

NetTrouble: Um TTS Flexível, Distribuído e Aberto

Luís Santos, Pedro Costa, Paulo Simões
Grupo de Comunicações e Serviços Telemáticos
Departamento de Engenharia Informática
Universidade de Coimbra - Portugal
3030 Coimbra

Email: lsantos@dei.uc.pt, p-jose-costa@telecom.pt, psimoes@dei.uc.pt

Palavras-Chave: Gestão de Redes, Sistemas de Acompanhamento de Problemas

Resumo

Os actuais sistemas de informação necessitam de ferramentas capazes de registar os diversos problemas ocorridos no sistema, coordenar os vários esforços de resolução desses problemas, e armazenar a experiência adquirida com a sua resolução. Essas aplicações são tradicionalmente designadas por Sistemas de Acompanhamento de Problemas ou *Trouble Ticket Systems* (TTS). Basicamente, pede-se aos TTS que funcionem como mapas hospitalares, “coordenando o trabalho de várias entidades (pessoas/grupos/organizações) envolvidas em cada problema e funcionando como bases de conhecimento de problemas passados”[1].

Existe neste momento um grande número de ferramentas enquadráveis nesta definição de TTS. Estas são habitualmente desenvolvidas para ambientes específicos (desenvolvimento de projectos, administração de *hosts*, manutenção de queixas, etc.) tendo acoplados, por essa razão, um modelo de informação e um modelo administrativo que, porque demasiado específicos, se mostram pouco flexíveis noutros ambientes. Talvez por isso elas ainda não consigam resultados completamente satisfatórios na área de gestão de sistemas e redes, apesar dos bons níveis de desempenho obtidos em outros ambientes. Entre as razões para esta situação, destacam-se a grande heterogeneidade das redes de dados, a sua disseminação geográfica (característica predominante em organizações de média e grande dimensão), o carácter multi-organização de certas redes de área alargada (com diversos fornecedores de serviços de comunicações entre os dois extremos da rede), e a decomposição hierárquica ou funcional das tarefas e responsabilidades de gestão.

Este artigo apresenta os principais resultados de um projecto desenvolvido na Universidade de Coimbra com o objectivo de inverter tal situação. O referido projecto envolveu a concepção e implementação do *NetTrouble*: um TTS que integra conceitos e características inovadoras, designadamente o conceito de domínio administrativo (como resposta ao carácter multi-organização do sistema), a autonomia na definição de políticas de cooperação entre quaisquer pares de domínios, um modelo administrativo flexível (com granularidade até ao nível do problema, com a possibilidade de envolver recursos humanos locais e remotos), e a virtualização no TTS local de problemas remotos em que o domínio é parte interessada. Esta virtualização permite o acesso descentralizado à informação e a capacidade de aceder a um TTS específico a partir da rede, aumentando a acessibilidade da informação e possibilitando a interacção com eventuais aplicações remotas. Merece ainda destaque o esquema de notificações desenvolvido, em que ao registo de cada problema (*ticket*) surgem associados a noção de *timeout* e um mecanismo de comunicação por *e-mail* mais sofisticado que o habitual.

1. Introdução

O conceito de TTS remonta a meados da década de oitenta, com referências a sistemas compostos por uma base de dados de grandes dimensões destinada às funções de atribuição, calendarização e armazenamento de tarefas. O objectivo de tais sistemas concentrava-se essencialmente no fomento de simplicidade e eficiência na coordenação de grupos de trabalho. Actualmente, no contexto dos sistemas de informação, o conceito e os objectivos subjacentes aos TTS permanecem praticamente inalterados. A evolução existente tem que ver predominantemente com a diversidade de campos de aplicação que acolheram este conceito e com as tecnologias usadas na sua implementação.

O objectivo do projecto aqui descrito centrou-se na procura de um sistema TTS que se adequasse às necessidades de um ambiente de gestão de sistemas e redes. Foram estudadas, por essa razão, as principais ferramentas de acompanhamento de problemas, o que permitiu, por um lado detectar diversos aspectos comuns às mesmas, por outro avaliar o nível de adequação destas ao ambiente alvo.

1.1. Sistemas actuais

Apesar de se estar perante uma área onde ainda não se encontram definidas fortes linhas de orientação, é possível identificar certos pontos comuns que de alguma forma caracterizam um TTS.

1.1.1. Estrutura

A estrutura de um TTS (fig. 1) comporta, em geral, três módulos :

- uma base de dados usada para o armazenamento de *trouble tickets*;
- um módulo responsável pela mediação entre a base de dados e os utilizadores finais;
- e o interface com o utilizador que fornece o ponto de acesso ao sistema.

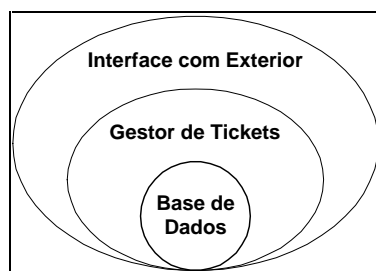


Figura 1: Estrutura Típica de um TTS

É no módulo intermédio que se concentram as unidades mais críticas do sistema: o modelo administrativo e o modelo funcional. Eles regem a interacção entre a entidade principal - o *ticket* - e os diversos utilizadores do sistema: o primeiro definindo as formas de intervenção possíveis a cada membro da equipa, o segundo impondo etapas coerentes no processo de resolução do mesmo.

1.1.2. Informação e Fluxo

A base de dados presente em todos os sistemas é invariavelmente centralizada, fechando-os sobre si mesmo. De frisar, porém, que nalguns pacotes encontrámos o módulo de interface com o utilizador desacoplado do restante sistema, permitindo que o acesso a este se fizesse por intermédio da rede local [2] [6] [4], ou até do sistema de correio electrónico [3]. A entidade central ao nível da informação armazenada é o *trouble ticket*. A estrutura deste é em geral comum, apresentando-se dividida em dois grandes campos: o cabeçalho e o corpo. O primeiro contém atributos bem definidos do ambiente servido, o segundo destina-se ao armazenamento

flexível de informação que o problema deve transportar. Também ao nível do fluxo de informação é possível delinear analogias nos sistemas actuais: os *trouble tickets* são registados no sistema, sendo-lhes vinculados recursos humanos. O sistema toma então a responsabilidade de facilitar a sua coordenação, processo que converge para a resolução do problema em causa. Neste ponto verifica-se a sua eliminação da base de dados (caso mais frequente), ou o armazenamento da experiência adquirida - em soluções mais sofisticadas.

1.2. Condicionantes de um ambiente de gestão de redes

Caracterizadas que estão as soluções existentes importa agora ponderar as peculiaridades que revestem os NOC¹, para assim poder decidir da validade e adequação dos conceitos e soluções actuais no ambiente particular da gestão de redes. Johnson [1] resume no RFC 1297 uma panóplia de requisitos a que um TTS orientado para NOCs deve obedecer. De forma geral, salienta a integração que tal sistema deveria possuir com as restantes ferramentas ao dispor do gestor e discrimina algumas das características desejáveis nos modelos administrativo, funcional e de informação associados (em especial neste último).

Numa primeira abordagem, apenas a baixo nível se encontra uma inadequação entre as soluções actuais e os requisitos referidos. Por outras palavras, bastaria em grande parte adequar os subsistemas existentes, tirando no entanto completo partido dos conceitos subjacentes. Porém, como será evidenciado, tal abordagem é manifestamente insuficiente.

São seguidamente apresentados determinados factores que se revelarão marcantes na abordagem do ambiente característico da gestão de uma rede de dados.

1.2.1. Disseminação geográfica

Poder-se-ia supor que as aplicações já existentes que apresentam uma separação física entre o módulo de interface com o utilizador e o TTS propriamente dito se adaptam de forma satisfatória à disseminação geográfica que as médias e grandes organizações comumente assumem (fig. 2) [2,6].

No entanto, o sistema TTS teria obviamente que residir numa das filiais (preferencialmente aquela que fosse mais central), dado o facto de a sua base de dados ser centralizada. Desta forma, as restantes filiais seriam prejudicadas em termos de tempo de acesso à informação. Haveria também latência e sobrecarga, injustificadas nas comunicações relativas a problemas locais a cada filial.

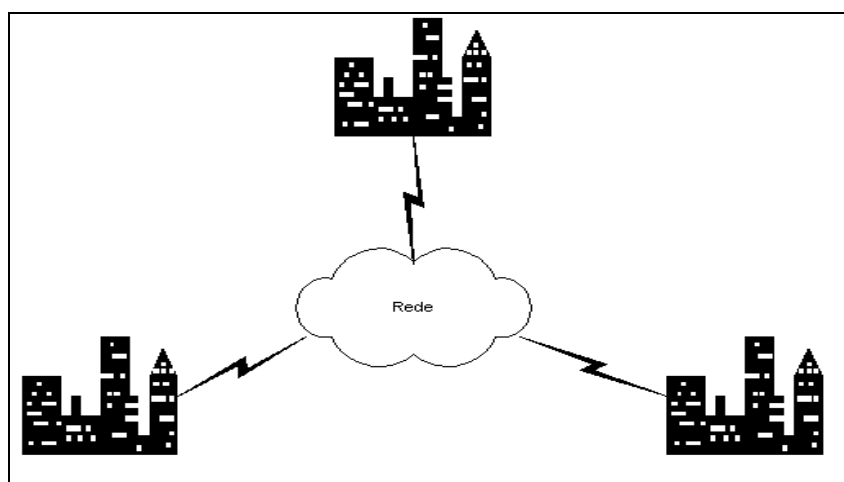


Figura 2: Disseminação Geográfica das Redes de Dados

¹ Network Operation Center

1.2.2. Carácter multi-organização

A um nível mais genérico, o carácter multi-organização de certas redes de área alargada, em que participam diversos fornecedores de serviço entre os dois extremos da rede, torna necessários novos conceitos no domínio dos TTS. Da responsabilidade circunscrita a determinada organização, passamos para uma interdependência entre várias organizações, onde a comunicação entre estas e a co-responsabilização dos problemas é condição fundamental para a sua resolução. Salienta-se assim a carência de mecanismos que viabilizem a comunicação entre TTS de diferentes organizações.

1.2.3. Decomposição hierárquica e funcional

Em organizações de dimensão considerável as tarefas de gestão de redes e sistemas encontram-se em geral consignadas a mais que um gestor, sendo a diversidade e complexidade dessas tarefas as razões óbvias para tal situação. Tal como sucede noutros ambientes, a fim de maximizar a eficiência do grupo, procede-se à sua divisão funcional adoptando habitualmente uma estrutura hierárquica de responsabilidades. Assim, será de todo útil que o sistema de acompanhamento de problemas utilizado inclua mecanismos nativos capazes de acomodar tais características. A este nível as soluções ainda são escassas, pecando as existentes por se mostrarem pouco flexíveis às diferentes realidades com que um ambiente desta natureza se pode confrontar.

1.3. A urgência de uma nova abordagem

Pode-se desta forma intuir que os conceitos que alicerçam os TTS actuais se mostram insuficientes perante as condicionantes atrás descritas, havendo portanto manifesta necessidade de uma abordagem inovadora neste campo.

Para solucionar o problema levantado pela disseminação geográfica foi adoptado o conceito de base de dados distribuída, procedendo a uma descentralização da informação. Quanto à segunda questão - carácter multi-organização - foi definido o conceito de domínio administrativo. Dada a evidente necessidade de uma especificação original, houve também a preocupação, relativamente aos conceitos tradicionais, de respeitar o espírito dos requisitos enunciados por [1], sem no entanto deixar de proceder a algumas inovações úteis (e.g., o conceito de categoria e o seu aproveitamento administrativo).

São estas e outras características do NetTrouble que, elegidas como as mais inovadoras, serão apresentadas seguidamente.

2. Principais características do NetTrouble

Nesta secção serão expostas as ideias chave do NetTrouble, começando por discriminar as características gerais para de seguida destacar questões específicas que, embora não marcando tão fortemente o sistema criado, também apresentam aspectos interessantes e inovadores.

2.1. Aspectos gerais

2.1.1. Domínio Administrativo

Como já foi frisado, o ambiente de gestão de redes reveste-se quase sempre de um carácter multi-organização. Como solução surge o conceito de **domínio administrativo** (fig. 3) capaz de mediar as relações entre as organizações envolvidas na actividade de gestão de uma rede de dados.

Desta forma, a cada organização estará associado um domínio administrativo (pelo menos), que se concretizará com a presença de um servidor NetTrouble. Cada organização será assim autónoma na definição

de políticas de cooperação para com o exterior, podendo especificar o conjunto de organizações com as quais deseja cooperar e, relativamente a cada uma das eleitas, decidir que informação partilhar e que recursos humanos disponibilizar para a resolução de problemas conjuntos.

Apesar de este conceito surgir como solução administrativa para o inter-relacionamento entre organizações com interesses distintos, é também viável o seu uso dentro de uma mesma organização com o fim de reflectir células desta que permaneçam geográfica ou administrativamente separadas. Pode eventualmente ser também utilizado para reflectir níveis hierárquicos de competências na gestão de grandes redes metropolitanas.

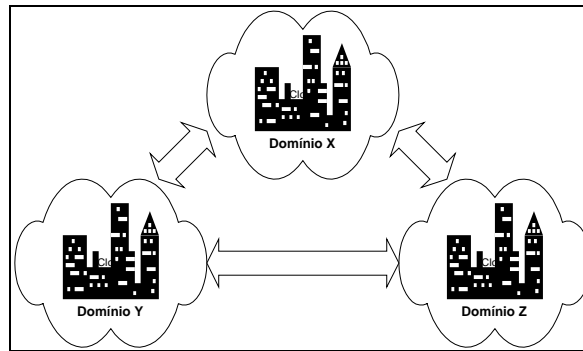


Figura 3: Domínios Administrativos

2.1.2. Arquitectura cliente-servidor e protocolo de comunicação

Surge então a necessidade de estabelecer canais e métodos bem definidos que facultem a comunicação entre quaisquer pares de organizações. Com base no tradicional modelo cliente-servidor, foram concentradas no servidor as responsabilidades de representação de determinado domínio administrativo (mecanismos de segurança no acesso à informação, interface com a base de dados subjacente, mecanismos de *forward*,...). O cliente será qualquer entidade, algures na rede, que pretenda obter e/ou manipular *trouble tickets* de determinada organização. Ao utilizar esta arquitectura incorre-se ainda noutra vantagem: o sistema NetTrouble passa a estar contactável de qualquer ponto da rede, aumentando assim a sua acessibilidade.

Com o objectivo de tornar o NetTrouble uma solução aberta foi ainda especificado um protocolo de comunicação² bem definido de que constam apenas duas primitivas: *data_request*, para obter informação, e *data_set*, para alterar informação.

2.1.3. Descentralização da informação

O conjunto de servidores NetTrouble forma então uma teia onde cada nodo terá completa autonomia para especificar que *trouble tickets* e recursos humanos pretende partilhar com determinada organização. A nível físico cada *ticket* residirá num único domínio administrativo, no entanto, se aquele envolver mais que uma organização (facto comum em gestão de redes) deve, virtualmente, residir em todos os domínios envolvidos, conseguindo-se assim a vantagem de obter em cada organização toda a informação que lhe diga respeito. Esta característica extender-se-à também aos recursos humanos disponíveis. Para determinada organização estes serão constituídos não apenas por gestores locais, mas também por gestores disponibilizados por outras organizações, sendo estes envolvidos na resolução conjunta de problemas comuns a ambas as organizações.

A concretização desta **descentralização de informação e recursos** foi viabilizada pelo mecanismo de *forward* (fig. 4) que seguidamente será descrito.

² Recorrendo-se à tecnologia ONC-RPC [7]

Após ser solicitado, o servidor NetTrouble começa por determinar o tipo de cliente em causa. Tratando-se de um servidor de outro domínio, responde apenas a pedidos sobre dados locais recusando o reencaminhamento desses mesmos pedidos. Por outro lado, se como cliente estiver um gestor local, começa por calcular qual a abrangência do pedido recebido (que domínio(s) envolve), procedendo então, se necessário, ao *forward* do pedido para os domínios necessários (domínio D_2 , ..., domínio D_n), actuando assim como cliente destes. Note-se que o acesso ao mecanismo de reencaminhamento abrange apenas os domínios para os quais o gestor em causa se encontra exportado. Segue-se então o processo de recolha das diversas respostas e a sua concatenação numa única resposta ($\text{resposta} = \text{resposta}_1 + \text{resposta}_2 + \dots + \text{resposta}_n$), que será finalmente devolvida ao gestor local.

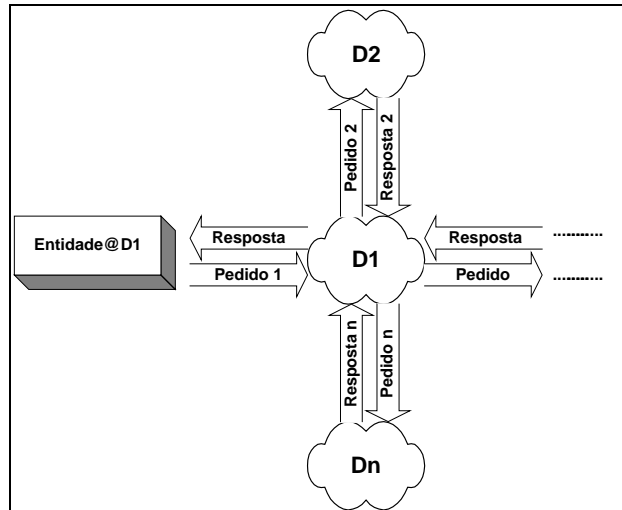


Figura 4: Mecanismo de *Forward*

2.2. Aspectos Específicos

Serão agora descritos os pontos mais relevantes da composição interna do NetTrouble. Em primeiro lugar, serão apresentadas as diferentes entidades com que lida o NetTrouble, sendo de seguida expostos conceitos marcantes associados ao núcleo de informação essencial: o *trouble ticket*. A organização conferida a esta exposição basear-se-á no acompanhamento das diversas etapas a que um *trouble ticket* é submetido.

2.2.1. Entidades do NetTrouble

Num ambiente de gestão de redes, mesmo quando limitado ao interior de uma organização, torna-se habitualmente útil considerar a possibilidade de a gestão ser assegurada por mais que uma pessoa; isto é, assumir um **carácter multi-gestor**. E se em pequenas redes isto parece ser uma boa ideia, para grandes organizações as competências são de tal modo vastas que tal se torna imprescindível. O sistema desenvolvido prevê, por essa razão, a existência de vários gestores por domínio, sendo esta característica suportada em termos funcionais e administrativos por três entidades distintas (fig. 5).

O **gestor** representa um qualquer técnico ao serviço da organização. Será desta entidade que fluirão todas as interacções com o sistema, mesmo que em interacções distribuídas esta origem se confunda com a entidade domínio (devido ao mecanismo de *forward*).

Com o objectivo de simplificar a gestão de cada domínio *per si*, o NetTrouble fornece ainda um mecanismo de agrupamento desses mesmos gestores materializado na entidade **grupo de gestores**. É assim possível agrupar sob uma única identificação gestores individuais ou mesmo outros grupos de gestores, conseguindo deste modo delegar competências e responsabilidades ao nível do grupo.

Por último surge a entidade **domínio**, que pretende materializar o conceito de domínio administrativo, podendo contudo ser vista como um grupo de gestores que se identifica com todos os técnicos de determinada organização. De frisar a participação desta entidade no mecanismo de *forward*. Um pedido que seja reencaminhado assumirá como requisitante não o gestor local, mas sim o domínio local, evitando deste modo que todos os gestores devam estar registados em todos os domínios para deles obterem informação. Basta assim que a entidade que representa o domínio local esteja registada num domínio remoto para que todos os gestores locais exportados para esse domínio tenham acesso à informação disponibilizada por essa organização.

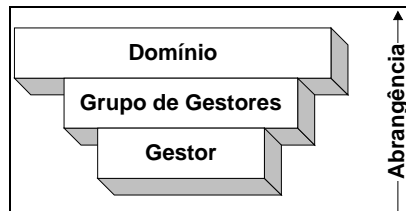


Figura 5: Entidas Administrativas do NetTrouble

2.2.2.O tratamento dos problemas

Inerente a todos os sistemas de acompanhamento de problemas está uma base de dados em contínuo crescimento onde se encontra informação sobre propostas de problemas, problemas correntemente em resolução (*trouble tickets*) e, possivelmente, problemas já resolvidos (servindo estes últimos como uma base de conhecimento para problemas futuros).

Ao nível funcional, a evolução de um problema apresentado ao NOC pode ser sintetizada pelo diagrama da figura 6, existindo diferentes estádios de evolução de forma a reflectir as diferentes exigências que as etapas de resolução de um problema apresentam, tanto ao nível de informação como em termos de recursos humanos. Será agora exposto um conceito fundamental do NetTrouble, as **categorias**, para de seguida se fundamentar a existência das diversas fases mencionadas no diagrama.

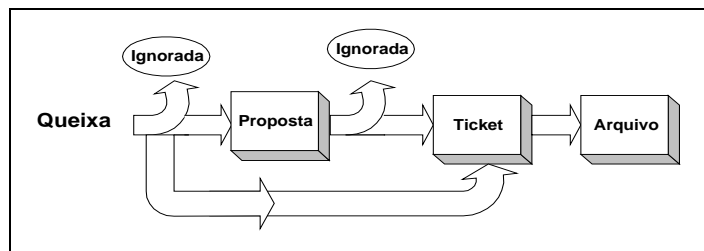


Figura 6: Evolução de um Problema no NetTrouble

2.2.2.1. Categorias

Dos TTS analisados, poucos fornecem a capacidade de categorização de problemas [2] assegurada pelo NetTrouble. Esta classificação implica que, no momento de criação de um problema, este tenha que ser enquadrado numa das categorias existentes, organizando logo à partida a informação existente. O esquema de categorias inerente ao NetTrouble possui uma estrutura hierárquica (em forma de árvore invertida) que permite a organização das próprias categorias e, de certo modo, reflecte a classificação habitual de problemas consoante o seu tipo e especificidade (fig. 7).

O NetTrouble foi ainda mais longe no aproveitamento deste conceito: entregou a gestão de uma categoria a um gestor local dito, neste contexto, **responsável pela categoria**. Obviamente, um gestor poderá ter a seu cargo mais que uma categoria, materializando estas o seu espaço de responsabilidades na tarefa de gestão. Conferiu-se ainda a capacidade de manipular esse mesmo espaço de trabalho, permitindo a sua adequação às

necessidades particulares de determinado gestor, quer por derivação de novas categorias, quer através da designação de novos responsáveis para essas sub-categorias.

Fazendo uso destes mecanismos, e em relação à gestão/acesso de informação sobre problemas, é facilmente criada uma estrutura que reflecta o esquema hierárquico de delegação de responsabilidades frequente num vasto leque de organizações.

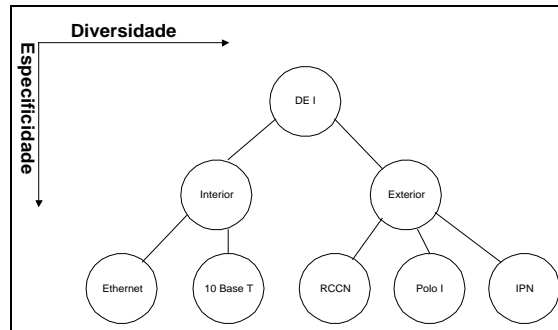


Figura 7: Hierarquização das Categorias de Problemas

2.2.2.2. A proposta

O surgimento de um problema é geralmente detectado pela presença de avarias, deficiências ou carências na funcionalidade de qualquer objecto de gestão (um troço de rede, um *router*, uma impressora, um servidor, uma aplicação,...) [1]. Assim, o problema chega ao NOC, na maioria dos casos, sob a forma de uma queixa. Para o NetTrouble esta queixa chamar-se-á “**proposta**”, uma vez que potencialmente virá a constituir um verdadeiro *trouble ticket*, muito embora ainda não seja tratada como tal. Fica assim viabilizado o registo temporário de diversos sintomas que, possivelmente, desencadearão um único *ticket* [1]. De frisar que a única entidade afectada à queixa será o gestor da categoria à qual esta foi vinculada.

2.2.2.3. O Trouble Ticket

O *trouble ticket*, ou simplesmente *ticket*, representará o estágio de evolução onde decorrerá a resolução do problema propriamente dita. Será portanto nesta etapa que haverá maiores necessidades a todos os níveis: desde a quantidade de informação a armazenar e gerir, até ao assegurar de credibilidade a todo o processo, passando pela necessidade de afectação e coordenação de recursos humanos associados, existirá um sem número de questões a ter em conta.

São de seguida analisados, com a profundidade possível, os aspectos mais marcantes desta fase.

Informação suportada pelo ticket

Tanto a maior parte dos TTSs analisados como [1] defendem que a informação associada ao *ticket* deve ser dividida em cabeçalho e corpo. O **cabeçalho** deve possuir um conjunto de campos fixos com o intuito de agrupar os aspectos bem definidos e comuns a todos os *tickets* e permitir a geração de informação estatística e implementação de procuras mais simplificadas. Já a **descrição** (ou o corpo) deve ter um formato flexível, onde seja possível descrever livremente toda a evolução da resolução do problema.

Também o NetTrouble adoptou este formato, reservando para o cabeçalho campos como o identificador do *ticket*, o título, a categoria, o estado e a prioridade. Ao nível da descrição, o sistema desenvolvido interpreta cada contribuição como uma parte distinta da descrição global, apesar de fornecer um espaço de formato livre. Desta forma associa a cada parcela, de modo automático, a assinatura do gestor contribuinte (reforçando a credibilidade do processo), bem como um "carimbo horário". De salientar que o próprio NetTrouble pode gerar contribuições, facto que sucede quando os campos relevantes do cabeçalho sofrem alterações (aumentando mais uma vez a credibilidade do processo).

Classes associadas

O acesso de determinada entidade à informação associada a um *ticket* é condicionado por dois mecanismos: o uso de classes e o uso de máscaras, servindo estas para estender o alcance daquelas. Uma entidade com acesso a um *ticket* encontra-se vinculada a este por pertencer a uma das quatro classes (fig. 8) previstas pelo NetTrouble.

A classe de **responsável** representa o coordenador da equipa de trabalho do *ticket* (implicitamente o gestor responsável pela categoria a que o *ticket* pertence). Ao responsável são associadas as seguintes competências: gestão de informação do *ticket*; manutenção dos recursos humanos afectos ao mesmo; e a definição de políticas de acesso das restantes classes aos diversos campos do *ticket*. A classe **envolvidos** representa o grupo de trabalho seleccionado pelo responsável para solucionar o problema em questão. Este grupo engloba gestores e/ou grupos de gestores locais ou remotos. Esta classe possui pleno acesso de leitura a toda a informação do *ticket*, podendo ainda participar na sua descrição. O acesso à alteração de campos do cabeçalho (e mesmo à gestão de classe inferiores) pode ser cedido pelo responsável, por recurso a máscaras. A classe **audiência** nasceu da necessidade de tornar a informação de um *ticket* acessível somente para consulta (para determinadas entidades). Por último, a classe **ouvintes** pretende abranger pessoas exteriores ao sistema que, de alguma forma, são afectadas pelo problema em curso. Esta última classe assenta no pressuposto de que tais entidades são contactáveis por correio electrónico.



Figura 8: Hierarquia de Classes

Modelo funcional do ticket

A resolução de um problema processa-se de forma livre. No entanto, o NetTrouble associa ao *ticket* um atributo, o “estado”, a fim de melhor discriminar a situação em que este se encontra (aberto, suspenso, bloqueado, resolvido ou arquivado). Cada um destes estados possui também implicações ao nível da evolução do *ticket* - associando-lhe assim um modelo funcional.

Prioridades associadas e mecanismo de *timeout*

Ao contrário dos sistemas habituais, onde o campo “prioridade” desempenha funções meramente informativas, o NetTrouble permite associar à prioridade do *ticket* um prazo temporal. É assim concretizado o conceito de ***timeout* associado a um *ticket*** [1]. Se durante esse prazo o *ticket* não sofrer nenhuma evolução é enviada uma notificação por *e-mail* ao responsável pelo mesmo.

Mecanismo de vínculos

Uma avaria pode facilmente desencadear reflexos tão distintos que se torna impossível aos gestores detectar, em tempo útil, a ligação entre as diversas queixas que vão recebendo. Torna-se então necessário um mecanismo de suporte que permita estabelecer vínculos entre os diversos *tickets*. No NetTrouble, estão previstos três tipos de vínculos, sendo assim possível que um *ticket*:

- aceite ser consultado pelas entidades associadas a outro;
- expresse a outro a vontade de poder consultar a sua informação;
- ou bloqueie a resolução de outro.

Esta última capacidade permite estabelecer **precedências** na resolução de problemas, facto que se pode revelar bastante útil evitando que o corpo técnico encarregue da resolução de determinado(s) *ticket(s)* desenvolva esforços desnecessários ou mesmo desconcertados.

Mecanismo de notificações integrado

Algumas das aplicações analisadas, porque destinadas a um ambiente de rede, possuíam um esquema de notificações baseado em correio electrónico. No entanto, esse esquema é sempre apresentado como uma funcionalidade do *browser* associado e não como uma característica intrínseca ao sistema de acompanhamento de problemas. O NetTrouble apresenta, de forma integrada um esquema de notificações que torna possível distribuir determinada mensagem a um dado *ticket*, sendo isso interpretado como a distribuição da mesma mensagem a determinado subconjunto de classes intervenientes no respectivo problema. Concretiza-se assim o conceito de **mail para um ticket** (fig. 9).

Note-se que o NetTrouble junta, como prefixo do assunto, a identificação universal do problema em causa (IdentificadorLocal@Domínio) e, como prefixo da mensagem, a assinatura completa do gestor remetente (dando assim credibilidade ao processo). De frisar ainda que é através deste mecanismo de notificações que se consegue contactar a classe de ouvintes associada a determinado *ticket*.

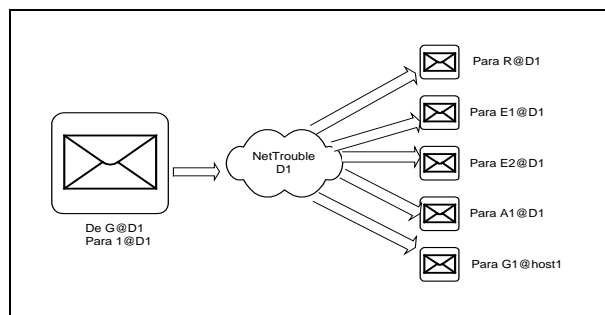


Figura 9: *mail para um Ticket*

2.2.2.4. O arquivo

Um *ticket* termina a sua existência activa quando o seu responsável o faz transitar para o estado *arquivado*, sendo então perdida toda a informação administrativa. No entanto, a sua informação descritiva é retida para tratamento estatístico e futura consulta, funcionando então o arquivo de problemas resolvidos como “base de conhecimento” [1].

3. Campos de aplicação

Com base na filosofia cliente-servidor e no modelo administrativo que o sistema NetTrouble prevê, torna-se possível uma boa adaptação aos moldes em que se realiza a gestão de redes em grande parte das organizações. Com a finalidade de ilustrar de forma mais objectiva as possíveis aplicações do sistema desenvolvido, expõem-se de seguida dois cenários típicos.

3.1. Estrutura Horizontal

Quando entre duas filiais de uma organização se encontram diversos fornecedores de serviço de transporte de dados (fig. 10), a gestão de redes terá que ser assegurada por diversas entidades independentes, que deverão unir esforços para assegurar a eficiente resolução de problemas comuns.

Como mecanismo de comunicação entre as diversas organizações pode ser implantado um sistema servidor NetTrouble em cada uma delas. Cada interveniente terá então oportunidade de disponibilizar recursos humanos ao exterior, abrindo deste modo a possibilidade de duas ou mais organizações procederem, com base num *trouble ticket*, ao tratamento conjunto de determinado problema.

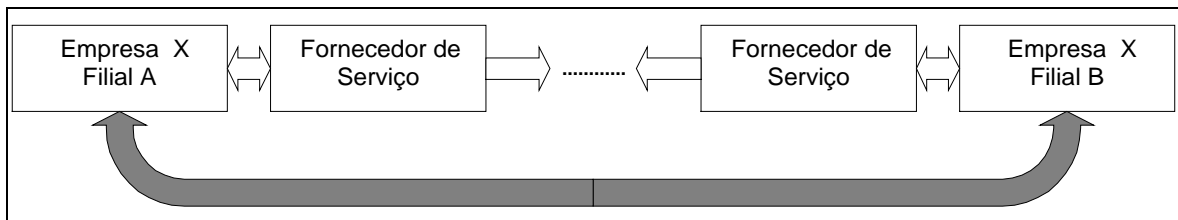


Figura 10: Estrutura Multi-organização Horizontal

3.2. Estrutura Hierárquica

Não menos frequente é o caso em que a gestão de redes assume um carácter hierárquico a nível geral (fig. 11). Também esta situação pode ser enquadrada pelo sistema desenvolvido, implantando um servidor NetTrouble em cada uma das secções. As relações de uma secção com o exterior processam-se por um único canal com o seu nível hierarquicamente superior, e por múltiplos canais com as secções subordinadas.

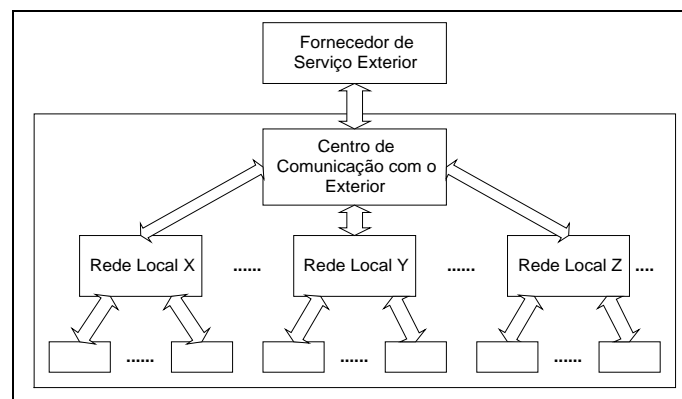


Figura 11: Estrutura Hierárquica

4. Implementação do NetTrouble

A fim de poder concretizar e testar os conceitos e linhas orientadoras apresentados na secção anterior, o NetTrouble foi completamente implementado e disponibilizado ao público [8].

A implementação actual (fig. 12) integra um cliente, um servidor e uma livreria de funções destinada a facilitar o desenvolvimento de novos clientes. O cliente foi desenvolvido em SunOS (ambiente X-Windows/OpenLook) e o servidor funciona em OSF/1, utilizando o Oracle [9] como sistema de gestão de base de dados. A comunicação cliente-servidor recorre à tecnologia ONC-RPC [7], tornando relativamente simples a construção de clientes para outras plataformas.



Figura 12: Aspectos do Interface de Utilizador do Cliente NetTrouble

5. Conclusões e trabalho futuro

Com base na identificação de manifestas insuficiências dos TTS existentes face ao ambiente de gestão de redes, foi apresentada uma nova abordagem concretizada no NetTrouble. Como pilares deste sistema encontram-se o conceito de domínio administrativo; a autonomia na definição de políticas de cooperação entre quaisquer pares de domínios; um modelo administrativo flexível; e o acesso descentralizado, quer à informação, quer aos recursos humanos do sistema global.

Refira-se ainda que as potencialidades do NetTrouble não se esgotam no tratamento de problemas de gestão de redes: ele poderá ser usado como um sistema de *Job Tickets* (em gestão de projectos), ou mesmo como um *News Ticket System*, dada a sua aptidão para organizar discussões temáticas.

A nível conceptual ficam por atingir ainda alguns objectivos que técnicas da área das bases de dados distribuídas permitiriam vulgarizar, tais como a aplicação de vínculos entre problemas residentes em domínios distintos. Seria também importante proceder à especificação (e consequente normalização) de um protocolo aberto para comunicação entre domínios (de forma independente do TTS).

Do ponto de vista técnico, e relativamente à implementação actual, realce-se a utilidade de implementar interfaces em linguagens interpretadas (e.g. Java) com maiores portabilidade e acessibilidade.

6. Agradecimentos

Este trabalho integra-se no projecto FADA, parcialmente financiado pela Junta Nacional de Investigação Científica e Tecnológica (referência PBIC/TIT/2482/95).

7. Referências

- [1] D. Johnson, "NOC Internal Integrated Trouble Ticket System Functional Specification Wishlist (NOC TT Requirements)", *Request For Comments 1297*, 1992.
- [2] "Problem Tracking System 1.05", Zombie Software, 1994.
- [3] "GNU GNATS (GNU Problem Report Management System) 3.2", GNU.
- [4] "Razor 4.0a", Tower Concepts, Inc.
- [5] "Under Control", KJT Software, Inc.
- [6] "PR-Tacker 1.2", Softwise 1994.
- [7] Sun Open Network Computing (SVR4)
- [8] NetTrouble System URL1: <http://cleo.uc.pt/> URL2: <http://www.dei.uc.pt/>
- [9] George Koch, ORACLE7 updates by Robert Muller, "ORACLE7: The Complete Reference", Osborne McGraw-Hill Inc., 1993.

Publication Information:

Luís Santos, Pedro Costa, Paulo Simões, "NetTrouble: Um TTS Flexível, Distribuído e Aberto", Workshop Encontro sobre Redes da FCCN, March 1997, Lisboa, Portugal.