



Universidade do Porto
Faculdade de Engenharia

FEUP



Gil António Oliveira da Silva

Painel de Controlo da Qualidade de Dados

Estágio Curricular Realizado no CICA

004(047.3) LEIC
EIC5202 2005/SILg

LEIC
Porto, 2005

Faculdade de Engenharia da Universidade do Porto
Licenciatura em Engenharia Informática e Computação



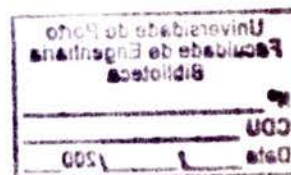
Painel de Controlo da Qualidade de Dados
no
CICA

Relatório do Estágio Curricular da LEIC 2004/2005

Gil António Oliveira de Silva

Orientador na FEUP: Prof. Gabriel David
Orientador no CICA: Eng. Manuel Machado

Março de 2005



Resumo

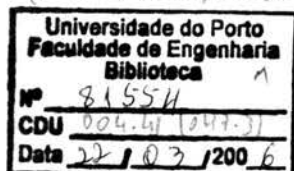
O SIGARRA é o sistema de informação académico da UP. É usado actualmente por mais de 20000 utilizadores que geram uma grande quantidade de dados, infelizmente nem sempre correctos. Este trabalho consistiu na construção de um *framework* capaz de suportar a implementação de indicadores da qualidade de dados para o SIGARRA.

Os indicadores são agrupados em painéis de controlo, cada um deles orientado a um tipo de responsável de dados. Um painel de controlo deve ser estruturado de forma a evidenciar a existência de problemas de dados, permitindo depois a navegação na procura da origem desses problemas. Uma dificuldade de monta surgiu da disparidade da natureza dos indicadores. Foram por isso analisadas as propriedades de alguns indicadores e a forma como devem ser organizados e consultados. Foi elaborado um protótipo que permite implementar os indicadores em SQL, com o apoio de um interface em HTML para o registo e formatação da apresentação desses indicadores e para a composição dos painéis. A identificação de problemas foi feita com o auxílio de heurísticas rudimentares. Foram estudadas e implementadas formas de, após a captura dos aspectos mais importantes de cada indicador, apresentá-los ao responsável de dados na forma mais adequada, por vezes em gráfico, por vezes em texto. Os valores dos indicadores são em si informação preciosa e por isso foi instituído um registo histórico. Foi construído um mecanismo de cálculo periódico de indicadores, e um mecanismo de avisos que permite ao responsável de dados alertar os donos dos mesmos para a existência de problemas.

Agradecimentos

Sou dono de um enorme pudor nestas questões. Sem querer mencionar nomes, gostaria de agradecer a todos aqueles que directa ou indirectamente me ajudaram neste trabalho. Obrigado!

004 (047.3) L866/ECC5202 2005 / SIL8



Índice de Conteúdos

Índice

1	Introdução.....	1
1.1	CICA/USINF.....	1
1.2	SiFEUP/SIGARRA.....	1
1.3	Objectivos.....	1
1.4	Estrutura e âmbito do relatório.....	1
2	Painel de controlo.....	3
2.1	Visão.....	3
2.2	Ponto de partida.....	3
2.3	Tecnologias.....	3
3	Reflexão sobre os indicadores.....	6
3.1	Indicadores.....	6
3.2	Dinâmica dos indicadores.....	7
3.3	Parâmetros.....	7
3.4	Estados.....	7
3.5	Reutilização e registo de código.....	8
3.6	Gráficos.....	9
3.7	A consulta do histórico.....	9
3.8	Relevância do indicador.....	10
3.9	Explicação.....	10
3.10	Organizando os indicadores.....	10
3.11	A consulta.....	12
3.12	Avisos.....	13
4	Descrição do módulo.....	15
4.1	Consultando o painel.....	15
4.2	Definindo um template.....	17
4.3	Registando funções.....	17
4.4	Definindo um indicador.....	18
4.5	Indicadores compostos.....	19
4.6	Indicadores agregados.....	19
4.7	Painéis.....	20
4.8	Cálculo.....	20
4.9	Histórico.....	21
4.10	Avisos.....	21
4.11	Actores.....	21
5	Trabalho.....	22
5.1	Método de trabalho usado.....	22
5.2	Cronologia.....	22
5.3	Implementação.....	24
5.4	Alguns mecanismos.....	30
6	Conclusões e perspectivas.....	44

6.1 Aspectos positivos.....	44
6.2 Aspectos negativos.....	44
6.3 Cumprimento dos objectivos.....	45
6.4 Perspectivas.....	46
7 Referências e Bibliografia.....	47

1 Introdução

A LEIC é a Licenciatura em Engenharia Informática e Computação da FEUP. “A LEIC pretende transmitir os conhecimentos metodológicos, científicos e técnicos que permitam satisfazer as necessidades crescentes em recursos humanos com educação superior nos domínios da arquitectura de sistemas informáticos e das tecnologias de informação e comunicação.” [LEIC] No último semestre da licenciatura, o aluno tem de efectuar um estágio curricular em ambiente laboral. É neste contexto que o presente relatório de insere.

1.1 CICA/USINF

O Centro de Informática Correia de Araújo [CICA] (CICA) é um serviço central da Faculdade de Engenharia da Universidade do Porto [FEUP](FEUP). Tem por missão disponibilizar e assegurar a operacionalidade de recursos e serviços de informática para toda a comunidade da FEUP, promovendo a sua utilização e inovação. Emprega cerca de 60 pessoas em áreas como a manutenção da rede informática da faculdade, gestão das salas de informática, *help desk*, ou desenvolvimento de sistemas de informação.

O presente trabalho foi realizado na Unidade de Sistemas de Informação (USINF) do CICA, que desenvolve e mantém o sistema de informação da FEUP e da Universidade do Porto (UP). Na USINF trabalham actualmente cerca de 20 pessoas no desenvolvimento, na manutenção e no apoio aos utilizadores dos sistemas de informação aí produzidos. A grande maioria destas pessoas trabalha exclusivamente no SIGARRA.

1.2 SiFEUP/SIGARRA

O SIGARRA [SIG] é um sistema de informação desenvolvido originalmente no CICA, com o nome de SiFEUP, com o objectivo de apoiar de forma progressiva todas as actividades académicas da FEUP. Posteriormente foi integrado com o GAUP, sistema de back-office de gestão de alunos, e com o GRHUP, sistema de gestão de recursos humanos desenvolvido pelo IRICUP (Instituto de Recursos e Iniciativas Comuns da Universidade do Porto). As suas funcionalidades principais são descritas em [SIG2], onde são divididas em 3 classes principais: processo pedagógico; recursos humanos; comunicação; e ainda um conjunto de funcionalidades de desenvolvimento recente não classificadas.

1.3 Objectivos

O objectivo deste módulo é construir uma ferramenta de monitorização sistemática da qualidade de dados do SIGARRA (SI + GA + GRH) que permita ao administrador de dados de uma instituição acompanhar a respectiva evolução, detectar potenciais problemas e desencadear acções preventivas e correctivas. O paradigma de interacção proposto é o de painel de controlo essencialmente gráfico, com duas vistas principais: a vista completa, destinada a uma análise sistemática, e a vista de alarmes, que mostra os indicadores que estão fora dos parâmetros.

1.4 Estrutura e âmbito do relatório

Este relatório pretende documentar as ideias e o trabalho mais importantes que surgiram ou foram executados durante os seis meses que durou este estágio.

A secção 2 do trabalho descreve e sustenta a necessidade da ferramenta, estabelece um ponto de partida para o desenvolvimento da mesma, e descreve o ambiente organizacional e tecnológico em que o trabalho foi desenvolvido.

A secção 3 tenta (o autor pensa não poder aspirar a mais) estabelecer uma abstracção na qual se possa encaixar a problemática da construção dos indicadores e que possa fornecer uma base para a arquitectura de uma solução. É uma tentativa de abordar a quixotesca tarefa de construir um painel de controlo capaz de agregar dados de muitas origens e naturezas diferentes sem perder o juízo. A secção é livre, são apresentadas ideias, conceitos, prós e contras, propõem-se nomes e outros artefactos úteis para um cavaleiro andante que não se quer perder. Como sempre acontece, a abstracção não é perfeita nem universal, mas pôs-nos a caminho e serviu como base para o planeamento de tudo aquilo que se fez.

A secção 4 é uma especificação alto nível do sistema a desenvolver. Escolheu-se a abordagem XPeana das *user stories*, embora com um nível de detalhe um pouco mais elevado. A razão da escolha deve-se unicamente às preferências do estagiário e à sua maneira de pensar. As *user stories* são agrupadas em áreas funcionais.

A secção 5 descreve vários aspectos do trabalho. A secção 5.1 tenta descrever o método de trabalho utilizado. O trabalho foi elaborado num ambiente muito informal, e não foi seguida nenhuma metodologia padronizada, embora a metodologia XP (Extreme Programming) tenha servido aqui e ali de inspiração. A secção 5.2 descreve alguns acontecimentos importantes e ordena-os cronologicamente. A ordem cronológica das coisas teve muita influência nas escolhas tomadas. Tivessem algumas ideias surgido mais cedo e teriam tido a primazia sobre outras entretanto desenvolvidas: *fortuna imperatrix mundi*. A secção 5.3 descreve a arquitectura do sistema. A secção 5.4 continua a secção anterior, descrevendo os mecanismos mais importantes do sistema.

A secção 6 é a secção da retrospecção e da prospecção. O que se fez? O que queria ter feito? O que se teria feito diferente? O que está bem? O que está mal? O que falta fazer? Que mais se pode fazer? O que é que vale a pena melhorar? O Demónio de Maxwell, o tal que conhece todo o futuro e todo o passado do universo, secundado pelo seu fiel discípulo Gil Silva, especialista em painéis de controlo, responde a estas e outras perguntas. Não perca, na secção 6 (como se pode notar pelo tom, o autor gosta muito destas partes).

São ainda incluídos anexos. O anexo A apresenta uma partição das funcionalidades que serviu para guiar e monitorizar o projecto. O anexo B lista os indicadores estudados durante a implementação do sistema e que serviram de prova à sua eficiência, utilidade e funcionalidade. Os anexos C,D e E apresentam alguns detalhes técnicos que poderiam também caber no capítulo 5.

2 Painel de controlo

2.1 Visão

O SIGARRA é um sistema de informação cada vez mais vasto e é usado por um universo que ultrapassa já as vinte mil pessoas. Grande parte das actividades da FEUP e da UP (em menor escala) apoiam-se no SIGARRA. A informação gerada na utilização diária do sistema de informação é um recurso valioso mas subaproveitado. Por outro lado o sistema nem sempre é utilizado da melhor forma, por vezes até não é, prejudicando o melhor funcionamento das instituições em que está implementado.

Um sistema que permita a cada responsável o acesso fácil e resumido q.b. aos dados gerados nas actividades sob a sua alçada promoverá um controlo mais fino e informado dessas mesmas actividades e uma melhor e mais completa utilização do sistema de informação em actividades e por pessoas que denotem deficiências na utilização, levando em última análise à melhoria dos serviços prestados pela instituição.

O sistema cresceu muito desde a sua criação, e essa grandeza traz consigo dificuldades técnicas. A implementação de novas funcionalidades cada vez mais específicas exige frequentemente a alteração de funcionalidades antigas, aumentando a complexidade do código e afectando a sua manutenção, para além do risco de danos laterais frequentemente associado a estas alterações. O Painel de Controlo oferece uma alternativa em algumas situações: em vez de implementar regras específicas, controla-se os dados e monitoriza-se a sua violação, tomando-se uma atitude correctiva sempre que se justificar.

O painel de controlo é concebido para facilitar aos responsáveis de dados a tomada de consciência da existência de problemas. Neste contexto, a condensação espacial dos dados apresentados é um objectivo importante.

2.2 Ponto de partida

A gestão da qualidade dos dados tem sido feita de forma quase manual pelas pessoas envolvidas no *help desk* de segunda linha e gestão de dados (Susana Gaio, Elisabete Silva). Estas pessoas têm por isso um conhecimento mais profundo sobre as necessidades a este nível. Existe já um conjunto disperso de páginas de monitorização da qualidade de dados cujo código pode ser reaproveitado na implementação dos indicadores.

2.3 Tecnologias

Base de dados Oracle e SQL

A Oracle foi a empresa pioneira na exploração comercial das bases de dados relacionais. Partiram em 1977 à conquista do mercado de base de dados com uma ideia pioneira e que com o tempo se tornou o standard para a implementação de bases de dados, de modo que hoje quase todas as bases de dados são relacionais.

A Oracle comercializa um sistema de gestão de bases de dados relacional, actualmente na sua versão 10g. Este sistema permite guardar, aceder e transformar dados usando SQL, uma linguagem de interrogação e manipulação de dados.

O SIGARRA usa o sistema de gestão de bases de dados Oracle, e portanto toda a informação deverá ser guardada neste sistema.

PL/SQL

A linguagem SQL é uma linguagem de manipulação de dados, não é uma linguagem de programação genérica. A Oracle introduziu a PL/SQL [PLS] (Procedural Language) como forma de contornar esta limitação. O PL/SQL tem a vantagem (ou desvantagem segundo alguns) de estar fortemente integrado com o SQL. Isto traduz-se na poupança de código de tradução de tipos, por exemplo, facilitando a manutenção do código. Por outro lado a linguagem é algo pobre e cheia de *qui pro quos*, fomentando o abuso de truques em aplicações que ultrapassam um pouco o trivial. Uma das principais falhas está na tipagem doentivamente forte, impossibilitando o polimorfismo. A Oracle contornou recentemente a questão introduzindo objectos na linguagem, porém estes são ainda um aspecto lateral e mal integrado com o resto da linguagem. A ausência de namespaces ou outra forma de organização faz com que os vários componentes fiquem todos a monte, causando náuseas a mentes desorganizadas com dificuldade em encontrar *aquele* pacote ou objecto. Em geral, a pobreza da linguagem não está nas funcionalidades disponibilizadas (é até bastante rica em pacotes), mas mais na sua organização, filosofia e flexibilidade. Apesar disso tem as suas vantagens. É muito conveniente para, por exemplo, criar páginas e formulários web.

HTML e Javascript

A linguagem de marcação de documentos HTML nasceu do SGML e foi concebida para a apresentação e formatação de documentos na WWW. Hoje a WWW está quase completamente baseada no HTML. A linguagem de programação Javascript, desenvolvida pela Netscape nos primórdios da WWW permite manipular o HTML ao nível do browser e adicionar algum dinamismo ao conteúdo. Deve ser usado com parcimónia, porque há alguns problemas de compatibilidade entre browsers e porque existe a possibilidade por parte do utilizador de impedir a sua execução.

XML

A linguagem de marcação XML (Extensible Markup Language) é uma espécie de canivete suíço que serve para tudo e mais alguma coisa. Há muitos standards implementados usando XML. Por exemplo, o XHTML, que será o HTML da próxima geração, o XSL ou o SVG descritos abaixo.

O XML é usado para descrever dados em documentos, trabalhando ao nível da semântica através do uso de marcas. As marcas (ou *tags*) XML não são predefinidas, podendo o programador usar as que achar convenientes para descrever os seus dados. A grande vantagem do XML está na sua simplicidade, sendo simultaneamente inteligível por homens e por máquinas. Para usá-la plenamente é necessário desenvolver novos dialectos da linguagem ou aplicar outros já existentes e standardizados. No primeiro destes casos, é preciso ainda construir programas que entendam esse dialecto XML, processem o documento e façam algo com ele. No segundo caso, esses programas normalmente já existem e em diversos sabores, sendo o XML um elemento fundamental na sua interoperabilidade.

Relativamente à sintaxe do XML há dois conceitos chave a abordar: o de boa formação, isto é, todo o documento XML deve obedecer a um conjunto de regras de sintaxe próprias; e a validade, isto é a conformidade com um dialecto específico. A validade é controlada através

de documentos que são anexados ou referidos no XML. Há duas formas de o fazer actualmente: DTD e Schema.

XSLT e Xpath

Segundo o consórcio W3C [W3C], “a XSL é uma linguagem para construir folhas de estilo. Uma folha de estilo ... é um ficheiro que descreve como mostrar um documento XML ... Tem também: uma linguagem para transformar documentos: XSLT. Originalmente direccionada para a execução de operações de estilo complexas... é hoje usada como uma linguagem de processamento de XML generalista. A XSLT é largamente utilizada para outros propósitos que não XSL, como por exemplo para gerar uma página HTML a partir de XML...” [XSL]

A linguagem XSLT é constituída por um conjunto de regras de transformação de um documento XML aplicadas iterativamente que permitem transformar um documento XML de um formato para outro. De facto, é possível até transformar um documento XML num formato de texto. Será este essencialmente o uso que lhe vamos dar: construir código PL/SQL a partir de definições em XML.

A XSLT assenta num outro standard, o Xpath, considerado também parte integrante da XSL. O Xpath é uma linguagem para especificar endereços de partes do XML. Com ele é possível seleccionar elementos específico dentro de um ficheiro XML, que depois podem ser processados pela XSLT.

SVG

O SVG é uma linguagem para descrever gráficos bidimensionais e aplicações gráficas em XML. Suporta formas bidimensionais simples como círculos, rectângulos e polígonos, bem como outras formas gráficas mais avançadas. Contém também uma API acessível via Javascript ou ECMAScript que permite desenvolver aplicações gráficas usando SVG.

Num futuro próximo, com a implantação do XHTML é possível que o SVG se instale como a principal ferramenta de construção de gráficos vectoriais para a WWW. Usando namespaces será possível misturar código XHTML com código SVG, assim os browsers o suportem. É plausível até que substitua o HTML em algumas aplicações, uma vez que possui muitas das funcionalidades do HTML mas possui um poder gráfico acrescido.

O SVG é um standard publicado pelo consórcio W3C (World Wide Web Consortium).

GESSI

O módulo de gestão do sistema de informação é uma ferramenta criada no CICA para controlar e aumentar a configurabilidade do SIGARRA. Entre outras coisas, suporta a definição das opções disponíveis numa determinada instalação, o controlo das autorizações de acesso, ou a definição do *look* do SIGARRA.

3 Reflexão sobre os indicadores

3.1 Indicadores

O objectivo dos indicadores é condensar informação sobre um conjunto de dados num valor. A análise desse valor deve permitir a um responsável o reconhecimento da existência de falhas, erros ou inconsistências.

A definição do indicador está fortemente dependente da sua natureza. Os indicadores definidos podem inspeccionar dados completamente distintos, podem inspeccionar dados idênticos usando fórmulas completamente distintas, e os resultados calculados podem também ter natureza diferente. Podemos conceber um indicador como uma caixa negra, em que o interior da caixa e seu funcionamento são desconhecidos. Sabemos apenas o que introduzir na caixa, e que essa entrada será processada e resultará em algo à saída. Assim, um indicador:

- lê dados, ou *entradas*, a analisar
- aplica uma transformação, ou *fórmula* sobre esses dados com o objectivo de filtrar problemas ou produzir estatísticas e produz um
- *resultado*, condensando em si a informação relevante para a análise pretendida;
- o *resultado* é interpretado de forma a fornecer informação sobre o grau de perigosidade, encapsulado no conceito de *estado*.



Figura 3.1: Processo de cálculo

Os dados de *entrada* são originários, em geral, de registos na base de dados. A *fórmula* a correr sobre os dados está fortemente dependente dos dados de origem e do aspecto particular a analisar. A dependência dos dados de origem pode ser eliminada através do uso de uma forma intermédia de dados. Uma fórmula transforma os dados iniciais numa forma intermédia capaz de ser tragada pela *fórmula* de inspecção da qualidade de dados, que por sua vez os analisa e produz o resultado.

Esta aproximação em dois passos é mais limpa, permite a separação total entre dados e análise, mas traria algum *overhead* de performance e talvez de compreensibilidade devido à construção de fórmulas ser essencialmente feita em SQL. Assim optou-se pela solução em um passo, em que a fórmula de processamento sabe à partida quais os dados que deverá ler e incorpora em si informação sobre as entradas.

O *resultado* exige algum grau de normalização porque se pretende manipular os indicadores todos da mesma forma. Em todos os casos analisados até agora é possível condensar a informação sobre a qualidade de dados num número. Mas há indicadores eminentemente vectoriais em que não é viável condensar toda a informação num número. Nestes casos não padronizáveis deverá ser usado um mecanismo de produção e armazenamento dos resultados assente em XML. Consoante o indicador, o número pode representar grandezas diferentes, como uma taxa, uma paridade ou um valor absoluto. Esta é uma informação que convém guardar explicitamente.

3.2 Dinâmica dos indicadores

Também é preciso definir os mecanismos que disparam o cálculo de um indicador. Alguns indicadores devem ser refrescados periodicamente, outros só devem ser calculados a pedido do utilizador. A carga do sistema também tem de ser considerada. Alguns cálculos mais pesados devem ser evitados durante as horas do dia. Da mesma forma, cálculos com periodicidade semanal podem ser realizados aos fins-de-semana. Indicadores *leves*, cujo cálculo não ocupa muitos recursos do sistema podem ser calculados unicamente no momento da visualização, de forma que o utilizador visualiza sempre o estado actual do sistema.

Assim, o cálculo do indicador pode ser feito:

- periodicamente;
- apenas a pedido.

O pedido de cálculo pode ser implícito ou explícito.

- Se explícito, o utilizador deve pedir o cálculo clicando num comando;
- se implícito, o cálculo deve ser lançado sempre que o utilizador consultar o indicador; este método introduz alguma sobrecarga sobre o sistema e deve ser usado apenas em indicadores muito leves.

3.3 Parâmetros

Um outro aspecto importante é o da necessidade da parametrização dos indicadores. Tomemos o exemplo das fichas de disciplina: não faz sentido definir um indicador para cada curso. Podemos usar o mesmo indicador, fornecendo um meio de distinguir a origem dos dados e os resultados entre si usando parâmetros. Entra aqui o conceito de parâmetro do indicador.

Façamos uma analogia. Pensemos num aparelho de medida comum, digamos, um termómetro. O termómetro pode estar integrado numa máquina ou pode ser portátil e usado para medir a temperatura em tudo o que se preste a isso, fornecendo diferentes leituras da temperatura de diferentes objectos. Da mesma forma o indicador pode ser fixo ou pode ter parâmetros que lhe conferem liberdade, fornecendo resultados diferentes de origens de dados diferentes.

Cada indicador pode ter um conjunto arbitrário de parâmetros. Por exemplo, o indicador das fichas de disciplina pode ter três: ano lectivo, período lectivo e curso.

Embora introduzidos para qualificar os resultados, estes parâmetros podem ser manipulados e usados para outras coisas discutidas mais à frente como controlar o acesso aos painéis ou organizar os dados de uma forma hierárquica, permitindo a navegação no painel de controlo “aumentando o nível de detalhe à medida que se desce na hierarquia.” O parâmetro será um conceito chave no processo de agregação de resultados proposto mais adiante.

3.4 Estados

Após a obtenção do *resultado* é necessário elaborar uma interpretação simples do mesmo. O indicador deve conter em si uma heurística capaz de avaliar o resultado e dizer se é bom, se é mau ou se é assim-assim. Esta informação de perigosidade é encapsulada no conceito de *estado*. A heurística pode basear-se apenas no resultado ou pode considerar outras variáveis relevantes como o instante de cálculo para produzir um valor inteiro que simplifica e condensa a informação.

Pode haver várias categorias de estados, embora em número limitado. Alguns indicadores podem ter três estados, mostrados na forma de um semáforo, ou apenas dois. Esta concepção aumenta a informação disponibilizada em cada indicador mas introduz algumas complicações. Por exemplo, torna-se mais difícil identificar estados *críticos* ou calcular um estado composto (discutido mais à frente). Dentro da mesma categoria, o significado particular de cada estado também pode variar de indicador para indicador, e pode ser sempre explicitado.

3.5 Reutilização e registo de código

Cada indicador tem uma fórmula e uma heurística de estado próprias, sugerindo o uso de alguma forma de polimorfismo. Num perspectiva orientada a objectos há toda a vantagem em delegar em cada indicador a responsabilidade pelo seu próprio cálculo e pela determinação do seu próprio estado. A implementação de um mecanismo de registo deste tipo de código torna-se aconselhável, quer para a sua utilização num esquema polimórfico, quer para efeitos de cadastramento e manutenção.

Por outro lado deve ser promovida a reutilização de código. As fórmulas de cálculo exibem uma forte dependência relativamente aos dados e uma lógica muito variada. Não são decididamente terreno fértil para a reutilização. Por seu lado as heurísticas de estado serão usadas recorrentemente, sendo vulgar a utilização de uma mesma lógica com um ou outro factor de diferença. Esta necessidade inspira a implementação de um mecanismo de templates.

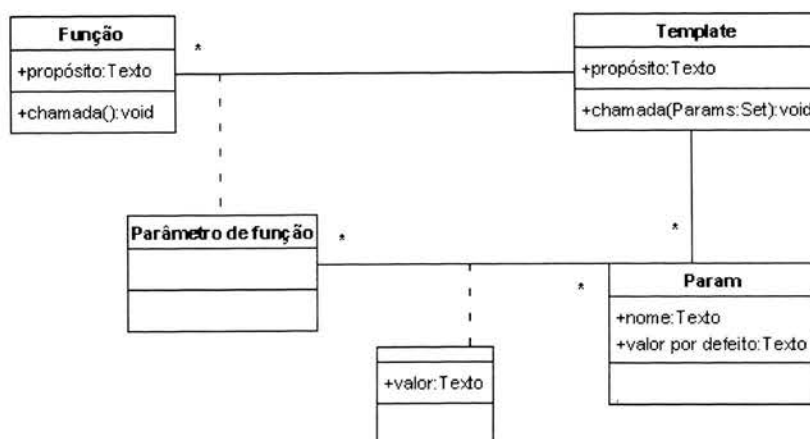


Figura 3.2: Repositório de funções e templates

A figura 3.2 ilustra o princípio de funcionamento do repositório de funções e templates. Cada função pode ser chamada por um indicador para efectuar uma tarefa. O tipo de tarefa é identificado pelo propósito da função. Funções de cálculo, heurísticas de estado, desenho de gráficos são alguns exemplos de propósitos.

A forma como a função está implementada fica escondida do indicador. Uma das formas de implementar a função consiste em derivá-la de um template que é no essencial uma função que pode ser adaptada mediante a especificação de um conjunto de parâmetros. Para derivar uma função de um template devemos apenas identificar o template e definir valores para os seus parâmetros. Exemplificando: temos um template para uma heurística de estado de um semáforo. A função contida neste template recebe um valor e compara-o com dois valores de transição. Da comparação resulta um estado, verde, amarelo, ou vermelho. Os dois valores de transição são os parâmetros de adaptação do template. Posso agora implementar várias

funções a partir do template fazendo variar o valor das transições. Esta aproximação permite definir as funções de forma fácil usando um formulário em HTML.

3.6 Gráficos

Devem-se apresentar os dados dos indicadores em forma gráfica sempre que for possível. Os gráficos são na maioria das vezes mais intuitivos que a informação textual e permitem uma apresentação dos dados mais compacta. A cor também é um instrumento útil para destacar partes importantes. Por fim, e não menos importante, fica mais bonito.

O tipo de gráfico e a informação necessária para desenhar cada gráfico variam de indicador para indicador. Devemos permitir barras, bolos, dispersão ou outras. Pode haver alguma tipificação, com alguns gráficos mais comuns e mais simples a serem desenhados pelo pacote responsável pela apresentação. No caso mais geral, o indicador pode chamar uma função própria para desenhar o indicador, definido um mecanismo de passagem dos dados do indicador às respectivas funções. As funções devem ser registadas no repositório de funções descrito na secção 3.5.

3.7 A consulta do histórico

O histórico do valor dos indicadores pode ser guardado. Há duas vantagens: como já foi dito, os indicadores são instrumentos de decisão; o conhecimento da evolução dos valores dos indicadores em períodos passados, associado ao registo das acções correctivas empreendidas permitem fazer uma análise do impacto dessas acções e podem ajudar a melhorar os processos.

O registo histórico implica a etiquetagem temporal dos valores calculados para os indicadores e a sua colocação num repositório. Podemos associar aos valores dos indicadores outro tipo de etiquetas, tais como anotações de utilizadores ou de programas que venham a ser criados para analisar os resultados dos indicadores.

O registo no repositório obriga a uma reflexão sobre a relevância do armazenamento dos resultados. Devemos armazenar os resultados de todos os cálculos efectuados? Ou isto é demais, e é preciso cortar no armazenamento de valores supérfluos? Vamos introduzir um conceito: *período mínimo de actualização do histórico*. Queremos impedir que dois resultados colocados no histórico estejam separados no tempo por um intervalo superior a este valor. Implementado na sua forma mais simples: se o instante de cálculo não obedecer à condição definida, não colocar no repositório. Há um inconveniente nesta aproximação: imaginem um responsável de dados consultando o seu painel. Pede um refrescamento, vê o resultado mais recente e quando volta a consultar o indicador vê novamente o valor anterior ao refrescamento. É confuso, no mínimo.

Uma alternativa um pouco mais complexa consiste em não ser tão estrito: o *período mínimo de actualização do histórico* (PMAH) passa a representar a diferença para o penúltimo resultado. Se a diferença entre o resultado fresco e o penúltimo resultado for inferior ao PMAH, o novo resultado substituirá o último. Se for superior, será simplesmente colocado no repositório. Não garante que não vou ter dois resultados muito próximos no tempo, mas garante um intervalo médio entre resultados igual ou superior a PMAH/2.

Na consulta do histórico devem ser apresentados, sempre que tal for razoável, gráficos com a evolução do valor do indicador, complementados com informação sobre as transições de estado ou outras etiquetas relevantes.

3.8 Relevância do indicador

Há indicadores que só fazem sentido durante um período curto. Por exemplo, os indicadores sobre o preenchimento das fichas de disciplina só fazem sentido no início de cada semestre. Podemos definir prazos em que o indicador faz sentido na definição do indicador, se tal for conhecido a priori. Senão, podemos permitir a inicialização e terminação manual do indicador. Uma outra possibilidade consiste em recolher a informação do registo de processos, que deverá conter informações sobre prazos de onde poderemos inferir sobre a relevância do indicador.

Digamos que o prazo para o preenchimento das fichas de disciplina é de $D-30$ dias a D , sendo D a data do início do semestre. Faz sentido começar a calcular o valor do indicador na data $D-30$ dias. Mas quando é que faz sentido parar? É possível chegar ao dia D com fichas ainda por preencher, e por isso é conveniente continuar a monitorização para lá desse dia. Uma boa regra é “terminar quando todas as fichas de disciplina estiverem preenchidas”.

Não é difícil imaginar regras ainda mais complexas. Uma forma relativamente simples de implementar tais regras é usando funções de pertença. De cada vez que um cálculo é lançado, faz-se uma verificação prévia da relevância. Se o cálculo for considerado irrelevante não será feito.

3.9 Explicação

Uma parte importante do sistema é a da explicação, humanamente inteligível, dos problemas ocorridos, no sentido de identificar as causas que levam o indicador a relatar uma falha.

Em simultâneo à implementação deste módulo está a decorrer a implementação de um módulo de registo dos processos. Uma vez pronto este módulo servirá como fonte de muita da informação apresentada na explicação. A implementação completa das explicações deve ser retardada um pouco, pelo menos até estar bem definida a interface do módulo de processos.

A explicação pode:

- explicar qual a origem do problema;
- explicar como se chegou à conclusão de que havia um problema;
- sugerir soluções para o problema;
- links úteis;
- indicar quais as consequências da existência ou da persistência do problema;
- indicar actividades associadas ou dependentes que também poderão ser afectadas pelo problema;
- se houver avisos associados, indicar o seu despacho e identificar todos os destinatários.

3.10 Organizando os indicadores

A consulta do painel de controlo deve ser estruturada de forma hierárquica semelhante, por exemplo, a um sistema de ficheiros, em que as directorias contêm dentro de si outras directorias numa estrutura em árvore. Nesta secção vamos reflectir sobre a forma de organizar

os indicadores. A forma mais conveniente de obter esta organização hierárquica consiste em mascarar de indicadores as nossas estruturas de organização.

O agrupamento de indicadores é feito com indicadores compostos. Um indicador composto é constituído por vários outros indicadores, e o seu estado determinado pelos estados dos seus componentes.

Um exemplo é o de um indicador de um subsistema ou módulo, cujo estado pode ser definido, por exemplo, como sendo o mais grave dos estados dos indicadores que integram esse subsistema. Um utilizador interpreta o estado “verde” no indicador de um subsistema como uma indicação de que não há problemas naquele subsistema, e um vermelho como uma indicação de que há problemas e de que deve navegar dentro do indicador para obter mais detalhes. O indicador composto herda os parâmetros dos seus componentes, garantindo assim que nenhum parâmetro fica em falta.

Esta não é a única forma de organizar. Tomemos novamente como exemplo o indicador sobre o preenchimento de fichas de disciplina. Podemos identificar uma hierarquia neste indicador. Eu posso querer visualizar os dados relativos a toda a faculdade, os dados de um determinado curso, ou posso querer saber se uma dada cadeira tem ou não a sua ficha de disciplina completa. Do ponto de vista de um responsável de dados em consulta faz algum sentido permitir a navegação dentro do indicador, como se o indicador fosse composto por outros indicadores.

A descida na hierarquia faz-se pelo uso de parâmetros que identificam os ramos que pretendemos visualizar. Os níveis mais altos da hierarquia são agregações dos valores dos níveis mais baixos. Se for definida a hierarquia *ano lectivo > período lectivo > curso*, ao entrar pode-se consultar os dados para todos os anos lectivos e uma lista de *componentes*, um por cada valor possível do ano lectivo. Escolhendo um ano lectivo, podemos consultar os dados totais para esse ano, e uma lista de *componentes*, uma por período, e assim por diante até às folhas.

Pode dar-se o caso de os valores não serem aditivos, como percentagens, e ser necessário guardar alguma informação extra para obter valores aditivos. No caso das percentagens, é preciso anexar ao resultado o número total de registos, que multiplicado pelo resultado dará o número de registos com problemas.

A agregação pode ser feita a partir dos dados constantes no repositório. Como consequência podem ser agregados dados calculados em instantes diferentes.

Os indicadores agregados têm uma fórmula de agregação e uma fórmula de estado, sendo inclusivamente possível ter várias agregações diferentes dos mesmos dados. Por exemplo, um indicador que agrega somando as taxas de preenchimento de fichas de disciplina que produz um resultado geral a nível de faculdade com o intuito de produzir informação estatística pode coexistir lado a lado com um que apenas determina o resultado mínimo (ou máximo) no intuito de determinar se há ou não problemas locais. Estas diferenças semânticas na interpretação do indicador devem ser explícitas. Pode ainda ser criado um mecanismo que permita associar, de alguma forma, estas diferentes agregações dos mesmos dados, clarificando a natureza da diferença e tornando-a mais intuitiva.

A agregação é sempre feita da mesma forma, independentemente do nível. Com alguma penalização na performance podemos implementar indicadores agregados que usem uma fórmula de agregação diferente em cada nível. Ou seja, os dados são agregados no nível mais

baixo. O nível seguinte agrega os dados da agregação anterior usando outra fórmula, e por aí acima até à raiz.

Há ainda um tipo de indicadores a mencionar: indicador de ramificação. Este indicador é útil para agregar indicadores compostos. O indicador herda os parâmetros dos seus componentes, mas um desses parâmetros é designado *parâmetro de ramificação* e deixa de ser necessário. Se o indicador for aberto sem este parâmetro, mostrará uma lista de *componentes*, um por cada valor possível do parâmetro.

3.11 A consulta

Pode haver vários painéis de controlo. Cada um destes painéis deve estar orientado a um responsável de dados e deve agrupar indicadores relevantes para esse responsável. Entendemos por responsável de dados alguém que tem por missão vigiar os dados associados a um conjunto de actividades. Exemplos de responsáveis de dados são:

- os administradores do SIGARRA, responsáveis por todos os dados;
- os directores de departamento, responsáveis pelos dados gerados nas actividades do seu departamento;
- os directores de curso, responsáveis pelos dados gerados nas actividades do seu curso;
- o director da faculdade, responsável por todos os dados da sua faculdade.

Alguns dos dados mostrados podem ser sensíveis ou estratégicos e devem ser mantidos longe de olhares indiscretos. O acesso deve ser reservado.

A entrada num painel tipo deve consistir numa lista de indicadores gráficos ou textuais apresentados por ordem alfabética. Cada indicador deve mostrar claramente o seu título. Abaixo do título surge a informação daquele indicador. Esta informação pode variar de indicador para indicador, o que exige um bom grau de configurabilidade. Abaixo da informação do indicador surge a sua lista de componentes, na forma de janelas miniaturizadas.

Devem ser definidas duas páginas: uma para a miniatura, que é apresentada nas janelinhas das listas de componentes dos indicadores compostos ou agregados, e uma para a página principal do indicador.

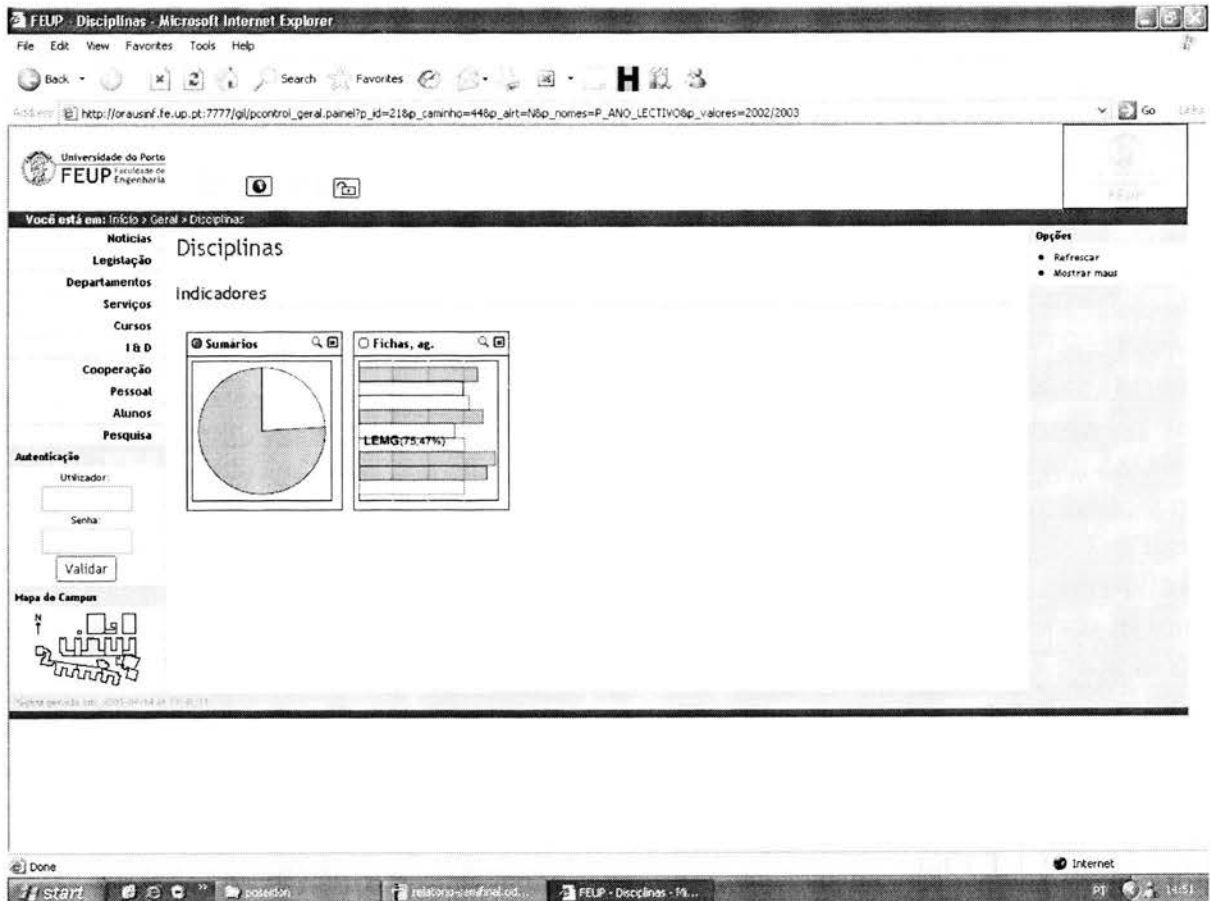


Figura 3.3: Um screenshot

A miniatura pode ter vários tipos de dados, como listagens ou gráficos. Podemos clicar no nome do indicador, que se encontra sobre a miniatura, e ser levados para a página desse indicador onde podemos consultar os dados do indicador de forma mais detalhada. A apresentação do indicador tem elementos obrigatórios que surgem sempre. O título e a lista de componentes são sempre apresentados. Todos os outros elementos devem ser declarados.

A apresentação do indicador deve ser acompanhada sempre que possível das vias de decisão respectivas. Isto pode querer dizer uma listagem de links para pontos do sistema onde se pode desencadear a correcção dos problemas. A causa do problema deve também ser explicada de forma clara, e quando necessário, devem ser explicadas as consequências que se podem esperar da existência e da persistência do problema. A listagem de links pode ter origem no registo de processos, tal como muita da informação da explicação.

Um princípio que deve ser tomado em conta na especificação da visualização do indicador é o da maximização da quantidade de informação por unidade de área. As coisas devem ser mantidas tão pequenas quanto possível e deve ser mostrada apenas a informação relevante.

Em cada indicador deve existir um botão de refrescamento, permitindo recalcular o indicador.

3.12 Avisos

Depois de calculados os valores e os estados dos indicadores o indicador pode ficar num estado indesejável. Neste caso é, às vezes, adequado notificar os donos dos dados. Os avisos podem ser, por exemplo, mensagens de e-mail enviadas para o responsável de dados contendo

a descrição dos problemas encontrados no sistema. A mensagem deve conter a descrição clara do problema e ligações para a apresentação do indicador, para as páginas de inserção ou alteração dos dados, ou para páginas que podem permitir ao responsável a resolução dos problemas indicados.

Se o aviso alertar mais do que um responsável, todos devem ser claramente identificados na mensagem, mesmo que a níveis diferentes. Por exemplo, ainda nas fichas de disciplina, podemos alertar o docente responsável e o director de curso. Ambos devem ser notificados de que o outro recebeu o mesmo aviso.

A determinação dos responsáveis a alertar pode ser feita dinamicamente, uma vez definidos e registados os processos e seus intervenientes na base de dados. Neste caso, os alarmes deverão ser dirigidos a papéis e não a pessoas concretas. O sistema de gestão dos processos deverá ser capaz de determinar a pessoa concreta a partir do papel desempenhado. A definição do papel terá de ser plenamente especificada. Exemplos: director do curso XPTO; regente da disciplina PTOX do curso XPTO.

No caso de o problema detectado estar associado a uma tarefa da qual dependem outras, o aviso pode conter mensagens avisando da possibilidade das tarefas dependentes poderem ser afectadas por este problema, ligações para as páginas relevantes para a resolução de problemas que venham a surgir nessas tarefas dependentes em consequência do actual problema e ligações para indicadores relacionados.

Os avisos podem ser emitidos manualmente, disparados através de um clique efectuado pelo responsável de dados num comando do painel, ou automaticamente usando um mecanismo apropriado. A emissão de um aviso deve basear-se no estado dos indicadores.

4 Descrição do módulo

4.1 Consultando o painel

Existe um conjunto de painéis de controlo de dados, cada um deles orientado a um determinado responsável de dados. O acesso a cada painel é permitido apenas aos responsáveis de dados respectivos e deve ser feito usando a infraestrutura de validação de utilizadores disponibilizada pelo GESSI. Após um cabeçalho de identificação do painel deve-se incluir uma lista com os vários indicadores em que se organiza a monitorização do sistema. Cada componente é mostrado na forma de uma pequena janela gráfica, tendo no seu cabeçalho o estado indicado por um semáforo ou outro código gráfico. O estado deve ter associado um texto breve explicando o seu significado. Por exemplo “Há problemas num ou mais módulos”, “A taxa de preenchimento de sumários é inferior à meta definida”, “Tudo OK”, etc. Após o estado surge o título do componente, ou parte dele. Se o título for cortado deve ser mostrado em balão o título completo. Abaixo do título e do ícone de estado há uma pequena janela onde deve ser resumida a informação sobre o indicador. É a vista *mini* do indicador.

Daqui o utilizador pode descer na árvore hierárquica, sendo que os dados dos indicadores seguem o mesmo esquema, aumentando o número de indicadores visualizados à medida que se restringe o âmbito do sistema e tentando um equilíbrio entre trazer os indicadores mais relevantes e mais consultados para os níveis superiores e manter uma estrutura lógica na respectiva arrumação. A possibilidade de duplicação da arrumação dos indicadores, com um indicador a poder ser incluído em vários sítios do mesmo painel pode ajudar a atingir este objectivo (ou não, consoante caso).

Os dados mostrados na página de cada indicador são definidos na vista *maxi* do mesmo. O título do indicador e os seus componentes são mostrados por defeito. Podem também ser mostrados dados adicionais do indicador, gráficos, listagens, etc. A informação a mostrar em cada indicador é totalmente configurável. No topo surge o título, abaixo deste toda a informação opcional e em baixo a lista de componentes do indicador.

O painel deve usar a infraestrutura disponibilizada pelo GESSI, mantendo assim o *look and feel* do SIGARRA. No cabeçalho da página, a barra onde deve conter o caminho até ao ponto actual na hierarquia, fazendo com que a subida na hierarquia seja tão simples quanto a descida. Comandos adicionais como *Refrescar* ou *Despachar aviso* devem ser colocados na barra da direita, à maneira do SIGARRA.

As janelas miniatura são encaixadas dinamicamente, adaptando a sua disposição às dimensões da janela do browser. Por vezes esta janela miniatura é pequena demais para uma apresentação adequada do seu conteúdo. Para evitar este problema, a largura da janela pode variar.

Maxi

A parte configurável da informação mostrada na página do indicador corresponde à vista *maxi* do indicador. O indicador pode incluir na página *maxi* vários componentes:

- Folha informativa, que é uma tabela com:
 - descrição;

- importância;
- natureza;
- data do último cálculo;
- estado;
- lista de parâmetros;
- explicação textual;
- a explicação, que é um texto humanamente inteligível explicando o que foi medido, como, qual o valor apurado e qual é a interpretação sugerida para esse valor; devem ser apresentadas as possíveis consequências do problema;
- uma lista de links (caso haja) onde o problema poderá ser resolvido; os links poderão ter de ser parametrizados;
- situação dos avisos, incluindo data de despacho e identificação dos receptores;
- lista de indicadores relacionados quer por correrem sobre os mesmos dados, quer por pertencerem ao mesmo procedimento, acessíveis no painel que está a ser visualizado;
- gráficos variados.

Mini

Na vista *mini* podemos ver as mesmas coisas que na vista *maxi*. Porém deve ser mantido em mente que a vista *mini* é apresentada numa janelinha pequenina. Deve ser mostrada apenas a informação essencial, e sempre que possível dimensionada de forma a caber na janelinha.

Organização

A organização é conseguida através do uso de indicadores compostos. A apresentação de um indicador composto consiste num painel acrescida de alguma informação adicional do indicador, como nome ou estado. Segue-se a lista de componentes, apresentados na sua forma *mini*.

Outra forma de organização é obtenível com indicadores agregados. Os indicadores agregados permitem observar os vários valores respeitantes às diferentes combinações de parâmetros de um indicador. O indicador agregado tem um estado e um resultado, também flexíveis. A navegação é feita segundo uma hierarquia de parâmetros. No nível mais elevado são mostrados os dados gerais. Abaixo mostra-se a lista de componentes, estruturada da forma já descrita, que correspondem às escolhas que o utilizador pode fazer. Abrindo um componente, o utilizador vê os dados para essa escolha, e por aí abaixo até às folhas da hierarquia.

Apenas informação relevante

A informação a mostrar em cada indicador é configurável. Nos nós internos da hierarquia deve ser mostrada a informação relevante, de forma a mostrar o máximo de informação no mínimo espaço possível. Nas folhas podemos dar-nos ao luxo de apresentar mais informação.

Lista de registos problemáticos

Deve conter a listagem de registos problemáticos, eventualmente com um link para a página respectiva. Deve ser apresentada na forma mais conveniente (tabela, lista, ...) e com toda a informação relevante.

Componentes gráficos

Os indicadores podem ter gráficos associados, que são mostrados na sua consulta. A responsabilidade do desenho dos gráficos pode ser do indicador, cabendo ao painel apenas a sua apresentação, ou do painel, cabendo ao indicador apenas o fornecimento dos dados.

- em gráficos comuns, o painel deve desenhar os gráficos a partir dos dados fornecidos pelo indicador;
- em gráficos específicos, o indicador implementará uma função de desenho dos gráficos.

Os gráficos podem ser usados para apresentação e enquadramento dos resultados. Categorias de gráficos possíveis são:

- barra vertical empilhada com valor actual, valor ideal, limite mínimo, limite máximo;
- barra horizontal empilhada com valor actual, valor ideal, limite mínimo, limite máximo;
- manómetro com ponteiro no valor actual, e marcas no valor ideal, limite mínimo, limite máximo;
- distribuição em gráfico de barras;
- queijo;
- prazo, com data inferior, de alerta e superior e dia corrente;
- temporal, linha com a evolução do indicador e o intervalo de agregação (horário, diário, semanal, mensal, anual) e as datas limite;
- acumulado, idem mas somando;
- sim ou não;

Em todos os gráficos usar o vermelho para as zonas de alerta, o verde para as de normalidade e o amarelo para as perigosas; outras cores para o resto.

4.2 Definindo um template

O mecanismo de templates é uma tentativa de instituir o reaproveitamento de código de forma simples.

O utilizador começa por implementar o template num procedimento PL/SQL na base de dados. De seguida deve registar esse procedimento. O registo implica a atribuição de um nome, de uma descrição e da escolha de um propósito, que poderá ser: fórmula, heurística de estado, etc. Finalmente devem ser registados os parâmetros de adaptação do template. Cada parâmetro deve ter um nome, tipo de dados PL/SQL, descrição e valor por defeito. Após este registo, o template fica pronto para ser usado no registo de funções.

4.3 Registando funções

A especificidade de cada indicador implica o uso de procedimentos PL/SQL próprios de cada indicador. Estes procedimentos podem ser implementados dentro do sistema de gestão de base de dados ou não. No primeiro caso, o administrador deve registar no repositório o nome do procedimento ou função PL/SQL implementado. No segundo, deve escrever explicitamente o código de um procedimento PL/SQL anónimo ou derivar a função de um template.

Para derivar uma função de um template o administrador começa escolhendo o propósito. Depois escolhe um template de uma lista de templates com o propósito pretendido. Por fim deve atribuir valores aos parâmetros de adaptação do template.

Em todos os casos é preciso dar um título à função e definir o propósito.

4.4 Definindo um indicador

Um indicador elementar deve ser definido da seguinte forma:

- devem ser definidos os seus dados:
 - título curto;
 - descrição um pouco mais extensa;
 - importância, que pode ser usada na organização da apresentação dos indicadores ou nos cálculos de *criticidade*;
 - natureza do indicador que representa o tipo de incorrecção a que o indicador é devotado; valores possíveis são: *completude*; *correccção*; *coerência*; *validação*;
- informação de auditoria:
 - criador;
 - data de criação/alteração;
 - observações;
- deve ser definido o seu período de relevância;
- devem ser escolhidos os parâmetros necessários;
- deve ser definida a fórmula de cálculo do indicador;
 - baseada em:
 - procedimento PL/SQL;
 - query;
 - ordem dos parâmetros;
 - informação adicional;
- deve ser definida a sua heurística de estado, a categoria gráfica do estado e a explicação de cada um dos estados;
- deve ser definida a explicação do indicador;
- devem ser definidos os dados relativos ao cálculo do indicador e respectiva forma de armazenamento;
- o indicador deve estar associado a um procedimento; o módulo de procedimentos ainda está em fase de desenvolvimento, mas mais tarde será possível obter muita informação útil por esta via, como prazos ou dependências;
- vista maxi;
- vista mini;
- explicação;
- lista de registos problemáticos:
 - título;
 - sqlx.

Período de relevância

O período de relevância é flexível. Pode ser definido a partir de um template ou pode ser criada uma função própria para o efeito. Se for definido a partir de um template, o utilizador deve escolher um de um conjunto de templates disponíveis, e depois deve especificar os valores dos parâmetros necessários. Se for usada uma função apropriada (deve ser raro), deve ser indicado o nome da função.

Fórmula

A fórmula de cálculo é flexível. Pode ser definida a partir de um template (deve ser raro para indicadores elementares) ou pode ser criada uma função própria para o efeito. Se for definido a partir de um template, o utilizador deve escolher um de um conjunto de templates disponíveis, e depois deve especificar os valores dos parâmetros necessários. Se for usada uma função apropriada, deve ser indicado o nome da função.

Estado

A formula de estado de um indicador elementar vai apenas ler o resultado e usar uma heurística muito simples para dizer se o resultado é bom ou mau. A heurística pode depender de outros dados que não o resultado (como o tempo), mas a referência a estes dados deve estar contida no código da heurística. Pode ser definida a partir de um template ou pode ser criada uma função própria para o efeito. Se for definida a partir de um template, o utilizador deve escolher um de um conjunto de templates disponíveis, e depois deve especificar os valores dos parâmetros necessários. Se for usada uma função apropriada (deve ser raro), deve ser indicado o nome da função. Associada à definição da heurística devem estar a definição da categoria gráfica e lógica do estado (nº de estados, conjunto de ícones) e respectivas explicações textuais.

4.5 Indicadores compostos

Os indicadores podem ser agrupados através de indicadores compostos. O primeiro passo para a organização consiste em definir um indicador composto. O indicador composto deve ter:

- dados:
 - título curto;
 - descrição um pouco mais extensa;
 - importância, que pode ser usada na organização da apresentação dos indicadores ou nos cálculos de *criticidade*;
- informação de auditoria:
 - criador;
 - data de criação/alteração;
 - observações;
- heurística de estado;
- significado de cada estado;
- composição usando um interface web:
 - o interface permite adicionar ou remover componentes do composto;
- página maxi;
- página mini.

4.6 Indicadores agregados

O indicador agregado é registado com a seguinte informação:

- dados:
 - título curto;
 - descrição um pouco mais extensa;
 - importância, que pode ser usada na organização da apresentação dos indicadores ou nos cálculos de *criticidade*;

Em todos os casos é preciso dar um título à função e definir o propósito.

4.4 Definindo um indicador

Um indicador elementar deve ser definido da seguinte forma:

- devem ser definidos os seus dados:
 - título curto;
 - descrição um pouco mais extensa;
 - importância, que pode ser usada na organização da apresentação dos indicadores ou nos cálculos de *críticidade*;
 - natureza do indicador que representa o tipo de incorrecção a que o indicador é devotado; valores possíveis são: *completude*; *correccção*; *coerência*; *validação*;
- informação de auditoria:
 - criador;
 - data de criação/alteração;
 - observações;
- deve ser definido o seu período de relevância;
- devem ser escolhidos os parâmetros necessários;
- deve ser definida a fórmula de cálculo do indicador;
 - baseada em:
 - procedimento PL/SQL;
 - query;
 - ordem dos parâmetros;
 - informação adicional;
- deve ser definida a sua heurística de estado, a categoria gráfica do estado e a explicação de cada um dos estados;
- deve ser definida a explicação do indicador;
- devem ser definidos os dados relativos ao cálculo do indicador e respectiva forma de armazenamento;
- o indicador deve estar associado a um procedimento; o módulo de procedimentos ainda está em fase de desenvolvimento, mas mais tarde será possível obter muita informação útil por esta via, como prazos ou dependências;
- vista maxi;
- vista mini;
- explicação;
- lista de registos problemáticos:
 - título;
 - sqlx.

Período de relevância

O período de relevância é flexível. Pode ser definido a partir de um template ou pode ser criada uma função própria para o efeito. Se for definido a partir de um template, o utilizador deve escolher um de um conjunto de templates disponíveis, e depois deve especificar os valores dos parâmetros necessários. Se for usada uma função apropriada (deve ser raro), deve ser indicado o nome da função.

Fórmula

A fórmula de cálculo é flexível. Pode ser definida a partir de um template (deve ser raro para indicadores elementares) ou pode ser criada uma função própria para o efeito. Se for definido a partir de um template, o utilizador deve escolher um de um conjunto de templates disponíveis, e depois deve especificar os valores dos parâmetros necessários. Se for usada uma função apropriada, deve ser indicado o nome da função.

Estado

A formula de estado de um indicador elementar vai apenas ler o resultado e usar uma heurística muito simples para dizer se o resultado é bom ou mau. A heurística pode depender de outros dados que não o resultado (como o tempo), mas a referência a estes dados deve estar contida no código da heurística. Pode ser definida a partir de um template ou pode ser criada uma função própria para o efeito. Se for definida a partir de um template, o utilizador deve escolher um de um conjunto de templates disponíveis, e depois deve especificar os valores dos parâmetros necessários. Se for usada uma função apropriada (deve ser raro), deve ser indicado o nome da função. Associada à definição da heurística devem estar a definição da categoria gráfica e lógica do estado (nº de estados, conjunto de ícones) e respectivas explicações textuais.

4.5 Indicadores compostos

Os indicadores podem ser agrupados através de indicadores compostos. O primeiro passo para a organização consiste em definir um indicador composto. O indicador composto deve ter:

- dados:
 - título curto;
 - descrição um pouco mais extensa;
 - importância, que pode ser usada na organização da apresentação dos indicadores ou nos cálculos de *criticidade*;
- informação de auditoria:
 - criador;
 - data de criação/alteração;
 - observações;
- heurística de estado;
- significado de cada estado;
- composição usando um interface web:
 - o interface permite adicionar ou remover componentes do composto;
- página maxi;
- página mini.

4.6 Indicadores agregados

O indicador agregado é registado com a seguinte informação:

- dados:
 - título curto;
 - descrição um pouco mais extensa;
 - importância, que pode ser usada na organização da apresentação dos indicadores ou nos cálculos de *criticidade*;

- indicador que é agregado;
- informação de auditoria:
 - criador;
 - data de criação/alteração;
 - observações;
- heurística de estado;
- fórmula de agregação;
- significado de cada estado;
- hierarquia de parâmetros:
 - ordem hierárquica;
 - títulos de cada nível;
- página maxi;
- página mini.

4.7 Painéis

O painel é uma forma de controlar o acesso aos indicadores por parte dos utilizadores e está encavalitado em cima de um indicador composto que lhe serve de raiz da hierarquia. O painel tem também informação sobre quem tem autorização para o consultar e sobre os parâmetros necessários para abrir um painel. Assim, para definir um painel deve-se:

- definir o indicador composto que serve de raiz à estrutura;
- escolher os grupos de validação que podem consultar o painel;
- escolher os parâmetros do painel:
 - definir valores por defeito;
 - definir se o valor do parâmetro é fixo ou se pode ser alterado.

4.8 Cálculo

Há dois aspectos a especificar para o cálculo:

- o cálculo periódico;
- o cálculo manual.

Após a definição da informação de periodicidade, que consiste em:

- período (em dias); deve ser usada fracção para períodos horários;
- data/hora do próximo cálculo (doravante actualizada);

o administrador pode inicializar o cálculo periódico. O indicador é calculado pela primeira vez na data e hora definidas para o próximo cálculo, e daí em diante de forma periódica.

Existe ainda uma listagem de indicadores em situação de cálculo periódico para facilitar a administração destes cálculos.

O cálculo manual é feito quando um utilizador pede um refrescamento ou quando é despoletado pela consulta de um indicador. O refrescamento pode ser directo ou diferido. Se o cálculo for leve, o refrescamento pode ser directo, e o resultado do refrescamento é (na óptica do utilizador) a apresentação dos novos resultados. Se o cálculo for pesado, deve surgir uma mensagem com uma estimativa, do género “O cálculo do indicador deve demorar x tempo. Volte a aceder à página do indicador mais tarde, por favor.” O indicador pode também ser definido de forma a ser calculado em cada consulta.

4.9 Histórico

O indicador pode ou não ter relevância histórica. Se tiver, todos os cálculos que obedecem à condição descrita na secção 3.7 são registados. Senão, apenas é registado o último cálculo.

O registo histórico pode permitir a análise da evolução do indicador nos casos em que isso não for possível fazer directamente a partir dos dados.

4.10 Avisos

Os avisos são definidos sobre os resultados ou estados dos indicadores. Quando estes valores atingirem um estado crítico, deve ser desencadeado um procedimento de aviso cujo comportamento perante a ocorrência e a persistência de problemas poderá ser configurado. O aviso consiste no envio de uma mensagem por e-mail ou por outro meio de uma notificação dirigida a um ou mais portadores de um papel.

O aviso pode ser despachado manualmente ou automaticamente. No primeiro caso, no momento da consulta do indicador, é apresentada ao responsável de dados uma mensagem e um botão, e se o responsável de dados assim o entender pode enviar a mensagem associada aos destinatários.

O registo de um aviso consiste na definição:

- do indicador associado (implicitamente);
- dos papéis a alertar;
- do comportamento a tomar:
 - manual;
 - automático:
 - avisar uma só vez;
 - avisar duas vezes;
 - avisar até o problema estar resolvido, com mensagens regularmente espaçadas;
 - avisar até o problema estar resolvido, com mensagens cada vez mais espaçadas entre si;
- das mensagens.

4.11 Actores

A consulta de um painel e actividades relacionadas, como refrescamento ou despacho manual de um aviso são actividades de um responsável de dados. Todas as restantes são da competência dos administradores do SIGARRA.

5 Trabalho

5.1 Método de trabalho usado

O trabalho partiu de um conjunto de ideias e requisitos algo vagos, e foi decidido tomar uma aproximação exploratória e iterativa. Após um período inicial de análise, partiu-se para a implementação de um protótipo que permitisse por um lado clarificar as ideias quanto à natureza do painel de controlo e por outro identificar as dificuldades técnicas que o problema levantava. O protótipo foi implementado iterativamente, uma funcionalidade de cada vez, até um certo nível. Chegado o ponto em que as novas funcionalidades podiam ser mostradas, o protótipo foi analisado, e os pedidos de alterações efectuados, numa perspectiva evolutiva.

Do posto de vista social, na USINF as pessoas trabalham sozinhas, apoiando-se sempre que necessário nos colegas. Este foi por isso um trabalho essencialmente individual e com necessidades reduzidas de comunicação. Não houve necessidade de especificações muito detalhadas. O processo de análise foi muito evolutivo e informal. Após uma análise inicial de cerca de quatro semanas, passou-se à implementação. A implementação de cada nova funcionalidade foi precedida de uma análise mais aprofundada da mesma e da sua integração com as restantes. Tipicamente a implementação de uma nova funcionalidade implicava duas coisas: novo código e alteração do código para acomodar a nova funcionalidade.

A análise assentou muito na reflexão sobre as ideias envolvidas, e na construção de abstrações capazes de sustentar essas ideias. Houve, no fundo, uma especificação de ideias e o desenvolvimento fez-se a partir daí, e não uma especificação baseada na acção como, por exemplo, usando casos de utilização. Os diagramas de classes e pontualmente de actividades foram as ferramentas de que o estagiário mais se socorreu. Cada funcionalidade foi anotada na forma de um pequeno parágrafo, que serviu de guião na implementação das mesmas.

Houve algum esforço no sentido da documentação do código *inline*, de forma a facilitar a sua manutenção num futuro próximo. Fora do código, o esforço de documentação dirigiu-se apenas à elaboração deste relatório, que foi evoluindo desde o relatório de análise inicial, e das ajudas, necessárias ao bom uso do sistema pelos seus utilizadores.

5.2 Cronologia

O trabalho foi iniciado a 9 de Setembro. Antes de começar o trabalho tinha já conversado com o Eng. Manuel Machado e o Prof. Gabriel David sobre os objectivos do trabalho e tinha elaborado já um modelo de classes do domínio, ainda muito cru para tentar perceber o problema. Comecei por analisar a documentação relativa ao registo de processos produzida no estágio da Raquel Amaral, e um pequeno relatório de especificação do módulo de monitorização da qualidade de dados que continha a visão do sistema e uma lista preliminar de requisitos. Tendo já essa lista preliminar requisitos prossegui com o trabalho de análise dos mesmos.

A análise guiou-me no sentido da construção de um framework sobre o qual poderia assentar a implementação dos indicadores. O framework define a forma como os indicadores se relacionam entre si, como são calculados, como estão organizados e como são apresentados. A implementação de um indicador poderia então concentrar-se apenas nas suas

particularidades, como dados e lógica, utilizando os serviços disponibilizados pelo framework para tudo o resto.

Ainda nesta fase li os manuais de PL/SQL e Oracle SQL, e livros sobre XML, XP e padrões de software, procurando conhecimento e inspiração a nível tecnológico e de organização do projecto. A análise prosseguiu, misturada com algumas experiências em PL/SQL até ao final de Setembro. Nesta altura, um documento que designei relatório de análise e arquitectura foi debatido numa reunião com os dois orientadores. Nesta reunião a maioria das ideias foi validada, uma ou outra recusadas e a maioria mereceu pequenos reparos.

No seu essencial, o framework ficou definido, sendo apenas corrigidos alguns aspectos que a implementação revelou não serem adequados. As recusas tinham a ver com o mecanismo de avisos, que eu pretendia que fosse feito ao papel e assente no registo de processos, e discordância quanto à implementação dos prazos de relevância com funções.

Ficou assente que se deveria construir um protótipo para avaliar a viabilidade das ideias e identificar as falhas. Pus de imediato mãos à obra, estimando em cerca de dois meses o tempo necessário para por um protótipo a funcionar.

Decidi começar por implementar o repositório de funções e templates. Pareceu-me simples o suficiente para um iniciado ao PL/SQL. Nesta altura, um dos colegas do CICA apresentou-me o PL/SQL como “a linguagem mais fácil do mundo; vais dominar aquilo em quinze dias.” Tal como esperava, a implementação deste pacote de funcionalidades revelou-me muitas das peculiaridades do PL/SQL. Confesso que não achei a linguagem nada fácil: das linguagens que conheço, pior só mesmo o Visual Basic. Nesta altura comecei a achar que a implementação do framework em PL/SQL ia ser dura.

Passei boa parte do tempo a engendrar truques (aquela coisa que um bom programador nunca deve fazer) para conseguir o que pretendia. Quando o registo de funções me pareceu bem, passei para a implementação dos indicadores. Continuei a engendrar truques, e confesso que por vezes perdi mais tempo a engendrar uma forma de fazer as coisas do que a fazê-las efectivamente. Nesta altura (meio de Novembro) veio-me uma luz: a Oracle também têm objectos na base de dados e é possível utilizar objectos em PL/SQL. Os objectos emprestam ao PL/SQL alguma flexibilidade e potência. Li umas coisas, fiz umas experiências, mas acabei por não levar para a frente. Apesar das vantagens relativamente a outras estruturas, eu já tinha muito código que teria de ser transposto, e o uso dos objectos obrigava a um corte com a estruturação do código em *packages* usual na USINF. Além disso, o editor usado na USINF, o SQL Navigator, não funcionava bem com objectos. Continuei implementando a parametrização, os indicadores agregados e os indicadores compostos. Passei depois para a implementação de alguns exemplos de indicadores relacionados com as fichas de disciplina e os sumários. Entretanto chegou o mês de Dezembro.

Nesta altura tive uma reunião com o Eng. Manuel Machado, que achou que o framework estava a ficar suficientemente maduro para começar a pensar em implementar indicadores. Eu deveria melhorar o aspecto gráfico do módulo, implementar alguns indicadores extra e marcámos uma reunião com o Prof. Gabriel David para avaliar o estado do projecto. Neste sentido, comecei a recolher informação para a implementação dos indicadores *de teste*. Fiz a transposição da aplicação para o GESSI, emprestando-lhe o *Look and Feel* do SIGARRA. Corrigi bugs. E, fui implementando funcionalidades menos urgentes, como a autorização, a barra onde, os avisos, a relevância histórica, a explicação textual e a listagem de registos problemáticos.

A única que originou uma ideia nova foi a da explicação textual. A minha ideia era colocar informação no meio de um texto de forma fácil e intuitiva. Eis como: o administrador define a explicação na forma de um documento XML, sendo que algumas das marcas representam informação que pode ser incluída na explicação em *runtime*; este XML é transformado, através de uma XSLT em PL/SQL que é executado sempre que for necessário mostrar a informação.

A reunião realizou-se finalmente a 21 de Janeiro, e acabou por originar uma considerável marcha atrás. O Prof. Gabriel estava à espera de algo mais gráfico, e criticou algumas funcionalidades. Também achou que havia desperdício de espaço útil com informação pouco importante.

No final da reunião soaram as sirenes de alarme. Algumas contas fizeram-me constatar que, tirando um mês para escrever o relatório, teria menos de 3 semanas para fazer alterações. Fiz contas outra vez e constatei que não era bem assim. Afinal, o mundo não ia acabar com o estágio, e a entrega do relatório podia ser feita mais tarde. De qualquer maneira, as alterações e as novas funcionalidades precisavam de mais tempo do que o que havia disponível. Entrei de novo em fase análise e de planeamento. Era necessário fazer com que o módulo fosse mais gráfico, e que a informação fosse mais condensada. Respondi a estes problemas com um aumento da configurabilidade do painel de controlo. Para cada indicador, deveria ser possível decidir o que iria ou não ser mostrado e onde. A *ferramenta* experimentada na explicação textual pareceu-me uma boa forma de implementar a configurabilidade extra. Faltava ainda a questão dos gráficos. Mostrar ícones gráficos condensados não era tarefa fácil. Após conversar com as pessoas do CICA que mais usavam os pacotes gráficos da Oracle, achei que não me serviam nos casos em que era preciso fazer gráficos pequeninos. Nesta altura, o Luís Ramos sugeriu-me o SVG para o desenho dos gráficos. Tomei a ideia, testei-a e implementei-a. Quanto à forma de inclusão dos gráficos, já tinha o trabalho planeado.

Implementei as alterações propostas, a mais notável das quais ao nível da listagem de registos problemáticos. Resolvi usar SQLX para construir a apresentação dos registos directamente a partir de uma query. Ou seja, a query produz XHTML que pode ser pura e simplesmente colado na página de consulta do indicador.

Validei as novas ideias com o Prof. Gabriel e o Eng. Manuel Machado e implementei-as. Este esforço ocupou a quase totalidade do tempo ainda disponível. Algumas funcionalidades foram atrasadas e outras deixaram de funcionar em pleno com as correcções efectuadas. A sua correcção fica delegada para o período pós-estágio.

Entretanto, também corriji os indicadores para irem de encontro às novas funcionalidades, e implementei uns poucos gráficos para testar o sistema.

É Março e dentro de poucos dias vou entregar o relatório.

5.3 Implementação

Arquitectura lógica

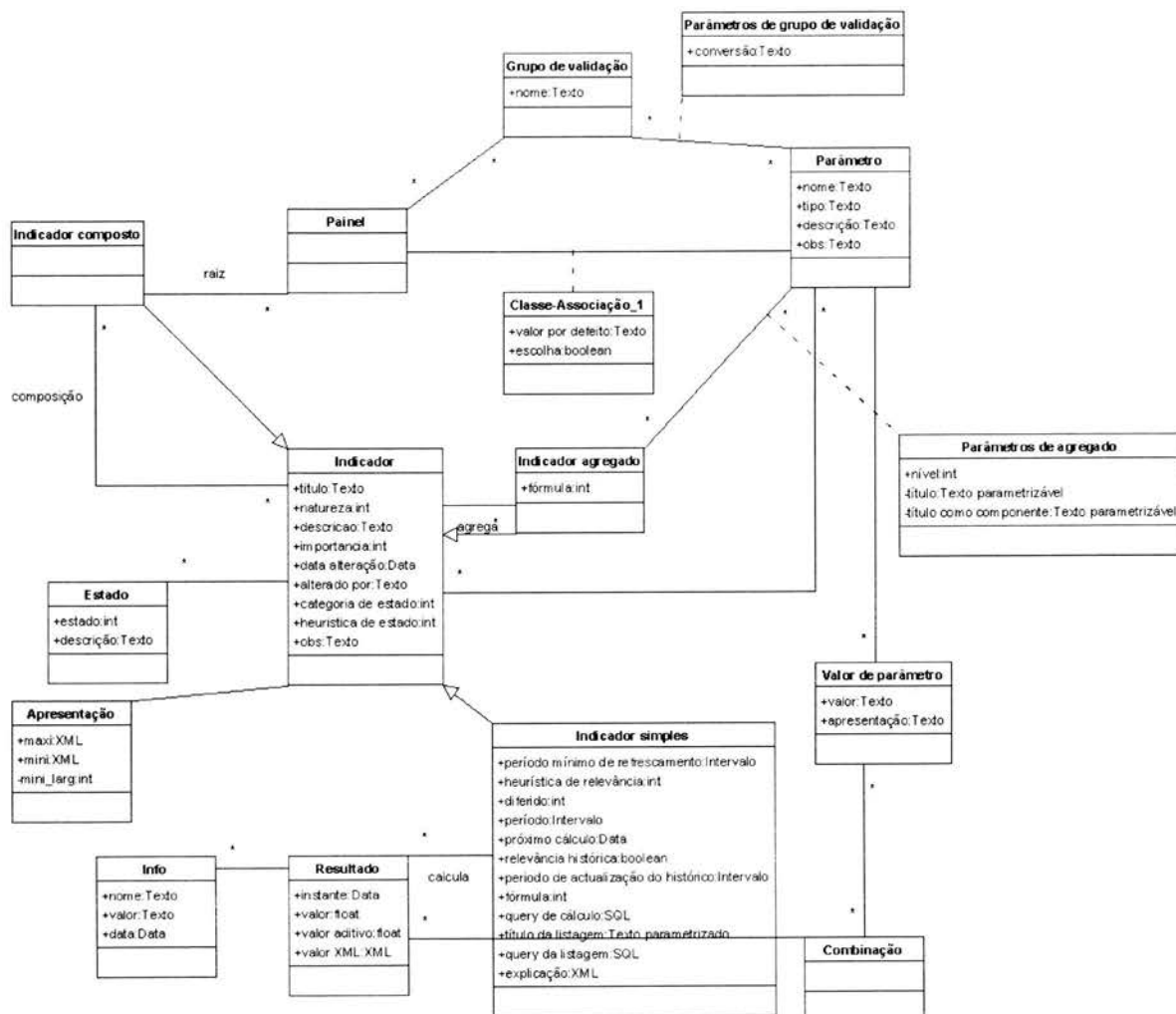


Figura 5.1: Diagrama de classes dos indicadores (excluindo avisos e autorização)

A figura 5.1 mostra o diagrama de classes dos indicadores.

Cada indicador possui um título identificativo curto, uma natureza indicando o tipo do indicador e dos seus resultados, uma descrição breve. A cada indicador atribuímos uma importância, codificada num número de 1 a 5, usada para ordenar os indicadores ou nas heurísticas de estado dos indicadores compostos. A categoria de estados encapsula a informação sobre o número de estados do indicador e a sua criticidade. Envolve também informação sobre a apresentação gráfica do indicador. A heurística de estado interpreta o resultado do cálculo e coloca o indicador num estado apropriado.

Há ainda três campos de auditoria: *data de alteração*, *alterado por* e um campo de observações, *obs*.

Há alguns campos associados à configuração da apresentação dos indicadores. O campo *maxi* configura a apresentação do indicador numa página própria. O campo *mini* configura a apresentação do indicador numa janela pequenina, como componente de um outro indicador. O campo *mini_larg* permite controlar a largura desta janela. Este valor poderia ser remetido para o interior do XML em *mini*, mas há alguns benefícios em tê-lo aqui.

Cada indicador tem um conjunto de estados possíveis. Cada estado possível é identificado por um número e deve conter uma pequena descrição de si próprio.

Os indicadores elementares têm uma grande quantidade de dados associados à configuração dos cálculos e do seu registo. Vamos então descrever a utilidade de cada um dos campos. O *período mínimo de refrescamento* controla a diferença entre dois refrescamentos consecutivos. A ideia é impedir refrescamentos inúteis e repetidos. Se for 0, todos os refrescamentos são permitidos. O campo *diferido* tem a ver com a forma como o cálculo é feito. Se o cálculo for muito leve, pode ser feito sempre que um utilizador consulta os dados do indicador. Se o indicador não é tão leve devemos usar o repositório como origem dos dados da consulta. Se um utilizador clicar no botão de refrescamento, o cálculo é feito e o utilizador é direccionado para a consulta do novo resultado. Se o cálculo demorar alguns minutos, o indicador deve ser *diferido*. Após o pedido de refrescamento, o utilizador é informado de que o cálculo está a ser efectuado e de que deve voltar mais tarde para ver o resultado.

Os campos *período* e *próximo cálculo* controlam o cálculo periódico do indicador.

O campo *relevância histórica* indica a forma de armazenamento dos resultados. Se devemos guardar todos os resultados dos cálculos coerentes com o *período mínimo de actualização do histórico (PMAH)*, ou se devemos guardar apenas o último resultado. O *PMAH* controla a frequência com que o repositório de resultados é aumentado.

A *heurística de relevância* indica se o cálculo e apresentação de um indicador fazem sentido no momento. É, de momento, uma função.

A *fórmula* de cálculo pode ser definida de duas formas: ou através de uma função PL/SQL capaz de efectuar o cálculo, ou através de uma query SQL para o mesmo efeito. O mecanismo de cálculo é descrito mais à frente.

Há alguns campos associados à apresentação dos dados dos indicadores. A *explicação* permite construir uma, como dizer... explicação... humanamente inteligível, incorporando em si alguns dos dados dos indicadores e dos seus resultados. A listagem de registos problemáticos é produzida com recurso a uma query. Pode ainda ser definido o título desta listagem.

Os indicadores são parametrizados. Na classe parâmetros encontram-se identificados os dados que foram considerados essenciais para cada indicador. São registados o nome, o tipo de dados, uma descrição do parâmetro e, se tal se justificar, algumas observações adicionais. Cada parâmetro tem um conjunto de valores possíveis. Cada um destes valores pode ter uma forma de apresentação, que substituirá o valor sempre que for utilizada para consulta.

Cada indicador elementar possui um conjunto de parâmetros. O campo ordem que surge na classe de associação entre os indicadores e os parâmetros é útil na identificação dos resultados gerados no cálculo e a sua atribuição ao parâmetro respectivo.

Se o indicador elementar tiver relevância histórica, o resultado é armazenado no repositório. A classe *Resultado* identifica esse conceito. Um só cálculo pode gerar múltiplos resultados, distinguíveis entre si pelas combinações diferentes dos parâmetros. Cada resultado é guardado num formato adequado (número, número+número aditivo, XML) e etiquetado com o instante de cálculo. A relação entre o resultado e os respectivos valores dos parâmetros também tem de ser preservada.

Os indicadores compostos são utilizados para agrupar indicadores. A única coisa que têm a mais relativamente ao indicador genérico é a lista de componentes.

O indicador agregado processa os dados de um indicador elementar e mostra-os de forma agregada. Cada indicador agregado processa apenas um indicador elementar. O mecanismo de agregação é descrito mais à frente. Interessa preservar a fórmula de agregação do indicador. Além disto, é necessário definir uma hierarquia de parâmetros. Neste desenho, cada indicador agregado tem uma única fórmula de agregação, agregando da mesma forma independentemente do nível. É concebível (e útil) um indicador agregado em que a agregação é diferente de nível para nível, mas fica para mais tarde. A hierarquia coloca os parâmetros do indicador que é agregado segundo uma ordem (*nível*). Além disso, é preciso configurar a apresentação dos *componentes* do nível. O campo *título* pode ser usado fora de contexto; o campo *título como componente* pode ser usado dentro de um contexto que dispense muita da informação contida no *título*.

A consulta dos indicadores é feita através dos painéis. Cada painel tem uma raiz que tem de ser um indicador composto e um conjunto de grupos de validação que listam o conjunto de utilizadores que podem aceder ao painel. O painel também tem um conjunto de parâmetros. Estes parâmetros são sempre passados aos componentes do painel. Os parâmetros podem ter valores por defeito. Neste caso, se um parâmetro não for passado, o painel assumirá um valor. Se o parâmetro não tiver um valor por defeito, o seu valor deve ser sempre fornecido. Além disso, pode ser dada ao utilizador a possibilidade de alterar o valor do parâmetro usando, por exemplo, uma caixa de selecção. É para isso que serve o campo *escolha*.

Alguns grupos de validação requerem a passagem de parâmetros. Estes parâmetros devem estar ligados de alguma forma aos parâmetros do painel. A classe de associação entre os grupos de validação e os parâmetros permite fazer o mapeamento, e se necessário a conversão dos valores.

A figura 5.2 mostra o diagrama de classes do repositório de funções e de templates. Há três tipos de funções, correspondendo a três formas diferentes de implementar a função. A função pode ser implementada na base de dados, na forma de um procedimento PL/SQL. Neste caso deve ser registado o nome do procedimento. Pode ser implementada num procedimento anónimo guardado num campo da tabela de funções. Ou pode ser implementada a partir de um template. Neste caso deve ser especificado um valor para cada parâmetro de adaptação do template. Se não for especificado um valor, será usado o valor por defeito. Independentemente do tipo, toda a função deve ter um pequeno título identificativo e uma descrição. Podem também ser anexadas observações.

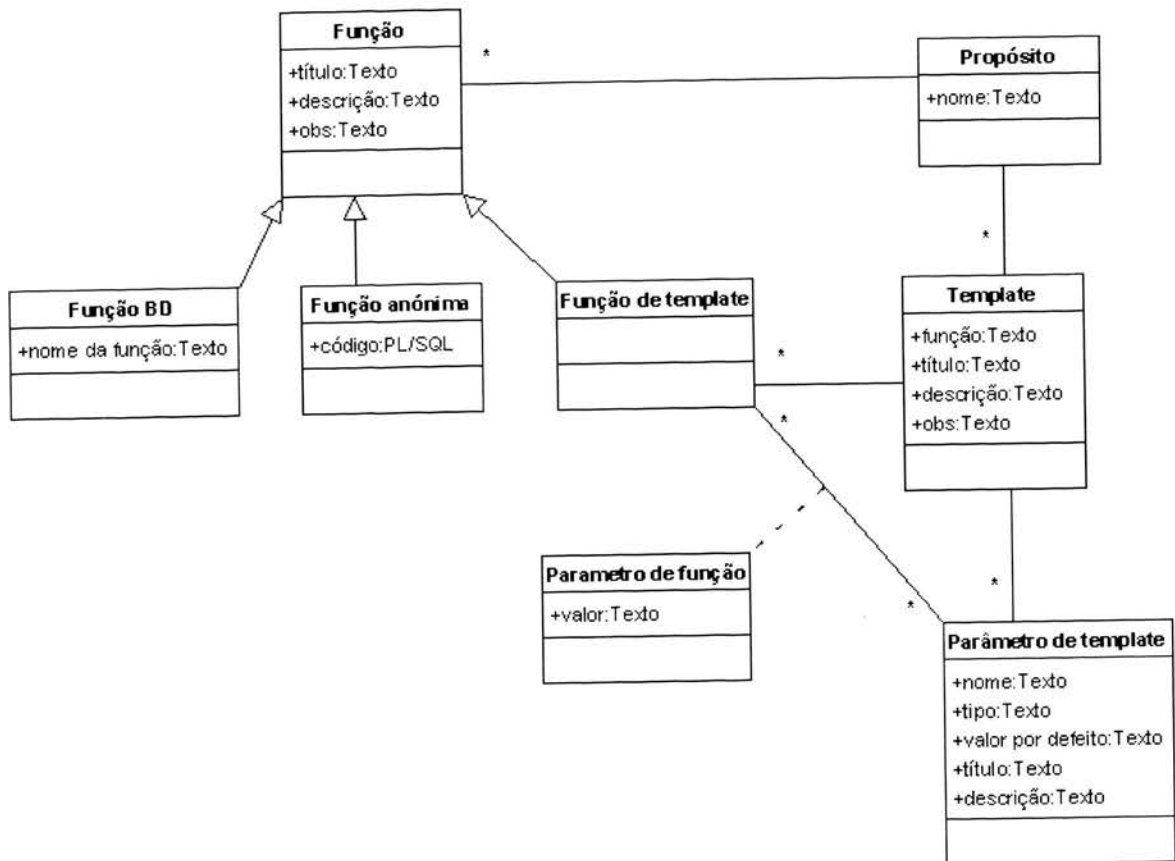


Figura 5.2: Modelo de classes do repositório de funções e templates

Os templates são uma tentativa honesta e generalista de reaproveitar código. Cada template é implementado na forma de um procedimento PL/SQL na base de dados. Cada template deve ter um título identificativo único, uma descrição e possivelmente observações.

As funções que implementam o template requerem um conjunto de parâmetros. Destes parâmetros devemos registar o nome do parâmetro, tal como a função o espera, o tipo de dados PL/SQL do parâmetro, uma descrição, um valor por defeito e observações, se as houver.

O propósito das funções ou templates deve ser declarado. O objectivo desta informação é permitir à aplicação a filtragem de funções ou templates de acordo com o uso que a função vai ter.

Pacotes PL/SQL

A estruturação dos pacotes foi essencialmente determinada pelas normas vigentes no CICA regendo esta questão, e que estão por sua vez limitadas pelas possibilidades de organização de código existentes na plataforma Oracle PL/SQL. O código tem de ser colocado no interior de procedimentos ou funções. Estes podem ser colocados directamente na base de dados ou podem ser agrupados em packages (pacotes). Segue-se o diagrama de pacotes do Painel de Controlo.

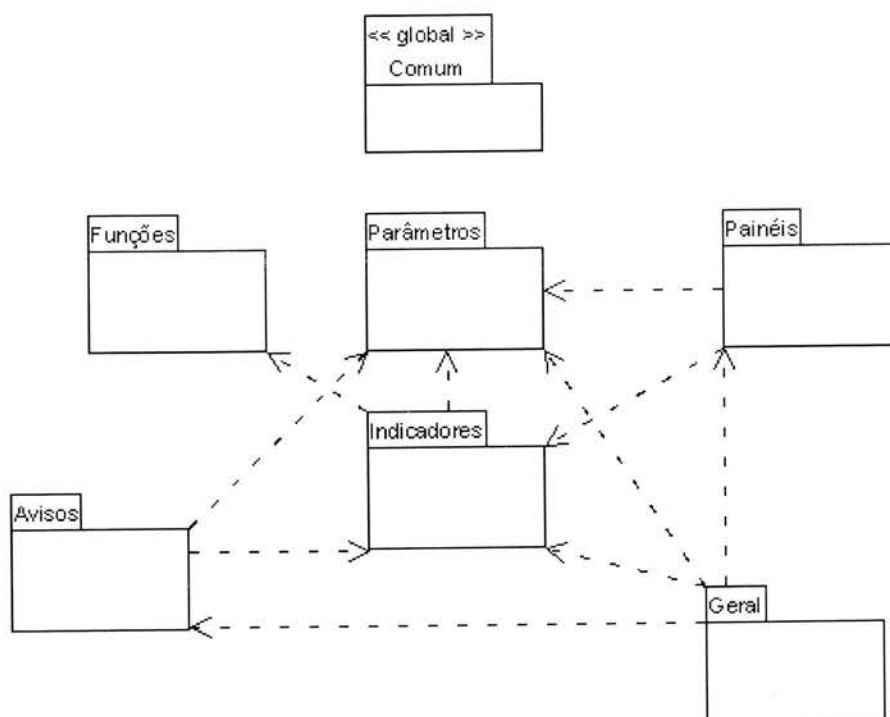


Figura 5.3: A organização em pacotes lógicos

Global

O package `PCONTROL_COMUM_UTIL` contém tudo aquilo que vai ser usado genericamente pelos restantes módulos, como funções utilitárias ou pequenas tabelas para as quais não vale a pena criar uma tabela própria na base de dados.

Funções

O package `PCONTROL_FUNC_UTIL` contém algumas funções de acesso aos dados das funções e dos templates e as API para a chamada de funções.

O package `PCONTROL_FUNC_ADM` implementa o interface web para a gestão das funções e dos templates.

Parâmetros

Uma vez mais as coisas estão divididas em dois packages. O package `PCONTROL_PAR_UTIL` contém funções de acesso aos dados e uma API para manipulação dos parâmetros e de funções utilitárias que dependem dos parâmetros. O package `PCONTROL_PAR_ADM` implementa a interface web de gestão dos parâmetros.

Indicadores

O package `PCONTROL_IND_UTIL` contém funções de acesso aos dados, e API para manipulação dos indicadores e dos resultados. A gestão via web vai para o `_ADM` respectivo.

Painéis e Avisos

Há também dois packages, um `_UTIL` e um `_ADM` com as funções idênticas aos anteriores, mas para os painéis.

Geral

O package PCONTROL_GERAL contém a interface web propriamente dita. São implementadas aqui todas as páginas web destinadas aos responsáveis de dados. As páginas de geração de gráficos também estão implementadas aqui.

O package PCONTROL_WEB contém funções auxiliares para as páginas. A autorização ou a construção da apresentação dos indicadores são feitas aqui.

5.4 Alguns mecanismos

Funções e templates

Dentro dos indicadores há determinadas partes que exibem um grau elevado de variabilidade. O repositório de funções e templates foi introduzido para lidar com esta variabilidade. A fórmula de cálculo do resultado, a heurística de estado, a heurística de relevância, a fórmula de agregação ou o desenho de gráficos específicos podem ser feitos por funções do repositório. A necessidade do registo de funções tem duas origens: por um lado a necessidade de cadastro para uma manutenção mais fácil. Por outro, para promover o reaproveitamento de código através do uso de templates.

Vamos descrever o ciclo de vida de uma função baseada num template. O ciclo de vida consiste em:

- criar o template;
- criar a função;
- chamar a função;
- alterar a função.

Começamos então criando o template. Damos um nome ao template, escolhemos um propósito, introduzimos o nome da função e introduzimos os parâmetros de que o template precisa. De seguida (ou antes, é indiferente) implementamos a função em PL/SQL na base de dados. É tudo.

Agora a função: título; descrição; escolhemos o propósito; escolhemos o template do qual vamos derivar a função; atribuímos valores a cada um dos parâmetros da função. É isto, a função está pronta a usar.

Agora vamos descrever o mecanismo de chamada da função. Digamos que é uma função de estado. Seja:

```
PROCEDURE sem3estdesc(
    p_transicaoava FLOAT,
    p_transicaoav FLOAT
)
IS
    ln_valor NUMBER;
BEGIN
    ln_valor := pcontrol_ind_util.grec_estado.n_valor;
    pcontrol_ind_util.grec_estado.rec_estado.tbl_trans :=
        pcontrol_comum_util.number_tbl_type(p_transicaoava, p_transicaoav);
    IF ln_valor > p_transicaoav THEN
        pcontrol_ind_util.grec_estado.rec_estado.i_estado:=1;
    ELSIF ln_valor > p_transicaoava THEN
        pcontrol_ind_util.grec_estado.rec_estado.i_estado:=2;
```

```

ELSE
    pcontrol_ind_util.grec_estado.rec_estado.i_estado:=3;
END IF;
END sem3estdesc;

```

Esta função muito simples implementa um template que recebe o valor do resultado e dois parâmetros de adaptação representando dois valores de transição. A função compara o resultado com os dois valores de transição, e devolve o estado respectivo. Quando na execução do cálculo a altura de calcular o estado chega, é necessário colocar todos os valores de que a função necessita numa variável global. Consoante o tipo de função, é gerado código para efectuar a chamada. Neste caso, uma vez que a função implementa um template, o código gerado consiste na chamada à função com os valores respectivos dos parâmetros de adaptação.

A transferência de dados entre a função e a aplicação é feita através de variáveis globais, por razões de performance e flexibilidade.

Cálculo do resultado e do estado de um indicador elementar

Nesta secção vamos analisar como e quando vamos calcular os indicadores. Os indicadores têm diferentes características. Uns são leves, calculados rapidamente e sem grande consumo de recursos, outros são pesados, consumindo muito tempo de processador e fazendo grandes quantidades de escrita e de leitura de dados. Uns têm valores com significado apenas no momento presente, outros têm valores cuja variação também pode ser alvo de análise. Uns devem ser calculados periodicamente, outros não. Uns devem mostrar o valor calculado há umas horas, outros devem calcular sempre um valor na hora. E quanto ao registo do histórico, sendo que um utilizador tem a possibilidade de refrescar o indicador sempre que achar conveniente, deverá a actualização reflectir-se no histórico ou não? Sumariando:

- peso:
 - pluma;
 - leve;
 - pesado ;
- relevância do histórico :
 - sim;
 - não;
- período mínimo de actualização do histórico;
- período mínimo de refrescamento;
- periódico:
 - sim:
 - próximo cálculo;
 - período;
 - não.

Peso

O impacto do peso sente-se sobretudo no refrescamento. Cálculos pesados devem ser diferidos quando um utilizador tenta refrescar o valor, sendo o utilizador convidado a voltar mais tarde para visualizar o resultado. Cálculos leves devem conduzir de imediato o utilizador ao resultado. Em termos de implementação, é a diferença entre um cálculo usando jobs e um cálculo feito de imediato na chamada à respectiva página web.

Por outro lado, se um indicador é considerado pesado, não faz sentido classificá-lo como não periódico com o cálculo efectuado na hora. Tal privilégio deve ser reservado aos cálculos leves.

Cálculos muito leves de indicadores que variam rapidamente podem ser efectuados implicitamente sempre que o resultado do indicador é necessário. O indicador deve ser calculado “na hora”.

Relevância e actualização do histórico

Os indicadores sem relevância histórica devem ver os seus resultados antigos descartados. Deve ser guardado apenas o resultado mais recente para cada combinação de parâmetros. Quando o indicador tiver relevância histórica, deve ser observado o mecanismo descrito na secção 3.7 para controlar o crescimento do repositório de resultados.

Refrescamento

O utilizador pode, se que achar conveniente, refrescar o conteúdo do indicador. Já referimos o impacto do peso do cálculo na forma como o refrescamento é feito. Vamos agora discutir limites a este refrescamento. É inútil, em pelo menos 99% dos casos, fazer dois refrescamentos seguidos. Para poupar os recursos do sistema podemos introduzir um *período mínimo de refrescamento*, abaixo do qual o refrescamento não será feito. Quando um utilizador tentar refrescar um indicador calculado há pouco tempo, deve ser-lhe dada uma explicação da razão pela qual o refrescamento não é feito e o utilizador deve ser encaminhado para o resultado (pouco) antigo.

Mecanismos

Como vimos o cálculo pode ser lançado por diversas vias. Pode ser lançado periodicamente, pode ser lançado através de um pedido de refrescamento do utilizador, ou calculado de cada vez que o indicador é mostrado. Vamos descrever primeiro cada um dos mecanismos de lançamento do cálculo, e depois vamos descrever o cálculo em si.

O cálculo periódico é definido por um administrador. Antes de lançar o cálculo periódico o administrador deve definir o período de cálculo, a data do próximo cálculo e o regime de relevância histórica do indicador. Em Oracle PL/SQL existe um mecanismo para correr processos de forma periódica, os *jobs*. Quando se lança um cálculo periódico, é lançado um *job*. O cálculo é corrido com todos os parâmetros do indicador indefinidos, cujas implicações dependem da fórmula de cálculo e são apresentadas mais à frente.

O refrescamento pode ser pedido por um utilizador comum. O utilizador pode pedir um refrescamento em qualquer tipo de indicador. Se o indicador for composto, serão refrescados todos os seus componentes. É necessário ter cuidado com os indicadores que surgem em mais do que um sítio. Não que cause problemas, mas não há necessidade de os calcular duas vezes. O refrescamento de um indicador agregado reduz-se ao refrescamento do respectivo indicador elementar. O cálculo é corrido com os valores dos parâmetros constantes no painel na altura do cálculo.

Se o cálculo for definido como diferido, é lançado um job para o efectuar. Senão, é calculado pelo processo web criado para responder ao pedido de refrescamento.

Há ainda a possibilidade de o cálculo ser efectuado automaticamente de cada vez que um indicador é lido. Neste caso o cálculo é lançado com os valores dos parâmetros constantes no painel na altura do cálculo.

Antes do cálculo ser efectuado, é necessário preparar uma estrutura de dados para receber os resultados. Em geral o cálculo consiste numa query que devolve uma tabela de resultados. O número de colunas da tabela depende do indicador em particular:

- uma coluna para o resultado;
- uma coluna para o resultado aditivo;
- uma coluna para o resultado XML;
- uma coluna para cada valor de informação adicional.
- uma coluna para cada parâmetro;
- uma coluna para o estado;
- uma coluna para as transições de estado.

A fórmula de cálculo pode ser um procedimento PL/SQL que preencha esta estrutura de dados, ou pode ser uma query SQL parametrizada com a estrutura exemplificada:

```
SELECT
    COUNT(*) ,
    NULL,
    NULL,
    NVL(TO_CHAR(f.ano), 'Sem ano') AS P_ANO_CURRICULAR,
    f.curso AS P_CURSO
FROM
    alunosof f,
    alunosx x
WHERE x.cod = f.numero
    AND
    x.estado='F'
    AND
    (
        <param>P_CURSO</param> IS NULL
        OR
        f.curso = <param>P_CURSO</param>
    )
    AND
    (
        <param>P_ANO_CURRICULAR</param> IS NULL
        OR
        <param>P_ANO_CURRICULAR</param> = <nulo>P_ANO_CURRICULAR</nulo>
        AND
        f.ano IS NULL
        OR
        f.ano = <param>P_ANO_CURRICULAR</param>
    )
GROUP BY
    f.ano,
    f.curso;
```

Esta query faz parte de uma fórmula que calcula o número de alunos de cada curso em cada ano curricular. O indicador tem dois parâmetros: o ano curricular e o ano lectivo. A primeira

coluna seleccionada é sempre o resultado e é tipicamente uma função de agregação (neste caso um COUNT). Depois vem a coluna do valor aditivo que é nula nesta fórmula. Se o resultado fosse uma percentagem, por exemplo, seria necessário colocar neste campo o número total de registos processados. A terceira coluna está reservada para os resultados XML. De seguida devem surgir colunas com informação associada. Neste caso particular não há qualquer dado para a tabela informação. No fim surgem as colunas contendo os valores dos parâmetros que identificam o resultado. Estes campos têm de surgir também na clausula GROUP BY, como é natural.

A cláusula FROM, juntamente com a cláusula WHERE identifica a origem dos dados. Vamos analisar a cláusula WHERE na sua dependência dos parâmetros P_CURSO e P_ANO_CURRICULAR. O que lá está pode ser traduzido assim:

Se é especificado um curso,

Calcular apenas resultados para esse curso.

Senão,

Calcular para todos os cursos.

Se é especificado um ano curricular,

Se o valor corresponde à ausência de ano curricular,

Calcular apenas resultados para alunos não incluídos num ano curricular.

Senão, calcular apenas resultados para esse ano curricular.

Senão calcular os valores para todos os anos curriculares.

Esta fórmula de cálculo pode então ser corrida de quatro formas:

- sem parâmetros, calcula o número de alunos em todos os anos curriculares de todos os cursos;
- calcula o número de alunos em cada ano curricular de um curso;
- calcula o número de alunos de um ano curricular para cada curso;
- calcula o número de alunos num ano curricular de um curso.

Os valores dos parâmetros são passados à fórmula marcando a sua posição com marcas tipo XML. Neste caso há duas marcas definidas:

- param, que é substituída pelo valor do parâmetro;
- nulo, que é substituída pelo valor que representa a ausência de um valor. É uma forma de dizer que queremos o resultado correspondendo às situações em que o dado representado pelo parâmetro é nulo.

A última forma é usada tipicamente num pedido de refrescamento de um indicador elementar. As três anteriores podem ocorrer num pedido de refrescamento de um indicador agregado. A primeira pode ainda acontecer no cálculo periódico de indicadores, situação em que nunca são passados parâmetros. Se for necessário fornecer um valor por defeito, deve ser adicionado código SQL para o efeito. Os resultados do cálculo são recolhidos numa lista e são, após algum processamento extra, escritos na base de dados.

Após o cálculo, e já com os dados empacotados numa estrutura de dados, a heurística de estado é corrida sobre os mesmos. O estado e os valores de transição são então anexados aos resultados respectivos.

Por fim, os resultados são gravados no repositório.

Relevância

A relevância é, de momento, determinada por uma heurística própria de cada indicador. No futuro pode basear-se em informações originárias do registo de processos da instituição.

Cada indicador tem a sua heurística. Esta heurística é uma função presente na base de dados que retorna 1 caso o indicador seja relevante e 0 se não. A heurística de relevância é usada nas seguintes situações:

- no cálculo de indicadores compostos; indicadores irrelevantes podem não contar na determinação do estado do indicador composto;
- na apresentação da página, os indicadores irrelevantes podem ser marcados como tal.

Composição

Os indicadores compostos são usados para agrupar outros indicadores. Têm estado, de forma a propagar para cima a informação sobre os estados dos seus componentes, mas não tem associado um resultado.

Criação

Na criação de um indicador composto é preciso ter atenção à seguinte regra:

- os indicadores elegíveis para integrar o indicador composto são todos aqueles que não são nem seus antecessores nem seus componentes.

Isto é, não posso adicionar ao indicador composto indicadores dos quais o indicador composto é um componente, nem indicadores que já fazem parte da composição. A segunda restrição é fácil de compreender, pois serve apenas para evitar a duplicação de componentes. A primeira impede a existência de ciclos na definição de indicadores compostos, factor que complicaria a implementação sem trazer qualquer vantagem.

Cálculo

O cálculo do estado é efectuado usando uma heurística de estado definida com o indicador. Antes da chamada à heurística devemos ler todos os componentes do indicador e publicá-los numa variável global acessível ao procedimento PL/SQL que implementará a heurística. A fórmula lê os estados de cada componente, verifica a relevância ou outras informações a considerar no cálculo, e computa um valor para o estado.

A heurística pode variar consoante as necessidades.

Agregação

Voltando ao exemplo clássico da taxa de preenchimento das fichas de disciplina, um dado responsável de dados pode ter apenas interesse nos dados de um curso, enquanto que um outro pode ter interesse nos dados de todos os cursos de uma instituição. Ao primeiro, o indicador elementar basta. Ao segundo permitimos observar o resultado agregado para todos os cursos. Se o utilizador pretender, pode fazer *drill in* clicando no ícone de inspecção do indicador. Essa acção abre um painel onde são listados os valores para cada um dos cursos. Para isto usamos indicadores agregados.

O indicador agregado funciona em cima de um indicador elementar, para o qual é definida uma ordem hierárquica para os parâmetros.

A inclusão de um indicador agregado num painel ou num indicador composto que requeiram já alguns dos seus parâmetros resulta na sua aparente supressão. Aparente porque o que acontece é que o indicador herda os valores dos parâmetros do pai, resultando não na supressão dos parâmetros, mas na supressão da navegação nos respectivos níveis. Por exemplo, o indicador agregado anterior incluído num painel que exija um ano lectivo, mostra a informação agregada dos dados no ano lectivo dado, não dando ao utilizador a possibilidade de escolher o ano lectivo.

A agregação pode ser feita usando os resultados constantes no repositório, o que em muitos casos leva a que sejam agregados dados calculados em instantes diferentes, consoante os refrescamentos requeridos pelos utilizadores. Este problema só não surge quando se agregam resultados de indicadores calculados na hora.

Quando um indicador elementar é criado, o criador deve ter em mente se o indicador vai ou não ser posto sob agregação. Se o resultado do indicador não for aditivo, devem ser armazenados dados que permitam recuperar a aditividade. Esta situação é tão comum que se adicionou ao repositório de resultados um campo só para esta informação. No caso de uma percentagem, por exemplo, podemos armazenar neste campo o número total de registos. As fórmulas de agregação destes indicadores devem ser concebidas tomando este campo em consideração.

Posto isto passamos a descrever sumariamente a implementação do mecanismo de agregação. Este foi um dos mecanismos que mais evoluiu ao longo do tempo. O mecanismo é agora algo simples, mas só atingiu um tal ponto de simplicidade após algum trabalho.

O indicador agregado é definido com uma fórmula de cálculo que efectua a agregação, e uma heurística de estado que pode ser herdada do indicador elementar cujos dados são agregados. No seu essencial as fórmulas de agregação serão provavelmente implementadas da mesma forma, variando apenas a função de agregação. Na implementação da agregação é preciso ter em atenção a origem dos resultados. No caso dos indicadores calculados na hora, os resultados são também eles calculados na hora. Noutros casos, os resultados vêm do repositório. Consoante a origem, é construída uma query para ler os dados. Esta query é passada então à fórmula usando variáveis globais. A fórmula toma a query, analisa os parâmetros e seus valores, e constrói dinamicamente uma outra query capaz de fazer a agregação desejada. Após a agregação é calculado o estado do indicador.

Consoante o uso que se vai dar ao indicador, a agregação pode ser feita a um nível ou a dois. A um nível, precisamos apenas de um resultado e de um estado. A dois, precisamos ainda de uma lista de componentes, que consiste nos resultados da agregação do nível seguinte para cada um dos valores possíveis dos parâmetros. O uso dado a estes é descrito já a seguir.

Construção das páginas

A produção da páginas é feita em dois passos. Primeiro os dados do indicador a mostrar são lidos e colocados num formato intermédio. Esse formato intermédio é posteriormente processado e transformado numa página HTML. Cada página é ainda dividida em várias partes.

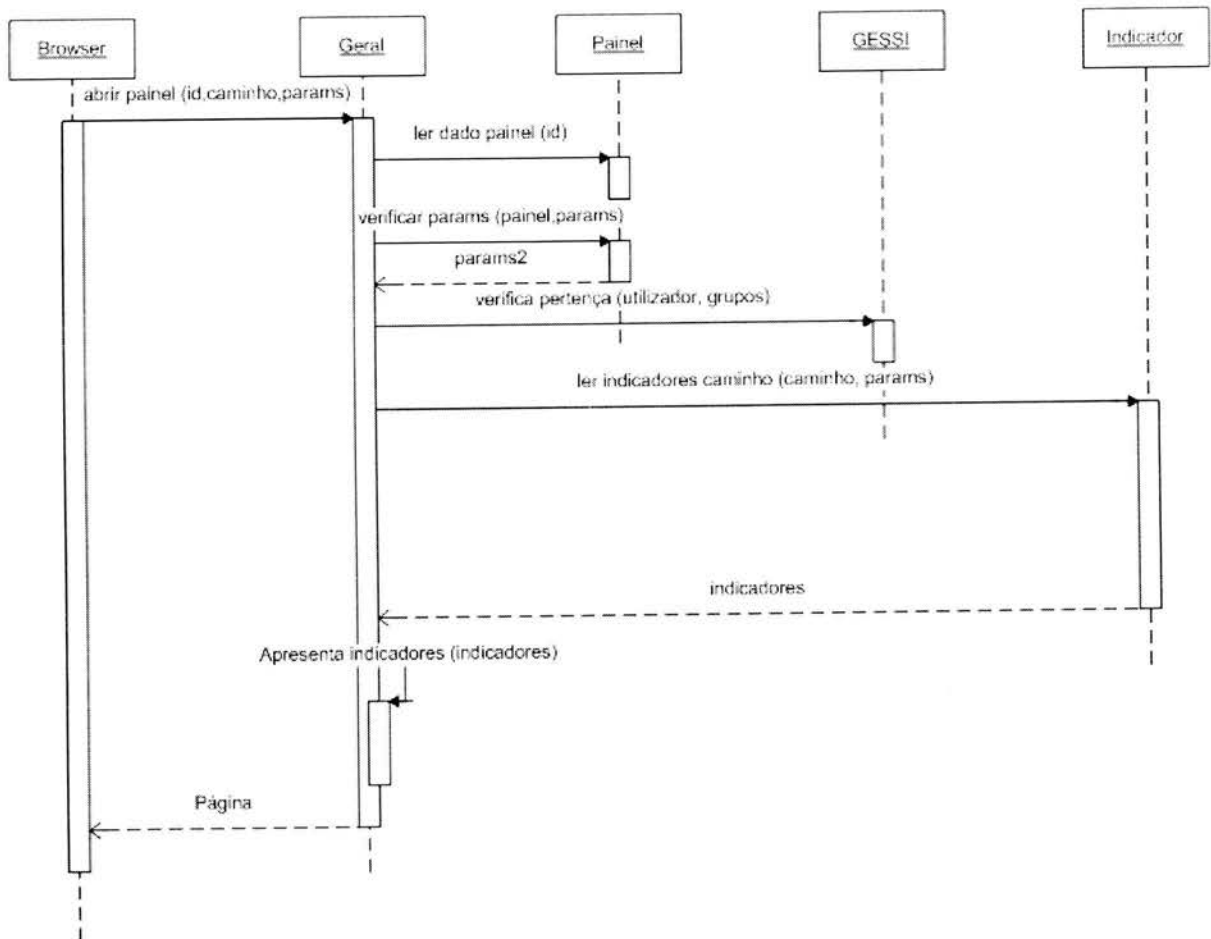


Figura 5.4: Chamada e construção da página

Autorização

Cada painel pode ser consultado por um grupo restrito de pessoas. A autorização faz-se mediante a verificação da pertença de uma pessoa a um grupo de validação com autorização para consultar o painel. Cada painel tem a sua lista de grupos de validação autorizados.

O GESSI tem um conjunto de funções que permitem fazer isto. Tudo o que temos de fazer é ler a lista de grupos e verificar um a um se o utilizador lhe pertence. Se o utilizador pertencer a um dos grupos obtém autorização. Senão o acesso é negado.

Barra onde

Acima, no cabeçalho, surge a *barra onde*. Esta barra indica o caminho até à página que estamos a visualizar. Cada elemento do caminho é clicável, permitindo subir na hierarquia. Esta funcionalidade exige a manutenção da informação sobre o caminho, que devemos passar num parâmetro HTTP. Este caminho consiste numa lista de códigos de indicadores separados por barras, cada elemento indicando a passagem por um indicador na navegação. Um exemplo:

`p_caminho=21/33/33/37&p_nomes=P_ANO_LECTIVO&p_valores=2004/2005 ...`

Esta linha ilustra um cenário em que o indicador 33 surge duas vezes, e é um indicador agregado. O indicador 21 é um indicador composto e o 37 é um indicador elementar. Ou seja, entramos no indicador composto 21, e abrimos o indicador agregado 33. Dentro deste indicador escolhemos ver os dados do 1º semestre, e abrimos de novo o indicador 33, desta

vez apenas com os dados do 1º semestre. Finalmente saltamos para os detalhes do indicador para a LEIC. Pretendemos cruzar a informação do caminho e dos parâmetros de forma a construir URL que nos permitam saltar para um nível por onde já passamos.

Indicador	Ano lectivo	Período	Curso
21	2004/2005		
33	2004/2005		
33	2004/2005	1º semestre	
37	2004/2005	1º semestre	LEIC

Figura 5.5: Herança dos valores dos parâmetros

O parâmetro *ano lectivo* de 21 é herdado por todos os seus descendentes. Quando se entra no indicador agregado, a descida na hierarquia faz-se especificando valores para os parâmetros, até chegar à folha 37, que é onde estamos. Temos o caminho (a primeira coluna) e a lista de parâmetros em 37 (a última linha), e temos de reconstruir os dados da figura 5.5 com eles.

Isto implica a leitura de todos os indicadores da lista, a identificação do tipo de cada um e dos parâmetros que requerem e a construção de um URL para cada um. O último é especial, porque representa o ponto em que estamos, e devemos ler todos os dados deste indicador necessários para construir a página.

Leitura de dados

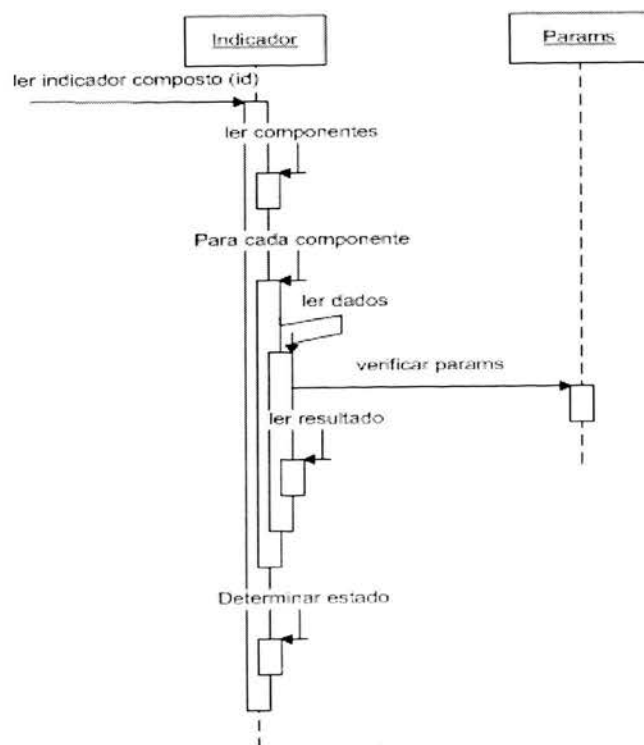


Figura 5.6: Indicador composto

Dependendo do indicador a ler, a leitura de dados a mostrar processa-se de forma diferente, embora o resultado final seja a sua colocação numa estrutura de dados comum. Começamos por ler os dados da tabela de indicadores. Consoante o tipo, é seguido um procedimento diferente de leitura de dados. O mais simples é o da leitura dos indicadores elementares. Consiste apenas em verificar os parâmetros e ler o resultado e estado.

A leitura de dados de um indicador composto processa-se como indicado na figura 5.6. Os dados constantes na tabela de indicadores são lidos, seguindo-se a leitura da lista de componentes. Os dados de cada componente são então lidos recursivamente. No final ficamos com uma lista completa de componentes e seus dados, bem como a lista completa de dados do indicador composto.

O indicador agregado começa, como os restantes, por verificar os seus parâmetros. De seguida a fórmula de agregação é corrida, quer com resultados calculados na hora, quer com

resultados do repositório. Como já foi referido os resultados são agregados em dois níveis. O estado é calculado a partir do resultado do primeiro nível. Depois, cada resultado do segundo nível é processado e são criados componentes idênticos aos componentes de um indicador composto, um por cada parâmetro. Nesta fase é necessário determinar o estado, criar um título, definir os parâmetros do componente, etc.

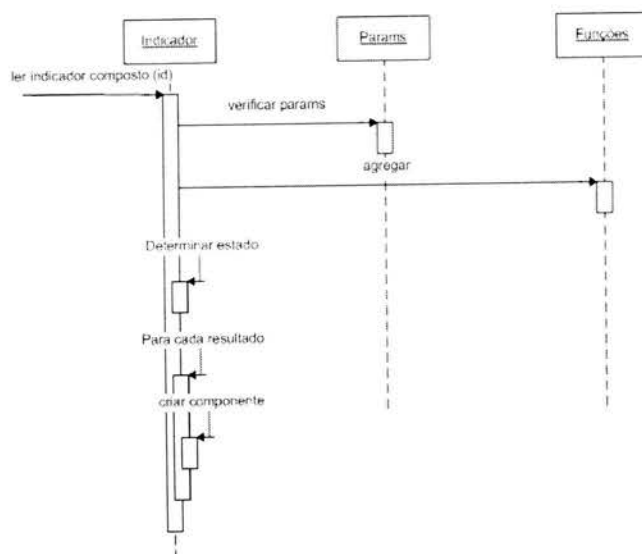


Figura 5.7: Indicador agregado

Maxi e Mini

Cada indicador pode ser mostrado de duas maneiras. Uma é como prato principal, num painel próprio, ocupando toda a largura disponível da janela para mostrar toda a informação relevante para a identificação de problemas. A outra é como componente de um indicador composto ou como pseudo componente de um indicador agregado, numa janelinha pequena, onde o objectivo é mostrar o máximo de informação no mínimo espaço possível.

Digamos que *maxi* é a definição dos componentes a mostrar na janela grande, e *mini* é a definição dos componentes a mostrar na janela pequena. Este objectivo é atingido através do uso de XML. O administrador deve usar um conjunto de marcas predefinidas para configurar a apresentação dos indicadores. Esse XML é depois processado por uma XSLT e transformado em código PL/SQL capaz de produzir a apresentação do indicador. Por motivos de performance, a transformação do XML em PL/SQL é feita na altura da definição, evitando a necessidade de correr a XSLT sempre que a página é chamada.

Os componentes que podem ser mostrados são:

- gráficos:
 - gráfico estado/resultado;
 - gráficos contextuais, que apresentam dados muito próprios do indicador;
 - comuns:
 - barras horizontais;
 - barras verticais;
 - bolo;
 - ...;
 - definidos;
 - usam o repositório de funções;
- textuais:
 - lista de registos problemáticos;
 - explicação;
 - lista de avisos;
 - lista de links;
- mistos
 - folha descritiva

- título;
- descrição;
- instante de cálculo;
- ...

O conjunto de marcas a é detalhado no apêndice D.

Explicações

A explicação é um texto humanamente inteligível. A construção deste texto é feita de forma semelhante à construção da página do indicador a partir de XML.

A explicação é no fundo um texto, no meio do qual podemos inserir valores relacionados com o indicador através de um conjunto de marcas XML. O XML é processado por uma XSLT na altura da definição para produzir código PL/SQL capaz de construir a explicação.

O conjunto de marcas disponíveis é detalhado no apêndice C.

Registos problemáticos

A listagem de registos problemáticos surge nos indicadores elementares. O formato exacto da listagem pode depender do indicador. Nalguns casos uma lista pode ser a forma de apresentação mais apropriada, mas noutros casos devemos optar por tabelas ou outras formas.

Uma forma conveniente de permitir este tipo de configurabilidade consiste em definir a lista de registos problemáticos usando SQLX. Com SQLX podemos definir XML a partir de uma query SQL. Podemos usar então o SQLX para produzir XHTML directamente a partir da base de dados. Um exemplo:

```
SELECT
  XMLElement( "ul",
    XMLAgg(
      XMLElement( "li",
        XMLElement( "a",
          XMLAttributes(
            'disciplinas_geral.FormView?p_cad_codigo='||d.codigo||'&p_ano_lectivo='||
              ano_lectivo AS "href",
            '_top' AS target
          ),
          d.nome
        )
      )
    )
  )
FROM
  disciplinas d,
  ocorrencias o,
  cursos c
WHERE
  d.codigo = o.cad_codigo
AND
  d.cur_codigo = c.codigo
AND
  o.ano_lectivo = <param>P_ANO_LLECTIVO</param>
```

```

AND
c.sigla = <param>P_CURSO</param>
AND
o.periodo = <param>P_PERIODO</param>
AND
o.objectivos IS NULL

```

Qualquer semelhança entre esta query e a query mostrada na explicação do mecanismo do cálculo não é mera coincidência. De facto, as marcas XML usadas coincidem com as apresentadas nessa secção, e têm o mesmo efeito. A query apresenta uma lista de links para as páginas das disciplinas cuja ficha de disciplina não tem o campo *objectivos* preenchido. O elemento mais externo é o *UL*, usado para enclausurar uma lista não numerada. Segue-se uma lista de elementos *LI*, ou *list items*, cada um dos quais contém uma ancora HTML para a página do SIGARRA com os dados da disciplina cuja ficha está em falta.

Gráficos

Como já foi referido, os gráficos têm o objectivo de tornar o painel de controlo mais compacto, intuitivo e agradável. O tipo de gráfico mais adequado para mostrar os resultados pode variar consideravelmente de indicador para indicador.

A tecnologia escolhida para desenhar os gráficos foi o SVG (Scalable Vector Graphics). As razões para a escolha residem no facto de ser um standard da internet gratuito, ser simples quanto basta, e de ter um conjunto rico de funcionalidades, garantindo muita flexibilidade. As desvantagens da tecnologia são sobretudo o facto de ser recente, e consequentemente a sua implementação ser ainda um pouco imatura (bugs) e de ser necessário um plugin para visualizar o gráfico. Mas é muito provável que muitas destas desvantagens desapareçam num futuro próximo.

Uma alternativa forte foi a utilização de pacotes gráficos Oracle, já utilizados em muitas aplicações do SIGARRA. Tinham duas pequenas desvantagens. Eram pouco flexíveis (algo com que era possível lidar) e de difícil compactação (esta desvantagem colide com a nossa necessidade de compactação dos dados). Por isso, por sugestão do Luís Ramos, a opção foi SVG.

O uso mais simples consiste em desenhar um pequeno gráfico para mostrar o estado e o resultado do indicador. Este tipo de pequenos gráficos pode ser construído e mostrado ao utilizador usando páginas web responsáveis pelo desenho genérico deste tipo de gráficos. A página funciona exactamente como uma página web normal, apenas devolvendo SVG em vez de HTML.

Colocamos na página HTML onde o gráfico deve ser desenhado uma tag *object* ou *embed* para chamar e apresentar o gráfico. Neste caso devem ser passados à página responsável pelo desenho do gráfico todos os dados que são necessários para o seu desenho. Por exemplo, para um gráfico de barras indicando o resultado e o estado devemos passar:

- largura do gráfico;
- altura do gráfico;
- valor mínimo;
- valor máximo;
- cores;
- resultado;
- estado;

- transições.

Outra situação é a de gráficos mais complexos em que não é conveniente passar todos os dados na chamada HTTP. Por exemplo, quando queremos fabricar um gráfico de barras com os resultados e estados de um indicador agregado. Continuamos a passar a informação sobre o desenho do gráfico por HTTP, mas os dados dos indicadores são lidos da base de dados:

- largura do gráfico;
- altura do gráfico;
- valor mínimo;
- valor máximo;
- painel;
- caminho;
- parâmetros.

A última situação a abordar é a de gráficos muito específicos. Nestes casos o administrador deverá construir uma função PL/SQL capaz de desenhar o gráfico e registá-la no repositório. A chamada HTTP é direccionada a uma página própria para estes casos, e identifica apenas o o painel, o caminho, os parâmetros e a função responsável pelo desenho do gráfico. A página deve apenas ler os dados do indicador e chamar a função de desenho respectiva.

Em todos os casos, a disposição do gráfico é definida nos campos mini ou maxi respectivos, usando marcas XML. O anexo D lista estas marcas e suas características.

Avisos

Podemos definir avisos sobre o resultado dos cálculos dos indicadores. Esses avisos devem ser lançados sempre que um indicador atinja um valor indesejável.

O aviso pode ser enviado via e-mail ou através de notificações no próprio sistema de informação.

A determinação das pessoas a avisar devia ser feita por papel. Tal exigiria a existência de um módulo suportando a definição de processos. Ainda não há, embora esteja na neste momento a ser desenvolvido pela Susana Pinho. A determinação pode ser feita por queries que determinam o papel dos utilizadores, ou os utilizadores com determinado papel. Há no SIGARRA um módulo de validação que agrupa as pessoas por grupos de validação. Estes grupos determinam se uma pessoa faz parte de um grupo implicitamente através de uma query própria ou explicitamente através de uma lista de pessoas. Não são exactamente definições de papéis, mas estão lá perto. O módulo de validações permite ainda controlar se um utilizador é ou não notificável, informação que nos permitirá controlar a decisão de quais os receptores dos avisos.

Os grupos de validação traduzem a posse e a delegação de papéis, mas penso que têm o defeito de ser orientados para a função de autorização e a sua implementação é um pouco deficiente na função de resolução de papéis. De qualquer maneira a determinação dos responsáveis a alertar é sempre dinâmica.

Os avisos podem ser manuais ou automáticos. No caso dos avisos manuais, o painel fornece ao responsável de dados um botão a clicar para alertar responsáveis. O botão deve surgir sempre que o estado indicar a existência de problemas.

Os avisos automáticos são despachados sem intervenção humana, após um cálculo que revela a existência de um problema. Para garantir que as pessoas a avisar não são bombardeadas todos os dias com dezenas de mensagens, definiu-se um processo de avisos implementado sobre o motor de workflow. O diagrama de actividades 5.8 ilustra o processo.

A emissão de avisos não é universal. Para que sejam emitidos avisos, eles devem ser registados primeiro junto do indicador respectivo. A definição de um aviso implica a definição do grupo de validação destinatário e do estado que o desencadeia. No caso de um aviso automático deve também ser definida a política a seguir no caso de persistência do problema.

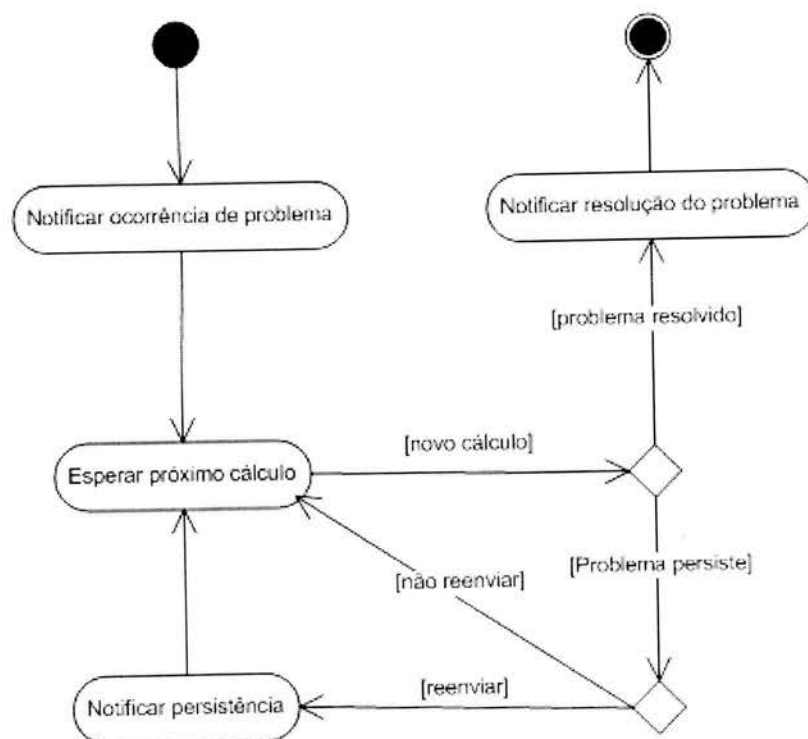


Figura 5.8: Diagrama de actividades dos avisos

6 Conclusões e perspectivas

Esta secção é dedicada à análise do trabalho, o que ficou bem ou mal e o que deve ser feito no futuro para o melhorar. A minha impressão geral é positiva. O trabalho era algo complexo e foram obtidos resultados concretos, e embora o trabalho não esteja completo é possível vislumbrar um fim (não, não é um dos 90% dos projectos a 10% do fim – ainda faltam uns 30% a 40%). Alguns dos aspectos do trabalho foram divertidos e desafiadores. Entre as coisas que correram bem, gostaria de salientar as seguintes.

6.1 Aspectos positivos

A visão do sistema

A ideia do que o painel de controlo deve ser, com um aspecto marcadamente gráfico e hierárquico, com os indicadores implementados apenas ao nível lógico, sem preocupações de programação da interface em cada um deles traz um enorme valor à instituição. Ficou provado neste trabalho que o módulo é viável.

Desenvolvimento iterativo

Dei-me muito bem com a abordagem iterativa. Tentei desenhar as coisas de forma a que fossem alteráveis à medida que as ideias fossem surgindo, o que se revelou uma boa aposta num trabalho exploratório como este.

Flexibilidade

Neste módulo, a ideia sempre foi a de misturar dados de natureza diferente. Cada indicador tem as suas peculiaridades e por isso tentei dar o máximo de flexibilidade ao sistema. Acho que fui bem sucedido.

SVG

Gostei! Tem um grande futuro. As implementações disponíveis ainda estão um pouco instáveis (não consegui usar o plugin em condições no Firefox) e o uso está dependente da instalação de um plugin para o browser, mas é muito fácil e agradável criar gráficos com esta ferramenta.

Configurabilidade

O nível de configurabilidade atingido permite espaço de manobra na implementação de muitos indicadores.

6.2 Aspectos negativos

PL/SQL

É muito conveniente para programar páginas web, mas não favorece a produção de bom código. É cheio de qui pro quos e pouco flexível, e carece de uma melhor organização.

Incompletude

Ainda ficou muito por fazer. Os avisos ou a vista de alarmes ao nível funcional, a usabilidade, a robustez ou a performance ao nível não funcional. Gostava de ter feito mais. Tenho no entanto a consciência de que o que está feito não foi fácil.

Configurabilidade

Um verdadeiro pau de dois bicos. Em excesso pode causar graves problemas de usabilidade. A definição de indicadores é um bocado difícil.

Organização

Se alguém pretender definir desorganização e quiser dar um exemplo, pode imediatamente apontar para mim. Como tal, eu tinha necessidade de um ambiente bem organizado e capaz de me manter *sobre os carris*. Fez-me falta a existência de um processo de software na USINF, para mais porque com o crescimento da unidade o chefe está cada vez mais ocupado. Esta falha está a ser colmatada, e fico feliz por saber que os 5 estagiários que se seguem vão ter a sorte de trabalhar num ambiente mais organizado.

Semi-spaghetti

Pois é, o código não está um emaranhado tipo esparguete, mas também não está muito longe. Vou ter que fazer um esforço, no tempo que ainda tenho, de documentar quase cada linha. A não trivialidade de muito do código a isso abriga.

Pequenas distrações

O facto de ter a máquina numa posição onde ninguém via o que estava a fazer, aliada à falta de metas frequentes não favoreceu muito a disciplina. Acho que perdi algum tempo (embora não demasiado) com pequenas distrações.

6.3 Cumprimento dos objectivos

O objectivo era, desde o início, construir um painel de controlo orientado à monitorização da qualidade de dados do SIGARRA. O que dá para a qualidade de dados dá para outras coisas, e no final o painel de controlo pode ser encarado como de monitorização de toda a actividade da instituição.

É para mim claro que no seu presente estado o painel de controlo já é usável e pode trazer valor acrescentado à instituição, por terem sido implementadas as funcionalidades essenciais. Falta no entanto um conjunto de actividades e de funcionalidades que aumentarão consideravelmente a qualidade e a utilidade do módulo. Entre as actividades em falta, saliento:

- teste exaustivo, por parte de um *tester*;
- melhoria da usabilidade, sobretudo ao nível da definição dos indicadores;
- uso em ambiente real e *feedback* de usabilidade;
- *performance tuning*;
- ajudas (há que actualizar as de Janeiro);
- muitos indicadores.

As seguinte funcionalidades não estão totalmente implementadas:

- avisos (parcial);
- relevância (parcial);
- resultados em XML;
- gráfico do histórico (pode assentar no mecanismo de gráficos já existente);
- funcionalidades identificadas tardiamente, como indicadores de ramificação.

Agora que olho para a lista, acho que se tivesse implementado tudo isto era um herói! De qualquer maneira, foi atingido um nível agradável de automatização da produção de indicadores e sua organização.

6.4 Perspectivas

Já combinei com o Eng. Manuel Machado a minha continuidade no CICA por mais uns meses e espero nos avançar nos tempos mais próximos com a implementação e execução da maior parte dos itens das duas listas anteriores.

Creio que o Painel de Controlo tem um enorme potencial porque pode permitir uma análise rigorosa e objectiva do desempenho das instituições de ensino em que seja corrido. Embora o objectivo primário tivesse sido a construção de uma ferramenta de monitorização da qualidade de dados a usar pela Gestão de Dados, procurei generalizar de forma a permitir a construção automática de painéis sobre indicadores já existentes, embora orientados a outros utilizadores, e não necessariamente sobre a qualidade de dados.

Futuramente deve ser equacionada seriamente a construção sistemática de indicadores estratégicos que permitam à gestão da instituição, do curso, do serviço ou do departamento um controlo mais efectivo das actividades e do seu alinhamento com a estratégia da instituição. O trabalho a desenvolver deverá centrar-se na identificação e implementação de um conjunto limitado de indicadores estratégicos (sim, o número é importante: poucos limitam a informação disponível, demasiados poluem-na) adequados a cada *gestor*, e posteriormente na sua manutenção (a estratégia também varia).

Mas sem perder de vista que o objectivo mais imediato é a construção de indicadores de qualidade de dados para a Gestão de Dados.

7 Referências e Bibliografia

Bibliografia

LEIC: FEUP, Engenharia Informática e Computação, 2003,
http://www.fe.up.pt/si/cursos_geral.FormView?P_CUR_SIGLA=LEIC

CICA: CICA, Página do CICA, 2004,
http://www.fe.up.pt/si/web_base.gera_pagina?P_pagina=2406

FEUP: FEUP, Página da Faculdade de Engenharia da Universidade do Porto, 2005,
<http://www.fe.up.pt>

SIG: FEUP e IRICUP, Sobre o Sistema de Informação, ,
http://www.fe.up.pt/si/web_base.gera_pagina?P_pagina=2403

SIG2: CICA e IRICUP, SIGARRA - Sistema de Informação Académico,

PLS: John Russell, PL/SQL User's Guide and Reference, 2003

W3C: W3C, World Wide Web Consortium, , <http://w3.org>

XSL: W3C, What is XSL?, , <http://www.w3.org/Style/XSL/WhatIsXSL.html>

Anexo A: Lista resumida de funcionalidades a implementar

A lista seguinte foi útil na monitorização do progresso da implementação.

Código: F001

Nome: Repositório de templates

Os templates são implementados na base de dados, como funções ou procedimentos PL/SQL que possuem um conjunto de parâmetros de adaptação. São armazenados dados descritivos do template e dos parâmetros que permitem adaptá-lo a um uso concreto. Os parâmetros podem ter valores por defeito.

Código: F002

Nome: Repositório de funções e procedimentos

As funções podem ser procedimentos/funções armazenados na base de dados ou procedimento PL/SQL anónimos armazenados na tabela. Podem também ser derivadas de templates. Podem receber parâmetros. As funções podem retornar valores.

Código: F003

Nome: Propósito das funções, dos procedimentos e dos templates

As funções e procedimentos devem ter um uso definido. As possibilidades identificadas são as seguintes: heurística de relevância; heurística de estado; heurística de estado de indicador composto; fórmula de calculo; fórmula de agregação; gráfico.

Código: F004

Nome: Gerir dados de indicador, geral

Introduzir, alterar ou remover indicadores do sistema, de qualquer um dos três tipos já definidos: *elementar*, *composto* e *agregado*. Procurar e listar indicadores.

Código: F005

Nome: Desactivação de indicadores

Indicadores desactivados não serão calculados nem apresentados. Apenas serão tomados em conta nas páginas de administração.

Código: F006

Nome: Gerir parâmetros

Os parâmetros qualificam os resultados de um indicador. Há que definir parâmetros e registar valores e combinações possíveis. Por vezes são produzidos resultados sobre combinações ou valores inexistentes, e é preciso registar os novos valores, numa perspectiva correctiva.

Código: F007

Nome: Parametrização dos indicadores e seus resultados

Associar os parâmetros aos indicadores e seus resultados. Usar os parâmetros e seus valores no acesso aos dados.

Código: F008

Nome: Suavizar apresentação dos valores dos parâmetros

Em geral os valores dos parâmetros são códigos, pouco apropriados para uma apresentação amigável. É preciso arranjar alternativas para apresentação. Há duas hipóteses: carregar os dados de apresentação para a tabela de valores dos parâmetros, ou definir uma query que permita sacar os dados de apresentação a partir dos valores dos parâmetros. A primeira tem a vantagem da eficiência, a segunda tem a desvantagem de ocupar mais espaço e da duplicação de informação.

Código: F009

Nome: Cálculo de indicadores elementares

O cálculo pode ser feito a partir de um procedimento PL/SQL na BD, um procedimento anónimo ou uma query SQL. É preciso definir um mecanismo que possa ser usado por todos os indicadores de maneira uniforme.

Código: F010

Nome: Registo e acesso aos resultados

Após o cálculo, temos normalmente uma quantidade de resultados e informação associada para registar no histórico. Verificar relevância histórica e período de actualização do histórico. Definição da ordem da informação adicional e dos parâmetros.

Código: F011

Nome: Indicadores calculados na hora

Podem haver indicadores cujo cálculo é leve e que devem ser sempre calculados na hora. Analisar implicações nos indicadores agregados e nos indicadores compostos.

Código: F012

Nome: Cálculo de indicadores agregados

O cálculo dos indicadores agregados deve ser feito a partir dos resultados já presentes na base de dados, ou sobre dados calculados na hora.

Código: F013

Nome: Cálculo de estados

O cálculo dos estados dos indicadores elementares faz-se a partir do resultado. O cálculo dos estados dos indicadores compostos faz-se a partir dos estados dos componentes.

Código: F014

Nome: Cálculo automático

O administrador pode definir a frequência e a data/hora em que um indicador deve ser calculado automaticamente. Deve haver um interface amigável para gerir os cálculos automáticos.

Código: F015

Nome: Refrescamentos

O responsável de dados pode, desejando, refrescar o indicador ou painel. Consoante o peso do cálculo este deve ser feito na hora ou deve ser pedido ao utilizador que volte mais tarde. Deve respeitar um período mínimo de frescamento.

Código: F016**Nome:** Explicações

Construir uma explicação textual num formato humano, incluindo muita da informação disponível sobre o indicador em questão. Pode incluir uma descrição do problema, de como foi detectado, uma lista de *links*, ...

Código: F017**Nome:** Registos problemáticos

Nos indicadores elementares, apresentar os registos com problemas. Dependendo do indicador, pode ser uma lista, uma tabela, etc.

Código: F018**Nome:** Composição de painéis e indicadores compostos

Pegar na lista de indicadores disponíveis e adicionar ou remover da lista de componentes do painel ou indicador.

Código: F019**Nome:** Avisos

Perante a obtenção de um registo considerado problemático, devem ser disparados alarmes alertando os responsáveis de dados. O endereçamento deve ser feito por papel.

Código: F020**Nome:** Vista de alertas

O painel de controlo deve ter uma vista de alertas, em que são mostrados todos os indicadores em emergência.

Código: F021**Nome:** Gestão de painéis

Inserção, alteração e remoção de painéis. Definição dos parâmetros de um painel, a sua apresentação e o seu uso no acesso aos dados.

Código: F022**Nome:** Apresentação do painel

Cada painel é acessível apenas a um conjunto de grupos de validação. O painel é essencialmente gráfico. No cabeçalho devem ser apresentados alguns dados do indicador, logo seguidos dos indicadores componentes ou de explicações. A informação deve ser compactada ao máximo.

Código: F023**Nome:** Miniatura

Os componentes de um indicador devem ser mostrados numa forma gráfica miniaturizada.

Código: F024**Nome:** Gráficos

A apresentação dos dados dos indicadores deve ser acompanhada sempre que possível e desejável por gráficos intuitivos. O tipo de gráfico depende do tipo de indicador. Construir um mecanismo de chamada e apresentação de gráficos.

Código: F025

Nome: Barra de estado e resultado

Construção de uma barra horizontal mostrando o estado, o resultado, e as transições de estado.

Código: F026

Nome: Gráfico de barras

Construção de gráfico de barras para mostrar o estado e o resultado dos indicadores.

Código: F027

Nome: Gráfico de bolo

Construção de gráfico de bolo para mostrar o estado e o resultado dos indicadores.

Anexo B: Lista de indicadores analisados

Fichas de disciplina

As fichas de disciplinas são inicializadas pelos *directores de curso*, como consequência da criação da ocorrência da disciplina, após a decisão das *disciplinas* que irão funcionar nesse ano lectivo.

Após a *distribuição de serviço*, os docentes podem concluir o preenchimento das *fichas de disciplina*, que ficam a aguardar aprovação pelo *director de curso* respectivo. O processo termina com a validação das fichas pelo *director de curso*.

Indicadores

Há dois indicadores interessantes a construir. Um indicando a taxa de preenchimento de fichas de disciplina, outro indicando a taxa de preenchimento de cada campo das fichas de disciplina. Estes indicadores devem ser implementados para as fichas em inglês e em português.

Taxa de preenchimento de fichas de disciplina

O que se pretende é calcular, para cada curso, cada ano lectivo e cada período é a percentagem de fichas de disciplina inicializadas. Uma ficha de disciplina é considerada inicializada se o campo *objectivos* estiver preenchido. O período de relevância é, para já, “sempre”.

Como o resultado é uma percentagem, vamos também armazenar um resultado aditivo com o nº total de ocorrências de cada curso, no período analisado.

O cálculo deve ser feito sempre nos períodos correntes, e no ano lectivo corrente, para todas as ocorrências de disciplinas.

O estado pode ser um semáforo com 3 estados dependente do tempo, isto é, o valor das transições verde/amarelo e amarelo/vermelho pode depender do tempo. Há medida que nos aproximamos do início das aulas, os valores críticos podem ir subindo, de forma que uma taxa de 20% a um mês do início das aulas não é grave, mas uma taxa de 90% no dia de início das aulas pode sê-lo.

O resultado deve ser apresentado num gráfico de barras com o valor e marcação das transições verde/amarelo e amarelo/vermelho. A listagem de registos problemáticos de incluir a listagem das ocorrências sem ficha inicializada, com link para a página respectiva.

Sumário

- **tipo:** elementar;
- **natureza:** completude, em percentagem;
- **estado:** semáforo, verde/amarelo/vermelho, com transições dependentes do tempo;
- **relevância:** sempre, para já;
- **gráfico:** barra horizontal;
- **problemas:** listagem de ocorrências sem ficha;
- parâmetros:
 - ano lectivo;
 - período;
 - curso;

- **base de trabalho:**

http://www.fe.up.pt/si/ficha_disciplina_validacao.estado_fichas_disciplina

Taxa de preenchimento de fichas de disciplina em inglês

Em tudo igual ao indicador anterior, excepto no campo analisado, que é *objectivos_ing*.

Campo de ficha de disciplina

O que se pretende é calcular, para cada curso, cada ano lectivo e cada período é a percentagem de fichas de disciplina que têm o campo *XPTO* preenchido. O período de relevância é, para já, “sempre”.

Como o resultado é uma percentagem, vamos também armazenar um resultado aditivo com o nº total de ocorrências de cada curso, no período analisado.

O cálculo deve ser feito sempre nos períodos correntes, e no ano lectivo corrente, para todas as ocorrências de disciplinas.

Para o estado, um semáforo, com verde, amarelo e vermelho.

Há campos mais importantes que outros, e para tratá-los de forma diferente temos de criar um indicador para cada campo. Há uma sobreposição entre o indicador para o campo *objectivos* e o indicador para o campo *objectivos*. Esta aproximação introduz um overhead considerável no tempo de cálculo.

Podemos reaproveitar código definindo um template para a fórmula de cálculo. Este template deve receber como parâmetro o nome do campo cujo preenchimento deve ser verificado.

O resultado deve ser apresentado num gráfico de barras, com a marcação dos valores de transição. A listagem de registos problemáticos deve consistir na lista das disciplinas em que o campo não está preenchido, com link para a página respectiva.

Sumário

- **tipo:** elementar;
- **natureza:** completude, em percentagem;
- **estado:** semáforo, verde/amarelo, com transição dependente do tempo;
- **relevância:** sempre, para já;
- **gráfico:** barra horizontal;
- **registos problemáticos:** Listagem de ocorrências com campo por preencher;
- **parâmetros:**
 - ano lectivo;
 - período;
 - curso;
- **base de trabalho:**

http://www.fe.up.pt/si/ficha_disciplina_validacao.fichas_disciplinas?p_ano_lectivo=2004/2005&p_curso=LEIC

Campo de ficha de disciplina em inglês

Idêntico ao anterior, mas para os campos em inglês.

Campos de fichas de disciplina

Indicador composto. Vai albergar os indicadores sobre os campos de ficha de disciplina.

Sumário

- **tipo:** composto;
- **natureza:** completude, em percentagem;
- **estado:** o pior dos estados dos componentes.

Disciplinas

Indicador composto, no qual vamos incluir todos os indicadores relativos às disciplinas.

Sumário

- **tipo:** composto;
- **natureza:** completude, em percentagem;
- **estado:** o pior dos estados dos componentes.

Sumários

Após a criação da ocorrência, a distribuição de serviço e a geração de horários, o docente pode iniciar as actividades relacionadas com o ensino. Entre estas actividades estão o planeamento das aulas e a escrita de sumários.

Indicadores

Há dois indicadores a implementar. Um indica a percentagem de disciplinas com sumários inicializados. Outro indica a percentagem de disciplinas que têm os sumários inicializados e em dia.

Taxa de preenchimento de sumários

O que se pretende é calcular, para cada curso, cada ano lectivo e cada período é a percentagem de ocorrências com sumários preenchidos. Considera-se que uma ocorrência tem sumários preenchidos quando existir pelo menos um sumário nessa ocorrência. Como o resultado é uma percentagem, vamos também armazenar o nº total de ocorrências de cada curso, no período analisado.

O cálculo deve ser feito sempre nos períodos correntes, e no ano lectivo corrente, para todas as ocorrências de disciplinas.

O estado pode ser um semáforo com 3 estados.

O resultado deve ser apresentado num gráfico de barras com o valor e marcação das transições verde/amarelo e amarelo/vermelho. A listagem de registos problemáticos deve consistir na listagem das ocorrências sem sumários, com link para a página respectiva.

Sumário

- **tipo:** elementar;
- **natureza:** completude, em percentagem;
- **estado:** semáforo, verde/amarelo/vermelho;
- **relevância:** a partir do fim do período de tolerância após o início das aulas, e até ao fim do período;
- **gráfico:** barra horizontal;
- **registos problemáticos:** listagem de ocorrências sem sumários;
- **parâmetros:**
 - ano lectivo;
 - período;
 - curso;
- **base de trabalho:** http://www.fe.up.pt/si/sumarios_adm.sum_falta

Taxa de sumários em dia

O que se pretende é calcular, para cada curso, cada ano lectivo e cada período é a percentagem de ocorrências com sumários inicializados que estão em dia. Há um prazo de tolerância no preenchimento de sumários, e considera-se que há sumários em atraso se houver sumários por preencher há mais tempo que esse prazo. Como o resultado é uma percentagem, guardamos também o nº total de ocorrências de cada curso com os sumários inicializados, no período analisado.

O cálculo deve ser feito sempre nos períodos correntes, e no ano lectivo corrente, para todas as ocorrências de disciplinas.

O estado pode ser um semáforo com 3 estados.

O resultado deve ser apresentado através de um gráfico de barras com o valor e marcação das transições verde/amarelo e amarelo/vermelho. Deve haver uma listagem das ocorrências sem sumários, com link para a página respectiva.

Sumário

- **tipo:** elementar;
- **natureza:** completude, em percentagem;
- **estado:** semáforo, verde/amarelo/vermelho;
- **relevância:** a partir do fim do período de tolerância após o início das aulas, e até ao fim do período;
- **gráfico:** barra horizontal;
- **registos problemáticos:** listagem de ocorrências com sumários atrasados e data do último sumário;
- **parâmetros:**
 - ano lectivo;
 - período;
 - curso;
- **base de trabalho:** http://www.fe.up.pt/si/sumarios_adm.sum_falta

Distribuição de serviço

Depois de ser decidido qual o plano de estudos para um ano lectivo, os directores de curso criam uma nova ocorrência das disciplinas para esse ano lectivo.

Depois é necessário definir as características das disciplinas, definindo os directores de curso o número de turmas de que a disciplina necessita e solicitando o serviço a um departamento ou secção.

A alocação de docentes é da competência dos directores de departamento, que verificam a lista de docentes disponíveis e analisam a lista de distribuição de serviço do ano lectivo anterior. Se decidirem manter a distribuição de serviço do ano anterior copiam para o ano lectivo actual, senão têm que efectuar a partir do zero.

A direcção da faculdade tem que analisar a distribuição de serviço efectuada em cada departamento, podendo aprovar ou rejeitar a distribuição de serviço efectuada.

Caso seja aprovada o procedimento termina, caso contrário terá que ser proposta nova distribuição de serviço até à obtenção da aprovação final.

Indicadores

Os indicadores desta secção diferem dos anteriores em natureza, o que é bom para testar a robustez do sistema. Indicadores a implementar:

- disciplinas sem distribuição de serviço;
- disciplinas com défice/excesso horário relativamente ao planeado;
- verificar se há distribuição de serviço a não docentes;
- verificar se há distribuição de serviço a funcionários inactivos.

Disciplinas sem distribuição de serviço

Contar, para cada curso, em cada ano lectivo, as ocorrências inicializadas mas sem distribuição de serviço.

O estado será binário, indicando a existência ou ausência de problemas. O cálculo do indicador será relevante desde o final do planeamento de ocorrências até ao final do ano lectivo ou até todas as ocorrências terem distribuição de serviço.

Sumário

- **tipo:** elementar;
- **natureza:** completude, nº de registos incompletos;
- **estado:** correcção correcto/incorrecto;
- **relevância:** desde o final do planeamento de ocorrências até ao final do ano lectivo;
- **registos problemáticos:** listagem de ocorrências sem distribuição de serviço;
- **parâmetros:**
 - ano lectivo;
 - período;
 - curso;

- **base de trabalho:**

http://www.fe.up.pt/si/ds.lista_plano?p_pe=348&p_ano_lectivo=2004

Disciplinas com défice/excesso horário relativamente ao planeado

Os processo de distribuição de serviço de geração de horários são diferentes e podem ocorrer incoerências. Determinar as incoerências horárias entre os dados dos horários e os dados da distribuição de serviço, por curso, ano lectivo e período.

O estado será binário, indicando a existência ou não de incoerências. É relevante desde a publicação do horário até ao final do período.

Sumário

- **tipo:** elementar;
- **natureza:** coerência, nº de incoerências registadas;
- **estado:** correcção, correcto/incorrecto;
- **relevância:** desde a publicação do horário até ao final do período;
- **registos problemáticos:** listagem de incoerências ordenadas por disciplina, incluindo o nº de horas na planeadas e o número de horas distribuídas;
- **parâmetros:**
 - ano lectivo;
 - período;
 - curso;

- **base de trabalho:**

http://www.fe.up.pt/si/ds_relatorios.ds_horarios?p_periodo=1S&p_cur_sigla=LEIC&P_PE=348&p_ano_lectivo=2004/2005

Verificar se há distribuição de serviço a não docentes

Na distribuição de serviço pode haver distribuição de aulas a pessoal não docente. Este indicador deve procurar contabilizar essas atribuições, contando o número de turmas atribuídas a pessoal não docente.

O estado será binário, indicando a existência ou não de distribuição de serviço a funcionários. É relevante desde a publicação da distribuição de serviço até ao final do ano lectivo.

Sumário

- **tipo:** elementar;
- **natureza:** validação, o número de turmas atribuídas a pessoal não docente;
- **estado:** validação, há incorrecções/não há;
- **relevância:** desde a publicação da distribuição de serviço até ao final do ano lectivo;
- **registos problemáticos:** listagem de turmas e funcionários atribuídos, agrupados por disciplinas;
- **parâmetros:**
 - ano lectivo;
 - período
 - departamento ou secção.

Verificar se há distribuição de serviço a funcionários inactivos

Por vezes acontece com docentes com contrato a termo certo. Os directores de departamento pretendem a continuação do docente, e contam com ele na distribuição de serviço, mas na secretaria o docente é “abatido” logo após o termo do contrato.

Detectar estas situações de forma a poder exigir uma correcção rápida.

O estado será binário, indicando a existência ou não destas situações. É relevante desde a publicação da distribuição de serviço até ao final do ano lectivo.

Sumário

- **tipo:** elementar;
- **natureza:** validação, o número de docentes inactivos com turmas atribuídas;
- **estado:** validação, há incorrecções/não há;
- **relevância:** desde a publicação da distribuição de serviço até ao final do ano lectivo;
- **registos problemáticos:** listagem de funcionários inactivos com serviço e respectiva identificação;
- **parâmetros:**
 - ano lectivo;
 - período;
 - departamento ou secção.

Anexo C: Sintaxe da explicação

A explicação textual é especificada num formato XML. Como todos os documentos XML, a explicação deve ter uma raiz, e por isso o conteúdo deve ser colocado dentro da marca **explicacao**.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<pce:explicacao xmlns:pce="http://www.fe.up.pt/pcontrol/explicacao">
</pce:explicacao>
```

Podemos incluir texto ou HTML, e onde for conveniente podemos colocar marcas especiais que serão substituídas por informação na hora da execução. Estão disponíveis as seguintes marcas:

- **param**
 - substitui a ocorrência da marca pelo valor do parâmetro;
 - exemplo:
 - O ano lectivo de <pce:param>P_ANO_LLECTIVO</pce:param>;
 - O ano lectivo de 2004/2005;
- **resultado**
 - substitui a ocorrência da marca pelo resultado do indicador; aceita 2 parâmetros:
 - *factor*, um número que multiplica pelo valor; por defeito é 1;
 - *round*, o número de casas decimais; o resultado é arredondado; por defeito é 0;
 - exemplo:
 - <pce:resultado factor="100" round="2" />% . não têm ficha de disciplina;
 - sendo o resultado 0,666666 é substituído por: 66,67% não têm ficha de disciplina;
- **resultado-adit**
 - substitui a ocorrência da marca pelo resultado aditivo do indicador; aceita 2 parâmetros:
 - *factor*, um número que multiplica pelo valor;
 - *round*, o número de casas decimais; o resultado é arredondado;
 - exemplo:
 - <pce:resultado-adit round="2" /> fichas;
 - sendo o resultado 75 é substituído por: 75 fichas;
- **estado**
 - substitui a ocorrência da marca pelo valor do estado;
- **codigo**
 - permite executar código PL/SQL e colar o resultado na página;
 - exemplo:
 - $2+2 = \text{sqrt}(16)$ equivalente a <pce:codigo>2+2</pce:codigo> = <pce:codigo>sqrt(16)</pce:codigo>;
 - $2+2 = \text{sqrt}(16)$ equivalente a $2+2 = 4$;
- **func**
 - permite chamar uma função PL/SQL; pode ser substituída por uma chamada a código; tem 1 parâmetro:
 - *nome*, o nome da função;
 - pode conter filhos:

- **p**
 - um parâmetro da função;
- exemplo:
 - sendo o resultado 16, `<pce:func nome="sqrt"><pce:p>
<pce:resultado /></pce:p></pce:func>`;
 - sendo o resultado 16, 4;

Anexo D: Sintaxe da apresentação da página

A apresentação dos dados opcionais da página é especificada num formato HTML. A raiz do documento é uma marca **maxi**.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<pc:maxi xmlns:pc="http://www.fe.up.pt/pcontrol">
</pc:maxi>
```

Podemos incluir texto ou HTML, e onde for conveniente podemos colocar marcas especiais que serão substituídas por informação na hora da execução. Estão disponíveis as seguintes marcas:

- **graf_res**
 - apresenta um gráfico com o resultado, o estado, e as transições de estado;
 - parâmetros:
 - *larg*, a largura do gráfico;
 - *alt*, a altura do gráfico;
 - *tipo*, o tipo de gráfico:
 - *barra_hor*, barra horizontal;
 - ...
 - um *factor* que multiplica pelo valor; útil para apresentar percentagens; valor por defeito é 1;
 - a *unidade* do resultado; nulo por defeito;
 - *max*, o valor máximo visível no gráfico;
 - *min*, o valor mínimo visível no gráfico;
- **graf_contex**
 - apresenta um gráfico contextual de tipo vulgar;
 - parâmetros:
 - *larg*, a largura do gráfico;
 - *alt*, a altura do gráfico;
 - *tipo*, o tipo de gráfico:
 - *barra_hor*, barra horizontal;
 - ...
 - um *factor* que multiplica pelo valor; útil para apresentar percentagens; valor por defeito é 1;
 - a *unidade* do resultado; nulo por defeito;
 - *max*, o valor máximo visível no gráfico;
 - *min*, o valor mínimo visível no gráfico;
- **graf_def** (não implementado)
 - apresenta um gráfico contextual de próprio do indicador em questão;
- **avisos** (não implementado)
 - apresenta a lista de responsáveis a avisar;
- **links** (não implementado)
 - apresenta uma lista de links determinada a partir do registo de processos;
- **explicacao**
 - mostra a explicação textual
- **lista_reg**
 - mostra os registos problemáticos;

- parâmetros:
 - *titulo*, se mostra ou não o título:
 - *sim* (valor por defeito);
 - *não*;
- **folha**
 - mostra uma tabela com informação geral sobre o indicador;
 - parâmetros:
 - *larg*, a largura da tabela;
 - filhos:
 - **descricao**
 - mostrar a descrição do indicador;
 - **params**
 - mostrar a lista de parâmetros e seus valores;
 - **instante**
 - mostrar o instante de cálculo;
 - **titulo**
 - parâmetros:
 - *tipo*:
 - *sim*;
 - *contextual* (valor por defeito);
 - **graf_res**
 - igual ao **graf_res** acima;
 - **resultado**
 - igual ao **resultado** acima;
 - **explicacao**
 - igual à **explicacao** acima



Anexo E: Sintaxe da apresentação das miniaturas

A apresentação dos dados opcionais da página é especificada num formato HTML. A raiz do documento é uma marca **mini**.

```
<?xml version="1.0" encoding="ISO-8859-1"?>  
<pc:maxi xmlns:pc="http://www.fe.up.pt/pcontrol">  
</pc:maxi>
```

A vista mini suporta os mesmos componentes que a vista maxi, acrescida de um:

- **componentes**
 - mostra a uma lista de componentes e respectivos estados.



FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

BIBLIOTECA



0000081554