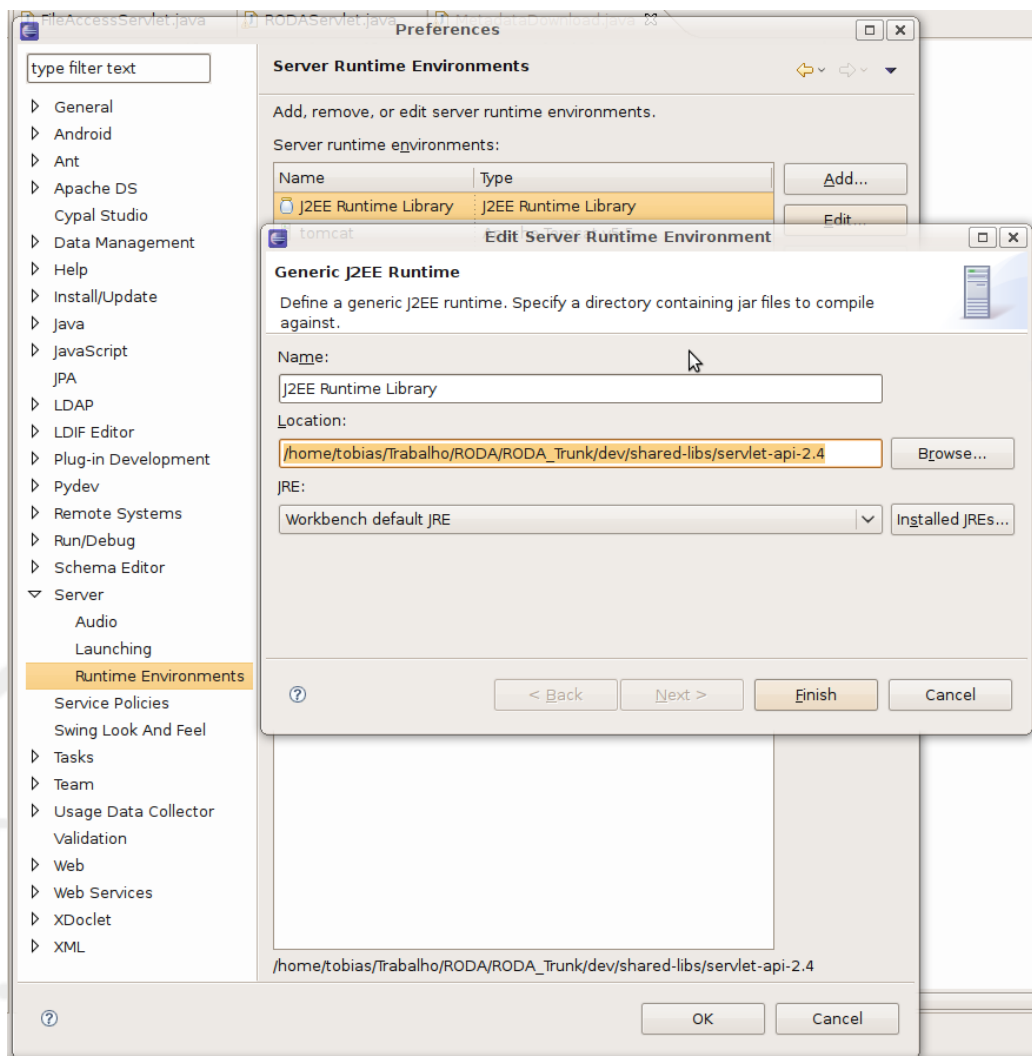


Figura 1: Eclipse: Configurar a *J2EE Runtime Library*

Para adicionar os projectos ao Eclipse, seleccione **File > Import...** e escolha a opção **Existing Projects into Workspace** (Figura 2).

Escolha a directoria `$RODA_DEV`, todas as subdirectorias serão pesquisadas e todos os projectos do RODA aparecerão na lista de projectos (Figura 3). Para importar os projectos basta seleccionar **Finish**.

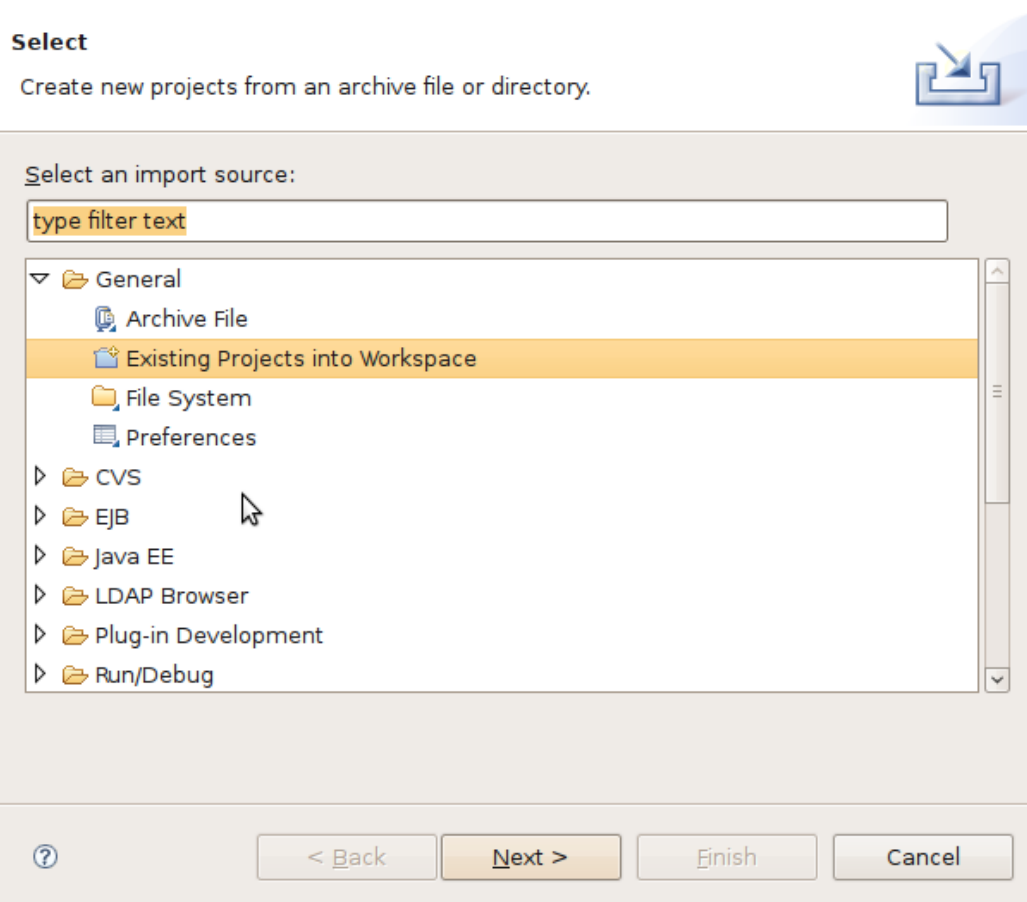


Figura 2: Eclipse: Janela para importar proyectos

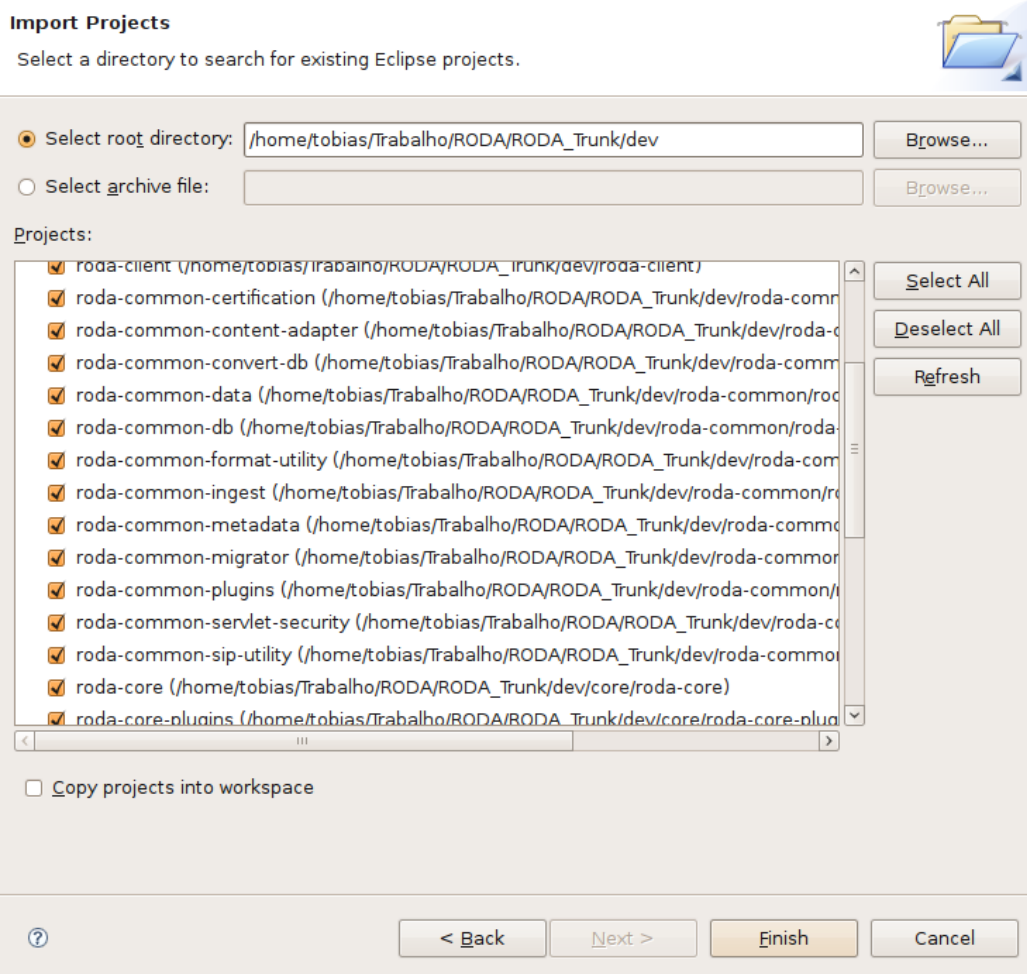


Figura 3: Eclipse: Janela com a lista dos projectos em \$RODA_DEV

5.2 Compilação e implementação

Na directoria base de cada projecto existem Ant scripts para compilar o mesmo. É apenas necessário mudar a directoria actual para a directoria base do projecto em causa e executar:

```
$ ant deploy
```

Para compilar todo o projecto, mude a directoria actual para `$RODA_DEV` e execute a comando supracitado. O resultado de uma compilação pode ser encontrado na pasta `$RODA_DEV/depoy`.

Para recriar um instalador com as alterações efectuadas, mude para a directoria `$RODA_DEV/./installer` e execute o comando

```
$ ant
```

Para actualizar uma instalação já existente terá de copiar o resultado da compilação para a pasta apropriada:

roda-plugins

O deploy dos plugins do RODA têm a forma de um conjunto de jars, que terão de ser copiados para `$RODA_HOME/core/plugins`

webapps

Todo os outros componentes tomam a forma de aplicações web, i.e. um ficheiro war. Este deve ser copiado para `$RODA_HOME/tomcat/webapps` ou para `$RODA_HOME/jboss/server/default/deploy`.

5.3 Componentes do RODA e projectos associados

5.3.1 RODA Common

Todos os projectos que são partilhados por vários componentes estão na directoria \$RODA_DEV/roda-common.

roda-common-certification Este componente contém uma classe (`SignatureUtility`) com métodos para assinar e verificar assinaturas digitais.

roda-common-content-adapter Este componente contém as classes de base do `Content Adaptor`. O `Content Adaptor` serve de intermediário entre uma fonte de dados e o serviço que disponibiliza os dados aos clientes.

Existem 3 tipos de fontes de dados que podem ser “adaptadas”: bases de dados SQL, bases de dados RDF - iTQL e serviços de directoria - LDAP. Para cada uma destas fontes existe um “content adapter engine”, `ITQLContentAdapterEngine`, `SQLContentAdapterEngine` e `JndiContentAdapterEngine`, respectivamente.

A tarefa de um “content adaptor engine” é extrair dados da fonte de dados de acordo com o “Content Adaptor” especificado pelo cliente.

roda-common-data Este componente contém todas as classes de dados e excepções que podem ser trocadas entre os vários componentes do RODA. Estas classes não contêm funcionalidade, apenas definem estruturas de dados. Todas as classes deste componente tem que ser serializáveis, definir métodos `get/set` para todas as variáveis e terem um constructor vazio.

roda-common-db Este componente contém a classe `DatabaseUtility` que serve de base a todas as classes que fazem acessos a bases de dados SQL.

roda-common-format-utility Este componente contém as classes `FormatUtility` e `JhoveUtility`.

A classe `FormatUtility` contém métodos para determinar o “mimetype” de um ficheiro. A classe `JhoveUtility` contém métodos para inspecionar um ficheiro e através da ferramenta JHOVE e devolver o resultado do JHOVE em forma de texto.

roda-common-ingest Este componente contém as classes `IngestManager` e `IngestDatabaseUtility`. A classe `IngestManager` é usada pelos serviços do `roda-core` para aceder ao estado dos SIPs e por todas as tarefas de

ingestão para manipular os SIPs. A classe `IngestDatabaseUtility` é a responsável por manipular as tabelas com a informação do SIPs (`SIPs` e `SIPStateTransitions`).

roda-common-metadata Este componente contém classes para transformar classes de dados em documentos XML e vice-versa (`EAD-C`, `METS`, `PREMIS` e `XACML`).

Para documentos `EAD-C` existe a classe `EadCHelper`.

Para documentos `METS` existe a classe `MetsHelper`. No RODA os documentos `METS` são usados para guardar a estrutura de uma representação do tipo “digitalized work“ (`DigitalizedWorkMetsHelper`) e para guardar a estrutura de um SIP (`SIPMetsHelper`).

Para documentos `PREMIS` existe a classe `PremisHelper`. Os documentos `PREMIS` podem ser objectos (`PremisObjectHelper`), agentes (`PremisAgentHelper`) ou eventos (`PremisEventHelper`).

Para documentos `XACML`, usados pelo Fedora para definir as políticas de acesso de um objecto, existe a classe `PolicyHelper`.

roda-common-migrator Este componente contém as classes comuns a todos os serviços de migração (`roda-migrator` e `roda-migrator-win`).

roda-common-plugins Este componente contém as classes comuns entre os vários componentes de plugins (`roda-plugins` e `roda-maintenance-plugins`).

roda-common-servlet-security Este componente contém vários “servlet filters“ usados pelos componentes do RODA (`LDAPAuthenticationFilter`, `RoleFilter` e `RemoteHostFilter`) e a classe `LdapUtility`, responsável por todos os acessos ao serviço LDAP.

roda-common-sip-utility Este componente contém a classe `SIPUtility` que contém métodos para ler e escrever SIPs e classes para auxiliar na criação de representações dos vários tipos (por exemplo, `AudioRepresentationBuilder`, `VideoRepresentationBuilder`, etc).

5.3.2 RODA Core

O RODA-Core está dividido em vários projectos.

roda-core O `roda-core` contém todos os serviços e todas as classes de apoio aos serviços organizadas nos seguintes pacotes:

services neste pacote estão todas as classes que implementam os vários serviços do `RODA-Core`. As classes que implementam “Web Services” estendem a classe `RODAWebService`. As classes que implementam serviços HTTP estendem `RODAServlet`. As classes que implementam “web services” funcionam apenas como intermediárias entre o exterior (os clientes) e as funcionalidades internas implementadas nas classes descritas de seguida. Cada método de um “web service” está sempre definido com a seguinte sequência de passos:

1. marcar a hora antes de executar a tarefa,
2. executar a tarefa,
3. marcar a hora de termino da tarefa e calcular a duração da mesma,
4. registar a tarefa executada
5. devolver os resultados caso existam.

O registo é feito através do método `RODAWebService#registerAction` ou `RODAServlet#registerAction` que fazem uso do serviço `Logger` para guardar os dados.

ingest neste pacote estão as classes relacionadas com a gestão de SIPs (`AcceptSIPTask`, `SIPUploadTask`).

logger neste pacote estão as classes relacionadas com a gestão de “logs” (`LoggerManager`, `LoggerDatabaseUtility`, etc).

reports neste pacote estão as classes relacionadas com a gestão de relatórios (`ReportManager`, `ReportDatabaseUtility`, etc).

scheduler neste pacote estão as classes relacionadas com a gestão de tarefas (`SchedulerManager`, `TaskInstanceDatabaseUtility`, etc).

statistics neste pacote estão as classes relacionadas com a gestão de estatísticas (`StatisticsManager`, `StatisticsDatabaseUtility`, etc).

As classes que não estão dentro dos pacotes descritos em cima são:

RodaWebApplication esta classe contém dois métodos (`contextInitialized` e `contextDestroyed`) que são chamados quando a aplicação `roda-core` é iniciada e finalizada pelo “web container”, respectivamente.

BrowserHelper esta classe contém a implementação das funcionalidades do serviço `Browser`.

EditorHelper esta classe contém a implementação das funcionalidades do serviço `Editor`.

IngestHelper esta classe contém a implementação das funcionalidades do serviço `Ingest`.

UserManagementHelper esta classe contém a implementação das funcionalidades do serviço `UserManagement`.

UserRegistrationHelper esta classe contém a implementação das funcionalidades do serviço `UserRegistration`.

fedora-client-utility Este componente contém as classes responsáveis pelo acesso aos serviços do Fedora. A classe `FedoraClientUtility` é um extensão da classe `FedoraClient`, para além dos métodos já disponíveis nas APIs do Fedora implementa alguns métodos para suportar as funcionalidades dos serviços `Browser`, `Editor`, `Ingest`. A classe `FedoraRISearch` contém métodos para aceder ao serviço `risearch` do Fedora para suportar as funcionalidades do serviço `Browser`. A classe `FedoraGSearch` contém métodos para aceder ao serviço Fedora Generic Search para suportar as funcionalidades do serviço `Search`.

roda-core-plugins Este componente contém as classes relacionadas com a gestão de plugins (`PluginManager`) e a interface pública para plugins (`Plugin`).

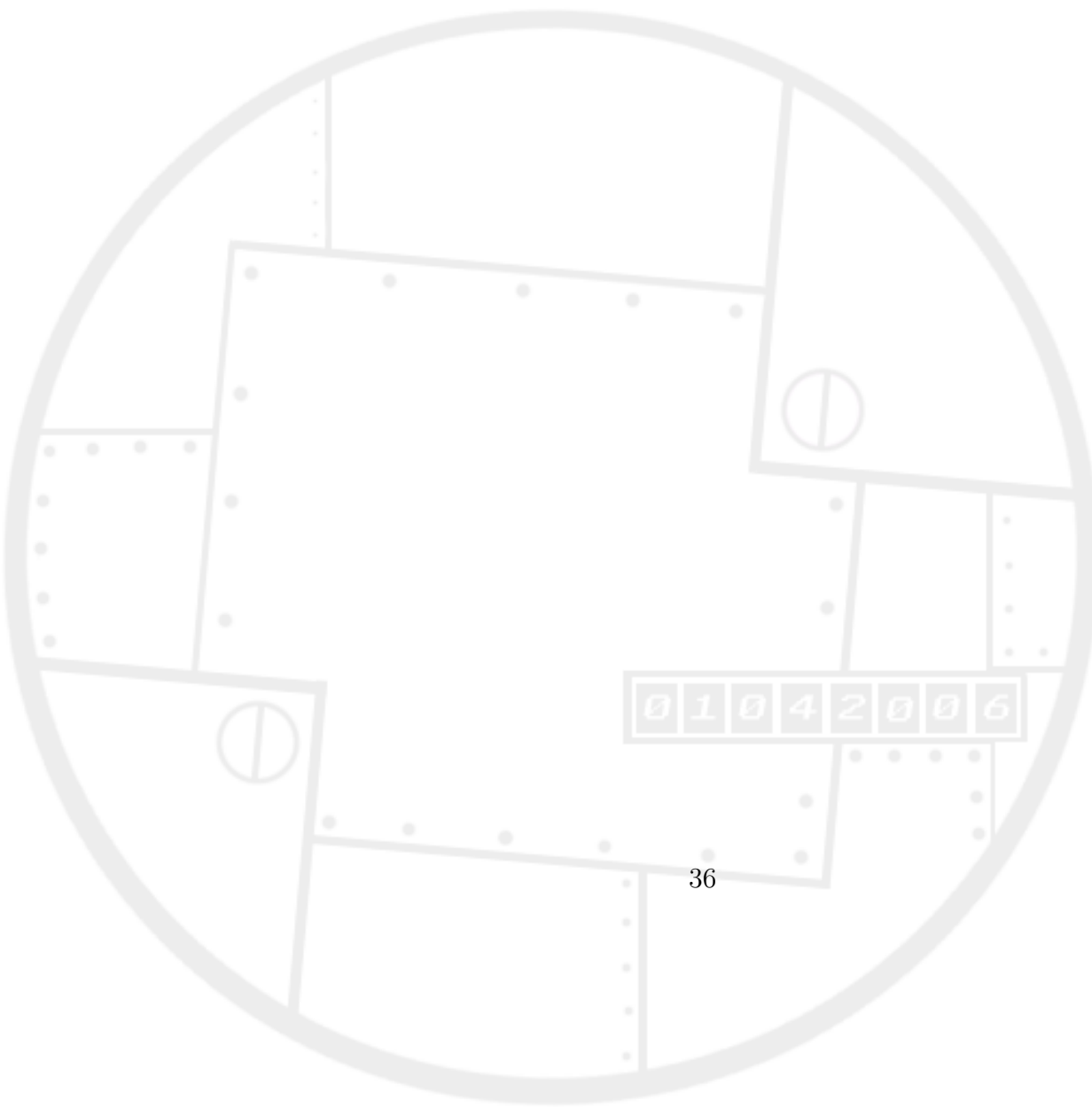
roda-fedora-security Este componente contém a classe `RODAFedoraFilter`. Este filtro é para ser usado na “aplicação web” Fedora e serve para adicionar informação sobre os utilizadores a cada pedido ao Fedora. Esta informação, como “roles” e “grupos” é necessária para a aplicação das políticas de acesso aos objectos Fedora.

5.3.3 RODA Core Plugins

roda-plugins Este componente contém todos os plugins básicos do RODA-Core como as tarefas de ingestão, a tarefa verificação de integridade e os conversores de representações, a tarefa de recolha de dados estatísticos, etc.

roda-maintenance-plugins Este componente contém plugins que podem ser usados para realizar tarefas de gestão do RODA que não são práticas de se fazerem manualmente, como por exemplo, rejeitar todos os pacotes de um determinado produtor num determinado estado.

Contém também plugins para efectuar actualizações/correções à estrutura de dados interna do RODA, como por exemplo, renomear uma determinada propriedade ou corrigir as permissões de base de determinados objectos.



5.3.4 RODA Client

roda-client O `roda-client` é um projecto que pode ser usado por qualquer outro projecto e por aplicações de terceiros para facilitar o acesso aos serviços do RODA-Core.



5.3.5 RODA In

O RODA-in divide-se em três fontes de código:

src

Classes Java de painéis e janelas que compõem a interface gráfica

config

Os vários ficheiros de configuração

resources

Recursos usados, nomeadamente imagens

A classe principal é a `SIPCreator`, que funciona como ponto de entrada para a aplicação. Esta classe inicializa o tema (Look&Feel) e cria a janela principal (`MainFrame`), que por sua vez cria o painel principal (`MainPanel`). Este painel divide-se em duas zonas fundamentais, a árvore com o plano de classificação à esquerda²² (e *toolbars* associadas), e uma painel removível à direita. Ao escolher um item na árvore, o painel removível é substituído por um painel apropriado ao tipo do item. O item pode ser de três tipos:

Item do plano de classificação

Item só de leitura que corresponde a um nodo do plano de classificação definido na pré-ingestão. O plano de classificação pode ser alterado no RODA-WUI pelo administrador, ou arquivista com permissões para tal, e actualizado online no RODA-in. O painel associado (`DOPanel`) mostra a metainformação descritiva de maneira não editável.

Pacote de ingestão (SIP)

Raiz do pacote de ingestão. Estes pacotes podem ser criados e removidos pelo produtor no RODA-in. O produtor também pode fazer *drag&drop* do SIP para outro local no plano de classificação. O painel associado (`SIPPanel`) mostra informação básica do SIP, como localização e estado, e permite guardar ou validar o SIP.

Documento do pacote de ingestão

Documento que compõe o SIP. Este pode ser um documento simples ou composto. Um documento composto pode incluir outros documentos compostos ou simples. Cada documento composto ou simples pode também ter várias representações. O painel associado (`SIPDOPanel`) mostra variados painéis em separadores (*tabs*). O primeiro será obrigatoriamente o que contém a metainformação descritiva num painel

²²A árvore que contém o plano de classificação é implementada pela classe `FondsTree`

editável (`DOPanel`). Outros separadores podem ser criados ou removidos com painéis que definem as variadas representações.

As classes que implementam os painéis de metainformação descritiva, tal como o suporte para a edição desta, podem ser encontradas no pacote `description`. As classes que implementam os painéis para criação e edição das representações podem ser encontrados no pacote `representation`.

Depois dos SIPs serem criados, é possível enviá-los para o RODA, de forma online ou offline. Um painel com a lista de SIPs por enviar é apresentado, dando a escolha ao utilizador se quer enviar de forma online, enviado directamente para o RODA pela internet, ou offline, exportando os SIPs para uma directoria à escolha, e apresentando instruções sobre o envio por correio. Todas as classes que implementam os painéis que suportam a funcionalidade de envio (online e offline) dos SIPs encontram-se no pacote `upload`.



5.3.6 RODA In Installer

O RODA-in-installer dividi-se em duas áreas bem definidas. O código fonte (Java) do serviço web que serve o instalador estático ou gerado e os recursos usados na compilação do gerador pelo Izpack e, opcionalmente para Windows, o Launch4j.

Todo o código fonte está na pasta `src`. O serviço web está implementado na classe `GetRodaIn`.

O ficheiro de configuração está em `config/roda-in-installer.properties`.

Caso seja pedido um instalador personalizado para um utilizador (com o seu plano de classificação embebido), um é gerado com a ajuda do IzPack. Caso o instalador pedido seja para o sistema operativo Windows, o instalador em Java em envolto num executável Windows, com a ajuda do Launch4j.

O recursos estáticos para a geração do instalador pelo IzPack situa-se em:

```
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/resources}
```

A configuração do instalador é definida nos ficheiros `ant-install.xml`, usado pelo ant para criar os instaladores genéricos, e `user-install.xml`, usado pelo serviço web para criar os instaladores personalizados. Documentação sobre como alterar a configuração dos instaladores encontra-se em:

```
$RODA_DEV/shared-libs/IzPack-4.2.0/doc
```

A configuração do Launch4j pode ser encontrada em:

```
$RODA_DEV/roda-in-installer/WebContent/WEB-INF/launch4j.xml
```

Informação sobre como alterar a configuração encontra-se em:

```
$RODA_DEV/shared-libs/launch4j/web/docs.html
```

5.3.7 RODA WUI

A Web User Interface (WUI) pode ser dividida em dois macro-pacotes: `wui` e `disseminators`. O pacote `wui` é a base da interface: o menu, as várias páginas estáticas, e interfaces para os serviços de navegação, pesquisa, administração de utilizadores e tarefas, estatísticas, relatório de acções, ingestão. O pacote `disseminators` contém os serviços de acesso ao objectos guardados, seja na forma original, derivada, ou como um pré-visualizador.

As classes dentro de ambos os pacotes utilizam a classe `RodaClientFactory` para adquirir o cliente para o RODA-Core. Nesta classe existem métodos para autenticar com um certo utilizador (login) e sair da sessão autenticada (logout). Nesta classe também são proporcionados métodos para aceder à configuração geral.

No macro-componente `wui`, os pacotes seguem a estrutura de módulos GWT, definindo o módulo num ficheiro com extensão `.gwt.xml`, e dividindo o código em recursos públicos (pacote `public`), código a correr no cliente (pacote `client`), e código a correr no servidor (pacote `server`). Existem vários módulos GWT no macro componente `wui`:

Main

Módulo principal, ponto de entrada à página web. Contém o esquema inicial: a banner, o painel de login, o menu, a breadcrumb, e o painel central, onde são apresentados os outros módulos. Este módulo monitoriza a história da aplicação²³ e chama o módulo correcto para ser apresentado no painel central.

Home

Contém a página inicial (estática)

About

Contém todas as páginas (estáticas) de todos os itens do menu *Sobre o RODA*

Dissemination

Contém as páginas estáticas do menu *Catálogo* e *Ajuda* respectiva, classes e métodos comuns para disseminação, como o painel de meta-informação descritiva, e sub-módulos para navegação e pesquisa no repositório.

Browse

Módulo de navegação no repositório, no qual é definido a árvore

²³a história é o estado mantido pela anchor no URL, sensível ao botão back e forward dos browsers e bookmarks, e.g. <http://www.example.com/app#anchor>

de hierarquia de metainformação descritiva, e os painéis de metainformação descritiva e de preservação e representações.

Search

Módulo que contém classes comuns para pesquisa, como o painel de resultados da pesquisa, e dois sub-módulos: pesquisa básica e pesquisa avançada.

BasicSearch

Painel de pesquisa básica, acessível no item do menu *Pesquisa*

AdvancedSearch

Painel de pesquisa avançada, acessível no item do menu *Pesquisa avançada*

Management

Módulo que contém as páginas estáticas do menu *Administração* e *Ajuda* respectiva, e sub-módulos de edição de metainformação, gestão de utilizadores e tarefas, e estatísticas.

MetadataEditor

Conjunto de painéis usados pelo navegador que proporcionam suporte para edição de metainformação descritiva

EventManager

Painel de monitorização e gestão de tarefas

Statistics

Painel de apresentação das várias estatísticas recolhidas pelo sistema

UserManagement

Painel de monitorização e gestão de utilizadores, e também monitorização das acções efectuadas pelos mesmos

Ingest

Contém as páginas estáticas do menu *Ingestão* e da *Ajuda* respectiva, e três sub-módulos: pré-incorporação, enviar e estado.

PreIngest

Painel que apresenta uma parte estática e outra parte com o plano de classificação do utilizador autenticado.

IngestSubmit

Painéis para submeter um SIP criado, ou criar e submeter um novo SIP

IngestList

Painel de monitorização do estado dos SIPs

Common

Conjunto de painéis e classes comuns a todos os módulos. Neste módulo painéis (para cliente) comuns como: botões, janelas, seleccionadores de datas, painéis de loading, wrapper de HTML, listas dinâmicas compatíveis com o **Content Adapter**, logging, etc. Também são definidas classes (para servidor) comuns como: VelocityMail²⁴, download de ficheiros, download de metainformação, download de relatórios em PDF e CSV, logging, etc. São também definidos dois sub-módulos: captcha e file upload.

Captcha

Módulo que oferece suporte para a geração de testes para diferenciar humanos de computadores.

FileUpload

Módulo que oferece suporte para o upload de ficheiros por parte dos utilizadores.

No macro-componente **disseminators** estão divididos em pacotes cada um dos disseminadores suportados por omissão:

AIPDownload

Disseminador para download da representação tal como ela está guardada no AIP. Caso a representação seja composta por um só ficheiro, esse ficheiro será enviado. Caso esta seja composta por vários ficheiros, então um zip composto por estes ficheiros é criado e enviado.

SimpleViewer

SimpleViewer²⁵ é um visualizador de imagens baseado em Flash e com uma interface optimizada para fotos. Este disseminador utiliza os conversores existentes no RODA-Migrator para converter imagens em imagens optimizadas para o SimpleViewer e definir a sua configuração.

FlashPageFlipFree

FlashPageFlip²⁶ é visualizador de imagens baseado em flash e com uma interface estruturada como um livro. Este disseminador utiliza os conversores existentes no RODA-Migrator para converter documentos de

²⁴Sistema de templates para envio de emails

²⁵<http://www.airtightinteractive.com/simpleviewer/>

²⁶<http://www.flashpageflip.com>

texto ou imagens para imagens optimizadas para o FlashPageFlip e definir a sua configuração. A versão do FlashPageFlip usada neste disseminador é grátis.

FlashPageFlipPaid

Este disseminador é em tudo igual ao anterior excepto usar uma versão paga do FlashPageFlip (Basic XML). Esta versão acrescenta as funcionalidades de zoom, ecrã completo, impressão de páginas, mudar papel de parede, controlo por teclado, entre outras.

MediaPlayer

Disseminador de áudio ou vídeo. Este disseminador usa os conversores existentes no RODA-Migrator para criar recursos em FLV ou MP3. Um media player baseado em Flash, denominado JW FLV Media Player²⁷, é usado para apresentar os recursos criados.

PhpMyAdmin

Disseminador de bases de dados. Este disseminador usa os conversores existentes no RODA-Migrator para inserir uma base de dados em formato DBML no DBMS MySQL. O conversor especializado em PhpMyAdmin também adiciona o dicionário de dados dentro de tabelas especiais. Posteriormente, uma versão optimizada do PhpMyAdmin²⁸ é usada para oferecer ao utilizador um sistema poderoso de navegação, pesquisa e exportação de informação da base de dados.

Signature

Disseminador que usa o projecto `roda-common-certification` para criar assinaturas digitais da disseminação AIPDownload. Um ficheiro zip é criado e enviado com a disseminação do AIPDownload e a assinatura digital respectiva.

Common

Conjunto de classes comuns a todos os disseminadores, como um serviço de cache, onde as derivadas são mantidas, uma página de loading, onde o utilizador é redireccionado enquanto os recursos são criados, entre outras classes utilitárias.

²⁷<http://www.longtailvideo.com/players/jw-flv-player/>

²⁸<http://www.phpmyadmin.net>

5.3.8 RODA Migrator

Os projectos relacionados com o serviço RODA-Migrator são o `roda-migrador`, `roda-migrador-win` e `roda-migrador-client`.

Todos os serviços de migração implementam a interface `SynchronousConverter`. Para facilitar a implementação de novos serviços existe já uma classe `AbstractSynchronousConverter` que pode ser estendida por qualquer implementação de um serviço de migração.

roda-migrador Este projecto contém a “aplicação web” com vários conversores baseados em ferramentas disponíveis num sistema GNU/Linux.

roda-migrador-win Este projecto contém a “aplicação web” com vários conversores baseados em ferramentas disponíveis num sistema Windows.

roda-migrador-client Este projecto contém o cliente para os serviços disponibilizados pelas aplicações `roda-migrador` e `roda-migrador-win`.



5.3.9 RODA Handle

O projecto `roda-handle` contém a classe `RODAHandleStorage`, uma implementação da interface `HandleStorage`, que é usada pelo servidor `Handle` para resolver os “handles” do RODA.

