

Comunicação entre Sistemas de Emergência através da Ontologia EDXL-RESCUER

Félix Simas de Souza Neto
Instituto Federal de Educação,
Ciência e Tecnologia da Bahia
Rua Emídio dos Santos, S/N
Barbalho, Salvador, Bahia
Email: felixneto@ifba.edu.br

Renato Lima Novais
Instituto Federal de Educação,
Ciência e Tecnologia da Bahia
Rua Emídio dos Santos, S/N
Barbalho, Salvador, Bahia
Email: renato@ifba.edu.br

Resumo—Sistemas de gerenciamento de emergência são utilizados para auxiliar as forças operacionais durante uma emergência. Porém a existência de um sistema de gerenciamento para cada força operacional pode prejudicar as atividades de gerenciamento. Portanto, para que a atividade de gerenciamento de emergência seja bem sucedido é necessário a interoperabilidade entre as forças operacionais alocadas para tal emergência. Nos últimos três anos, foi desenvolvido um sistema chamado *RESCUER* que utiliza informação da multidão e de forças de segurança para obter informações sobre casos de emergência. Uma das necessidades do *RESCUER* era garantir a interoperabilidade das forças operacionais através da comunicação com outros sistemas de gerenciamento de emergência. Nesse contexto, este trabalho apresenta um componente criado para transferir dados entre sistemas de gerenciamento de emergência com o *RESCUER*. A abordagem utilizada pelo componente LSC (*Legacy System Connector*) baseia-se em uma integração semântica através do suíte de ontologias EDXL-*RESCUER*. A ontologia atua como um vocabulário comum permitindo uma integração entre os sistemas de gerenciamento de emergência.

Keywords—*RESCUER*, *EDXL-RESCUER*, *Ontologia*, *Emergência*, *troca de dados*, *interoperabilidade semântica*

I. INTRODUÇÃO

Emergência é “um evento súbito e geralmente imprevisto que exige medidas imediatas para minimizar suas consequências adversas”[1]. O gerenciamento de emergências constitui a coordenação de atividades envolvidas na preparação, suporte e reconstrução da sociedade quando ocorrem desastres naturais ou provocados pelo homem. O gerenciamento de uma emergência consiste em um conjunto de 4 fases: Mitigação, Preparação, Resposta e Recuperação [2]. Estas fases devem ser conduzidas por forças governamentais locais, estaduais ou federais.

Atualmente existem algumas ferramentas, abordagens e métodos que ajudam nas diferentes fases do gerenciamento de emergência. Entretanto, ainda existe um conjunto de desafios durante o gerenciamento de uma emergência. Um desses desafios é justamente o da comunicação. A falta de comunicação entre as forças operacionais e os civis pode prejudicar a operação de resposta a emergência. Esta falta de comunicação e interoperabilidade pode causar óbitos ou maiores desastres.

Mano e Baker [3] identificaram três categorias de desafios de comunicação: tecnológico, sociológico e organizacional. Após a ocorrência de um desastre, o desafio tecnológico é

a rápida implantação de sistemas de comunicação para os primeiros atendentes e a força operacional que está auxiliando no gerenciamento da emergência. A informação fornecida por testemunhas oculares do desastre é de extrema importância para a força operacional. Tais informações providas podem guiar qual a melhor abordagem para gerenciamento da emergência.

O projeto *RESCUER* [4] (*Reliable and Smart Crowd-Sourcing Solution for Emergency and Crisis Management*) é uma solução inteligente e confiável que coleta informações da multidão para gerenciamento de crises e emergência. Este projeto foi proposto e desenvolvido através de uma parceria entre renomadas instituições de pesquisa e desenvolvimento do Brasil e da União Europeia. Tal projeto viabiliza a comunicação das pessoas com as forças operacionais durante uma situação de emergência

O *RESCUER* utiliza-se de tecnologias atuais como dispositivos inteligentes (por exemplo *smartphones*, *smartwatches*, *drones*) e possivelmente qualquer dispositivo que possa obter uma comunicação utilizando-se da rede de *internet*. O Tratamento de uma emergência normalmente requer coordenação e cooperação dos serviços de organizações externas como também organizações governamentais e a sociedade em geral.

A coordenação e comunicação entre sistemas de gerenciamento de emergência é de crucial importância para um gerenciamento de emergência bem sucedido. Sabendo da existência de outros sistemas que atualmente estão em operação e ajudando as forças governamentais durante situações de emergência, este trabalho desenvolveu um componente dentro do sistema *RESCUER* para permiti-lo comunicar-se com outros sistemas de gerenciamento de emergência.

Atualmente a tecnologia permite diversas maneiras de efetuar a comunicação e integração entre diversos sistemas (por exemplo, requisições REST¹ e SOAP²). A informação trocada entre dispositivos através da *internet*, ou outro meio, é administrada por regras e convenções que podem ser definidas em especificações técnicas, geralmente definidas em protocolos de comunicação.

Com o avanço da tecnologia, melhoria dos protocolos

¹<https://www.w3.org/TR/soap/>

²http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm

e existência de diversos *hardwares*, essa comunicação entre sistemas de informação parece ser um dos menores problemas. Desta forma, carregar os dados entre sistemas através da *internet* é um problema que os protocolos de comunicação pretendem resolver. Porém os dados recebidos têm que ser avaliados no contexto de progresso da conversação entre tais sistemas.

Os sistemas têm que especificar regras para descrever o contexto. Tais regras expressam a sintaxe da comunicação, podendo expressar também o quão significativo os dados podem ser para o contexto quando há troca de informações. Este tipo de regras são definidas para expressar a semântica da comunicação.

Semântica é o ramo que estuda o sentido linguístico e lógico dos termos. A similaridade semântica é uma métrica definida sobre um conjunto de termos onde a ideia da diferença entre eles é baseada no significado dos termos. A integração semântica é o processo de relacionamento entre os termos de diversas fontes que possuem uma similaridade semântica. Ela foca na organização e age como um intermediário entre fontes heterogêneas de dados no qual pode conflitar não somente pela estrutura mas também pelo contexto ou valor dos dados integrados.

A ontologia EDXL-RESCUER fornece uma semântica global para o domínio de gerenciamento e resposta à emergência. Através desta semântica, o componente *Legacy System Connector* (LSC) fornece uma integração entre os sistemas externos que desejem se comunicar com o sistema RESCUER. O LSC atua como facilitador que mapeia as fontes de dados e disponibiliza tais dados na semântica especificada pela ontologia. Cada sistema interessado nas informações provenientes poderão receber os dados na semântica da ontologia.

Este trabalho realizou testes de integração com o sistema RESCUER e o comportamento dos sistemas de gerenciamento de emergência em relação a integração da informação. A definição de um vocabulário comum para sistemas de emergência possibilita a utilização dos constructos³ da linguagem OWL (*Ontology Web Language*). A ontologia EDXL-RESCUER fornece uma semântica comum para o domínio de resposta a emergência e sem ambiguidade para integração entre sistemas de gerenciamento de emergência.

Além desta Introdução, este trabalho está organizado como segue. A Seção II discute os principais tópicos relacionados a pesquisa desenvolvida neste trabalho. A Seção III descreve como foi desenvolvida uma solução para integração de dados entre RESCUER e sistemas legados. A Seção IV demonstra os resultados obtidos na avaliação do RESCUER e do componente LSC. A Seção V trata dos trabalhos relacionados a troca de dados utilizando-se uma ontologia do domínio de gerenciamento de emergência. A Seção VI discorre sobre as lições obtidas no desenvolvimento do componente de integração e os trabalhos futuros.

³Constructos é uma definição utilizada para expressar as construções feitas através de elementos de uma linguagem. A linguagem utilizada para a criação da ontologia EDXL-RESCUER é a OWL *Ontology Web Language*. A linguagem OWL foi projetada para representar ricos e complexos conhecimentos em diversas coisas, grupos de coisas ou como tais coisas se relacionam. Os elementos da OWL permitem a construção de constructos que irão representar este conhecimento.

II. REFERENCIAL TEÓRICO

Esta seção discute os principais tópicos relacionados a pesquisa desenvolvida e implementação apresentada neste trabalho, bem como alguns termos aqui utilizados.

A. Sistemas de Gerenciamento de Emergência

Serviços de emergência são fornecidos pelo governo como parte da gestão de segurança pública. Há diversas agências especializadas em cada tipo de ocorrência de incidente para atendimento ao público. Muitas dessas agências se envolvem em programas de conscientização da comunidade e prevenção de desastres na sociedade. Tais programas tem o intuito de ajudar o público a evitar, detectar e relatar efetivamente as emergências.

Um serviço eficaz de gerenciamento de emergência exige o trabalho unificado de diferentes agências que garantem tais serviços. Muitas agências têm procedimentos vinculados, ou pelo menos deveriam ter, com o objetivo de garantir uma efetiva comunicação, principalmente, em uma emergência de grande escala.

Como dito na seção I existem quatro fases específicas no gerenciamento de uma emergência. Conforme demonstrado na Figura 1:

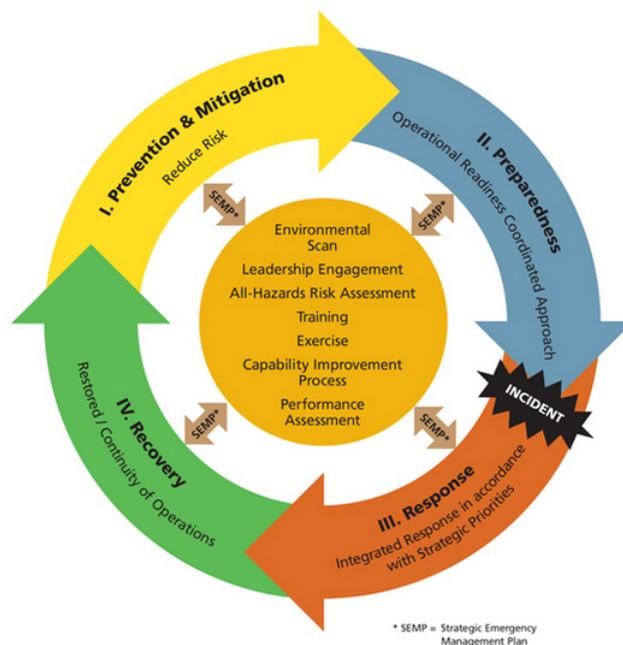


Figura 1. As quatro fases do processo de gerenciamento de emergência [5]

- 1) **Prevenção/Mitigação:** Nesta fase o objetivo é de reduzir os riscos com esforços de prevenção. A equipe de planejamento e redução de riscos deve realizar avaliações de risco específicas a emergência futuras.
- 2) **Preparação:** Nesta fase o objetivo é planejar uma abordagem coordenada e eficaz para o gerenciamento de emergência e a prontidão das forças operacionais para tal emergência.
- 3) **Resposta:** Nesta fase o objetivo é ter uma resposta eficaz e integrada com as devidas prioridades estratégicas da emergência.

- 4) Recuperação: Nesta fase o objetivo é fornecer a restauração e continuidade de serviços e operações críticas afetadas durante a emergência.

Em diversos países já foram implantados sistemas que fornecem uma estrutura de integração entre os sistemas de gerenciamento de emergência. No Brasil, a SEDEC (Secretaria Nacional de Defesa Civil) propôs uma iniciativa de integração de informações sobre ocorrência de desastres no Brasil. O S2ID (Sistema Integrado de Informações sobre Desastres) tem o objetivo de integrar os dados provenientes de diversos sistemas de gerenciamento de emergência do território nacional entre 1991 e 2010 e dar transparência a essas informações.

O CENAD (Centro Nacional de Gerenciamento de Riscos e Desastres) e a Anatel (Agência Nacional de Telecomunicações) propuseram a implantação de um sistema de alerta para catástrofes naturais. O objetivo deste sistema é alertar os usuários cadastrados de um possível caso de desastre natural. Esta comunicação é feita por mensagens de texto utilizando o serviço SMS (*Short Message Service*). Os avisos serão enviados pelo CENAD e pela Defesa Civil dos estados e municípios em casos de deslizamentos, inundações e outros casos de desastres naturais que possam ser previstos pela defesa civil.

A diferença entre tais sistemas e o RESCUER é que estes dois sistemas estão relacionados ao âmbito federal, o envio de relatos através de civis não é intuitivo. Porém, O RESCUER pretende apoiar a comunicação entre as forças operacionais e as testemunhas dos desastres ocorridos.

O RESCUER tem como foco emergências de larga-escala. As emergências de larga escala são mais complexas por tratarem com um grande número de pessoas, como por exemplo em um jogo sede da copa do mundo ou durante as olimpíadas onde há informação proveniente de milhares de pessoas. Mas também pode atender eventos e situações cotidianas onde o tráfego de pessoas é intenso como a estação rodoviária ou estação da lapa em Salvador.

O RESCUER aplica a técnica de *crowdsourcing*, técnica de coletar informações ou serviços provenientes de muitas fontes. A informação é proveniente de pessoas afetadas pelos incidentes, testemunhas oculares, forças de segurança e outras pessoas que possam disponibilizar informações sobre a emergência ou situações em áreas próximas. O RESCUER pretende dar uma resposta ao usuário do sistema através da análise dos relatos recebidos.

As situações de emergência que ocorrem em grandes centros urbanos ou que haja um grande número de pessoas ao redor desta emergência pode acarretar um grande consumo da rede dos dispositivos móveis. Este grande consumo da rede de comunicação é devido as ligações efetuadas e as mensagens enviadas ao serviço das forças operacionais durante uma situação de emergência.

Essas ligações e mensagens geram tráfego com uma massiva carga de dados para as forças operacionais que atuam na resposta da emergência e os agentes do centro de comando e controle. Pensando nestas informações e como gerenciar uma emergência em grande escala o projeto RESCUER viabiliza algumas soluções de gerenciamento e resposta a emergência.

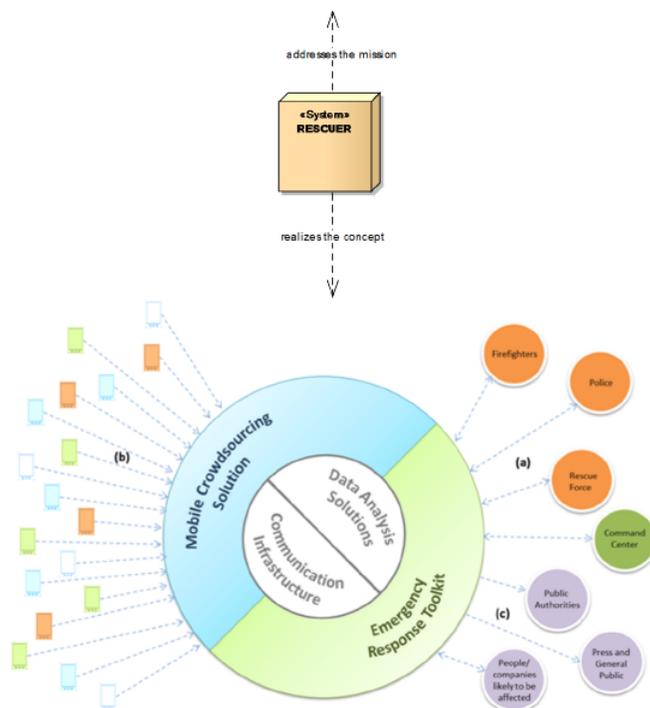


Figura 2. Missão RESCUER [6]

Como citado na Seção I, o projeto RESCUER pretende fornecer informações contextuais relacionadas a emergência para os centros de comando em tempo real, através da coleção, combinação e agregação de informações provenientes da multidão. A Figura 2 representa o conjunto de quatro soluções que englobam diversos componentes do RESCUER, as quais são definidas a seguir:

- Solução para dispositivos moveis: Suporte a testemunhas, comunicação oficial para serviço de ajuda (policia, bombeiro, segurança, etc..). É permitido o envio de informação em texto, formatos de imagem e vídeo.
- Solução de Análise de dados: Composto de algoritmos que processam e filtram os dados com objetivo de extrair informações requisitadas.
- Infraestrutura de Comunicação: Oferece o equipamento necessário para permitir a comunicação entre todos os interessados.
- *Toolkit* de Resposta a Emergência: É um conjunto de soluções para gerenciar a informação analisada da multidão, exibindo tais informações dentro do centro de comando e controle com a adequada metáfora de visualização.

O RESCUER possui definições de comunicação expressas desde a fase de criação do projeto e arquitetura. Vale, Arif e Gasparin [7] relatam os desafios relacionados ao desenvolvimento da arquitetura do RESCUER, configurações de ferramentas e equipes distribuídas em diversas localidades e instituições.

B. Ontologia EDXL-RESCUER

O termo ontologia tem origem grega e tem como objeto principal o estudo do ser e das suas propriedades gerais. Ontologia foi a palavra que guiou os estudos dos filósofos como Aristóteles, Platão e dentre outros. Ontologia em termo de tecnologia da informação é um modelo de dados que representa um domínio e o relacionamento dos conceitos dentro de tal domínio.

Uma das definições mais citadas de ontologia é de Thomas Gruber [8] "Uma ontologia é uma especificação explícita de uma conceitualização. O termo é emprestado da filosofia, onde uma ontologia é uma explicação sistemática da existência. Para sistemas baseados no conhecimento, o que "existe" é exatamente o que pode ser representado. Quando o conhecimento de um domínio é representado em um formalismo declarativo, o conjunto de objetos que podem ser representados como o universo do discurso".

Para Gomez et al [9] são necessários cinco componentes para compor uma ontologia, são eles:

- **Classes:** as classes representam conceitos, estes conceitos podem ser qualquer coisa proveniente de uma ampla interpretação de um domínio. As classes podem representar conceitos abstratos (intenções, crenças, tarefas, sentimentos) ou até mesmo conceitos concretos (Pessoas, computadores, móveis e etc). As classes podem ser organizadas em taxonomias, ou seja, organizado através de uma hierarquia de conceitos conforme a descrição, identificação e classificação dos conceitos, individualmente ou em grupo.
- **Relações:** as relações representam um tipo de interação entre os conceitos de um domínio. Ontologias normalmente possuem relações binárias onde o primeiro argumento é conhecido como domínio da relação e o segundo argumento como o alcance da relação. Relações binárias também são utilizadas para expressar atributos dos conceitos, nesse caso o alcance da relação é sempre um tipo de dado como texto, numérico, entre outros.
- **Funções:** são casos especiais de relações onde o n -ésimo elemento da relação é único para o elemento que precede ele ($n-1$).
- **Axiomas formais:** de acordo com Gruber [8], axiomas formais servem para modelar sentenças sobre conceitos e relações que são sempre verdade. Eles são, normalmente, utilizados para representar conhecimento que não podem ser formalmente representados por outros componentes.
- **Instâncias:** são usadas para representar elementos ou indivíduos de um conceito em uma ontologia.

A suíte de ontologias EDXL-RESCUER [10] foi criada com o intuito de apoiar a interoperabilidade semântica dentro do projeto RESCUER. A equipe de desenvolvimento das ontologias baseou-se no padrão EDXL [11] (*Emergency Data Exchange Language*). Este padrão tem como objetivo facilitar o compartilhamento e troca de dados entre organizações locais, regionais, nacionais e não governamentais.

O padrão EDXL viabiliza uma estrutura de comunicação comum utilizando constructos da XML (*Extensible Markup Language*). O EDXL foi subdividido em sete módulos dentro do domínio de emergências, dentre estes sete módulos, quatro módulos foram utilizados para escrever a ontologia EDXL-RESCUER, são estes:

- **EDXL-DE [12] (*Distribution Element*):** O propósito do DE (Elemento de distribuição) é facilitar o encaminhamento de mensagens de emergências para os destinatários. O Elemento de distribuição pode ser considerado um recipiente, isto é, elemento que fornece o conjunto de mensagens como os alertas ou mensagens para recursos. O DE também fornece informação para redirecionamento da carga (como as mensagens de alertas e mensagens de recursos) através das informações chave de roteamento como o tipo de distribuição, geografia, incidente e o ID do remetente e do destinatário.
- **EDXL-CAP [13] (*Common Alert Protocol*):** O protocolo de alerta comum especifica um formato de alertas de emergências e avisos públicos sobre todos os tipos de redes. Este protocolo permite o compartilhamento de uma mensagem de advertência consistente para diferentes sistemas de alertas. O CAP fornece um *template* para mensagens efetivas de alertas baseado nas melhores práticas identificadas em pesquisas acadêmicas e em experiências do mundo real.
- **EDXL-RM [14] (*Resource Messaging*):** O objetivo da especificação RM é fornecer um conjunto de formatos padrão para mensagens de respostas a emergência. Estas mensagens de recursos são mensagens com recursos necessários para responder e se adaptar a incidentes de emergência.
- **EDXL-SitRep [15] (*Situation Reporting*):** O propósito do SitRep é fornecer um conjunto de formatos padrão para mensagens de respostas a emergência. Especificamente estas mensagens transmitidas relatam a situação a tempo. O módulo SitRep destina-se a agilizar relatórios completos necessários para responder eficazmente e adaptar-se a incidentes de emergência, facilitando a comunicação entre várias organizações respondentes e a cadeia de comando. O SitRep contém um conjunto de relatórios pré-definidos.

Por decisão dos colaboradores do projeto RESCUER não foram incluídos outros três módulos (EDXL-HAVE, EDXL-TEP, EDXL-RIM), por não apresentarem um contexto necessário para o domínio amplo de resposta a emergência e tomadas de decisões relacionadas ao RESCUER.

A ontologia EDXL-RESCUER foi construída tomando como base os modelos de dados dos requisitos do RESCUER e dos modelos de dados descritos na linguagem EDXL, a técnica utilizada para esta análise é chamada de taxonomia facetada definida por Prieto-Díaz [16]. As facetas são perspectivas, visões ou dimensões de um domínio no qual cada faceta modela um aspecto deste domínio.

No intuito de atualizar e aplicar a abordagem de uma classificação facetada na ontologia EDXL-RESCUER, Barros et al [17] seguiu uma série de passos:

- 1) Coletar informações provenientes do EDXL e entendimento do domínio de SGEs.
- 2) Listar as entidades presentes.
- 3) Criar faceta, a principal entidade escolhida para representar o domínio de segmento EDXL.
- 4) Combinar as facetas com as devidas entidades de acordo com a documentação EDXL.
- 5) Ordenar a citação e classificar as fases que referem como a taxonomia deveria ser descrita. No caso EDXL-RESCUER, uma ontologia escrita em OWL, definição o que é uma classe, sub-classe, propriedade de objeto e propriedade de dado.

C. Engenharia do conhecimento

Engenharia do conhecimento é uma área de estudo que visa representar conhecimento em modelos formais, alguns exemplos que utilizam abordagens desta área de estudo, são:

- I Aplicações de *web* semântica⁴
- II Integrações entre sistemas
- III Sistemas especialistas
- IV Inteligência Artificial ou sistemas baseados no modelo arquitetural *Blackboard*
- V Portais de conhecimento
- VI Aplicações envolvendo *big data*

Para Hapel e Seedorf [18] ontologias podem ser utilizadas para resolver problemas discutidos da área de engenharia de *software*. Eles identificaram e atribuíram abordagens que explicam o benefício de ontologias em diversas fases do ciclo de vida de um *software*.

Hapel e Seedorf [18] definem os problemas em cada processo (por exemplo durante o processo de análise e projeto, durante o processo de levantamento de requisitos e outros processos relacionados ao ciclo de vida do *software*) e como a ontologia soluciona tais problemas. Eles citam que a ontologia pode servir como componente de reuso e documentação, ajudar com as regras de negócio do projeto.

Para Dillon, Chang e Wongthongtham [19] há diversos propósitos para o qual uma ontologia possa ser utilizada:

- Para viabilizar um forte mecanismo de comunicação sem ambiguidade. Ontologia pode ser um meio de referência para os *stakeholders* que trabalham no desenvolvimento do *software*.
- Para fornecer uma mediação entre mecanismos de comunicação para acessar dados heterogêneos e fontes de informações, particularmente na *web*.
- Para possibilitar a construção de aplicações na *web* fornecendo uma semântica definida claramente para serviços *web*.
- Para proporcionar uma base comum de conhecimento para múltiplos agentes trabalhando em um domínio particular.
- Para fornecer uma definição semântica clara, provendo credibilidade entre as interações na *web*, especificamente, Sistemas baseados em privacidade, confiabilidade e reputação.

D. Heterogeneidade dos dados

Os casos de estudos apresentados por Patil, Dutta e Sriram [20] identificam a fase de desenvolvimento de produtos a partir de duas ferramentas (Unigraphics⁵ e Solidworks⁶) de CAD (*Computer-Aided Design*). Eles identificam as diferenças entre a produção de produtos com duas ferramentas distintas.

Patil, Dutta e Sriram [20] fazem o mapeamento semântico do desenvolvimento do produto e definem uma estrutura para troca semântica dos dados dos produtos. Eles apresentaram um *framework* de troca de dados semânticos para produtos, chamado de PSRL (*Product Semantics Representation Language*).

Equipes colaborativas para desenvolvimento de produtos têm enfrentado diversas barreiras (conhecimento, diferentes ferramentas, geográficas, culturais e etc.) e estas equipes, geralmente, não utilizam de conceitos compartilhados. Cada equipe tem a tendência de desenvolver os produtos com as ferramentas e os conhecimentos a disposição da equipe.

Essa tendência pode ocasionar uma heterogeneidade no ambiente de desenvolvimento. A heterogeneidade no ambiente de desenvolvimento do produto pode não se tornar um problema caso haja uma representação dos conceitos e um compartilhamento das informações dos produtos entre diferentes domínios de aplicações. Além da heterogeneidade no ambiente de desenvolvimento é passível que haja heterogeneidade nas fontes de dados também.

As fontes de dados podem ser heterogêneas em sintaxe, esquema e semântica. Heterogeneidade sintática é causada pelo uso de diferentes modelos ou linguagens; heterogeneidade esquemática refere-se a diferença estrutural entre os dados; heterogeneidade semântica é causado pelos diferentes significados ou interpretações dos dados em diversos contextos [21]. Para que haja interoperabilidade entre as fontes de dados é necessário que se resolva os problemas de heterogeneidade dos dados.

Um esquema XML fornece uma maneira de especificar um modelo hierárquico dentro do padrão XML e os constructos definidos. Porém, mesmo que os dois sistemas que estão se comunicando utilizem o mesmo esquema XML (homogeneidade esquemática e sintática), existe a possibilidade de haver heterogeneidade semântica. Em um esquema XML há uma grande chance de existir ambiguidade pois o esquema XML não fornece constructos para definição de uma classe e suas propriedades para um modelo semântico.

Integração semântica é o processo de usar uma representação conceitual de dados e seus relacionamentos para resolver a heterogeneidade semântica. A integração semântica pode facilitar ou até mesmo automatizar a comunicação entre sistemas de gerenciamento de emergência usando metadados publicados. Tais metadados oferecem a possibilidade de vincular diretamente a uma ontologia.

Uma abordagem de mapeamento do dado (semi)automática requer a definição da distância semântica, similaridade semântica e regras do contexto. A integração semântica também pode ser usada para facilitar as atividades em tempo de projeto.

⁵<https://www.plm.automation.siemens.com/en/products/nx/about-nx-software.shtml>

⁶<http://www.solidworks.com>

⁴<https://www.w3.org/standards/semanticweb/>

Esta abordagem precoce da ligação semântica pode melhorar a performance do sistema no geral enquanto possui os benefícios de uma abordagem de projeto dirigido a semântica – *semantic driven design*.

Cruz e Xiao [21] identificaram quais são os cinco usos de uma ontologia em relação a integração semântica:

- Usabilidade para uma abordagem de integração de dados, como forma de representação de metadados, ou seja, metadados em cada fonte de dados podem ser representados por uma ontologia local, usando uma única linguagem.
- A ontologia global fornece uma visão conceitual sobre diversas fontes de esquemas.
- Para oferecer apoio as consultas de alto nível, sendo assim, o usuário pode formular uma consulta sem precisar ter conhecimento específico de diferentes fontes de dados.
- Uso de uma ontologia global para ser um mediador declarativo para consultas; pode ser usada como um meio de processar consultas entre sistemas.
- Termos formais de uma ontologia podem ser usados como tabela de mapeamento para facilitar o processo de automação.

III. LEGACY SYSTEM CONNECTOR – LSC

O RESCUER possui documentos, *Deliverables*, para demonstrar todas as especificações e informações relacionadas ao desenvolvimento de cada componente dentro do RESCUER. A Figura 3 disponibiliza uma visão contextual dos componentes do RESCUER, esta Figura foi retirada do *Deliverable* 1.2.2 [6].

Essa visão demonstra como os módulos estão organizados dentro do projeto RESCUER e a forma de comunicação entre eles. O componente LSC (*Legacy System Connector*) fornece a interface de comunicação necessária entre o RESCUER e o componente externo *Operation Force System*. Esta seção tem como objetivo descrever o componente LSC e como tal habilita a comunicação bilateral entre o RESCUER e sistemas externos.

O RESCUER *backend* representa o conjunto de componentes responsáveis por análise de textos, imagens, vídeos, dados de sensores dos dispositivo e ainda contém a plataforma de integração, Conectores para redes sociais, Conector para sistemas legados e serviço de armazenamento na nuvem.

A plataforma de integração é um mecanismo de integração genérico que fornece uma comunicação assíncrona entre os componentes. Todos os componentes no RESCUER se comunicam através da plataforma de integração obedecendo a arquitetura do projeto onde a comunicação é feita através de uma sintaxe definida para emergências.

As informações processadas por cada componente são enviadas enviada para outro componente através da plataforma de integração que é responsável por direcionar as mensagens para os componentes de destino. Devido ao cenário de comunicação interna do projeto RESCUER definiu-se a aplicação de uma comunicação *peer-to-peer*.

A abordagem *peer-to-peer*, apresentada no trabalho de Cruz e Xiao [21], define uma integração semântica no qual o SGE (Sistema de Gerenciamento de emergência) irá se inscrever para recebimento dos dados semânticos e publicar as suas próprias informações sobre um incidente em um vocabulário comum, no caso desta integração o vocabulário da ontologia EDXL-RESCUER.

A Figura 4 representa o diagrama de componentes que são utilizados para compor a comunicação entre o sistema externo e o RESCUER. Nesta mesma figura podemos identificar os componentes utilizados pelo RESCUER e as *interfaces* proviadas para que tal estrutura de comunicação opere normalmente.

O Componente LSC foi desenvolvido na linguagem de programação Java com a utilização da técnica de *threads*. As *threads* foram criadas para realizar um processamento paralelo da informação recebida pelo SGE. Durante a fase de desenvolvimento do componente foi realizado mapeamento das fontes de dados e escolha de uma ferramenta de apoio para tal mapeamento. As subseções a seguir irão detalhar o processo de desenvolvimento do LSC.

A. Mapeamento das fontes de dados

Apesar de ser um projeto de grande proporção que envolve instituições de pesquisa da união Europeia e do Brasil, o projeto RESCUER não apresenta nenhuma comunicação concreta com sistemas legados. A equipe responsável pelo componente LSC conseguiu acesso a fontes de dados de dois sistemas relacionados a resposta de diversas emergências.

A fonte de dados do CECOCO (Centro de Coordenação de Comunicações) são provenientes do parceiro STELECOM (Superintendência de Telecomunicações da Secretaria de Segurança Pública da Bahia – SSP-BA). CECOCO é o sistema utilizado pela polícia militar do estado da Bahia para registrar e acompanhar os recursos disponibilizados para cada ocorrência.

Além de acompanhar os recursos, o CECOCO permite aos diversos envolvidos nas operações policiais realizarem a alocação necessária de recursos de acordo com os desdobramentos do teatro de operações. O cadastramento da ocorrência é feita conforme a comunicação com o teleoperador da PM-BA.

O CECOCO é um sistema que trata de um domínio amplo de segurança pública. A partir desta verificação podemos notar o registro de diversas ocorrências que não condiziam com o domínio apresentado para o RESCUER. A fonte de dados do CECOCO foi modificada para consultar somente registros que poderiam ser relacionados a situações de desastres em larga-escala.

A fonte de dados do CECOCO repassada para equipe do LSC encontra-se no formato CSV (*Comma-Separated Values*)⁷, um formato de texto plano onde os campos são separados por vírgulas. Um exemplo do registro da ocorrência do CECOCO encontra-se no Apêndice A.

Foi utilizada uma segunda fonte de dados externa proveniente da base pública do governo Canadense. O CDD, acrônimo para *Canadian Disaster Database*⁸, contém informações deta-

⁷<https://tools.ietf.org/html/rfc4180>

⁸<https://www.publicsafety.gc.ca/cnt/rsrscs/cndn-dsstr-dtbs/index-en.aspx>

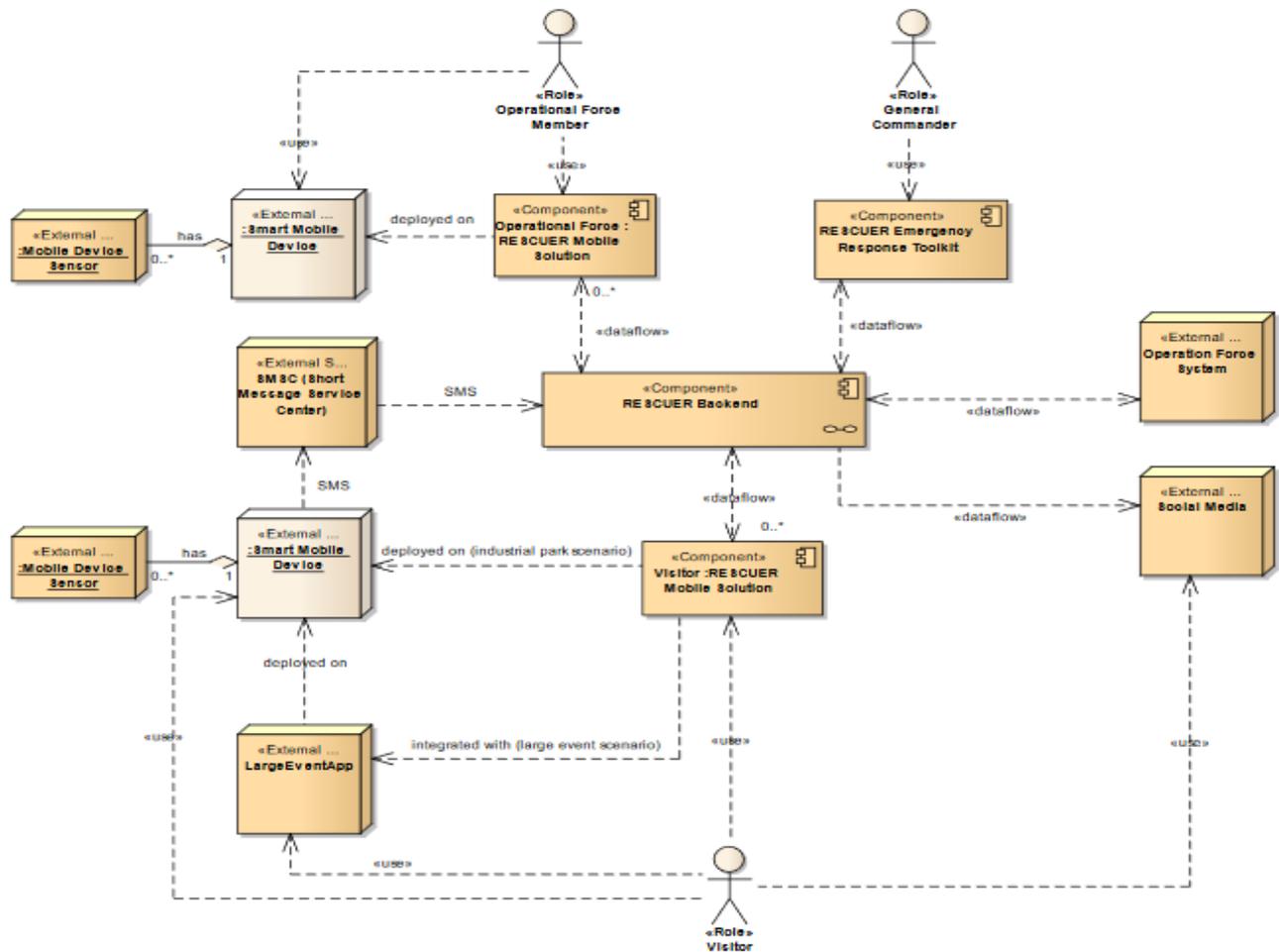


Figura 3. Visão geral do contexto [6]

lhadas de desastres naturais, tecnológicas e eventos de conflitos (exceto guerras) que afetaram canadenses desde 1900.

A base de dados de desastres canadense descreve onde e quando um desastre ocorreu, o número de feridos, fatalidades e até mesmo o custo para o governo deste desastre. Estes dados também foram extraídos no formato CSV. Um exemplo do registro da ocorrência da fonte de dados Canadense, CDD, encontra-se no Apêndice A.

Como parte da integração bilateral a fonte de dados do RESCUER no formato JSON (*JavaScript Object Notation*)⁹ também foi mapeada. As fontes de dados foram mapeadas no domínio de gerenciamento de emergência, seguindo as especificações da ontologia EDXL-RESCUER. Um exemplo de um relato enviado no RESCUER encontra-se no Apêndice A.

O mapeamento é essencial para que todas as sintaxes e esquemas possam ser traduzidos em uma definição semântica para o domínio de gerenciamento de emergências. Sendo assim qualquer SGE poderá integrar com outro SGE através da semântica EDXL-RESCUER. A Figura 5 representa uma visão dos arquivos de mapeamento para a semântica de interme-

dição especificada pela EDXL-RESCUER. O conjunto de ontologias estão disponíveis para *download* através dos *links*:

- EDXL_SitRep: http://fpc.ufba.br/ontologies/edxl_sitrep
- EDXL_CAP: http://fpc.ufba.br/ontologies/edxl_cap
- EDXL_DE: http://fpc.ufba.br/ontologies/edxl_de
- EDXL_RM: http://fpc.ufba.br/ontologies/edxl_rm

B. Legacy System Connector

A equipe responsável pelo LSC identificou algumas ferramentas de apoio a mapeamento de ontologias e esquemas de fonte dados como a MapOnto¹⁰, Coma¹¹, OntoM¹², Karma¹³. Destas ferramentas, o Karma foi escolhido por possuir diversas características e ser uma ferramenta *web*.

O Karma é uma ferramenta de integração baseada em ontologias desenvolvida pela USC (*University of Southern California*). Esta ferramenta permite integrar dados de diversas fontes de dados, inclusive acesso direto a esquemas de banco

⁹<http://www.json.org/>

¹⁰<http://www.cs.toronto.edu/semanticweb/maponto/>

¹¹<https://dbs.uni-leipzig.de/Research/coma.html>

¹²<https://bitbucket.org/tomers77/ontobuilder/wiki/OntoM>

¹³<http://usc-isi-i2.github.io/karma/>

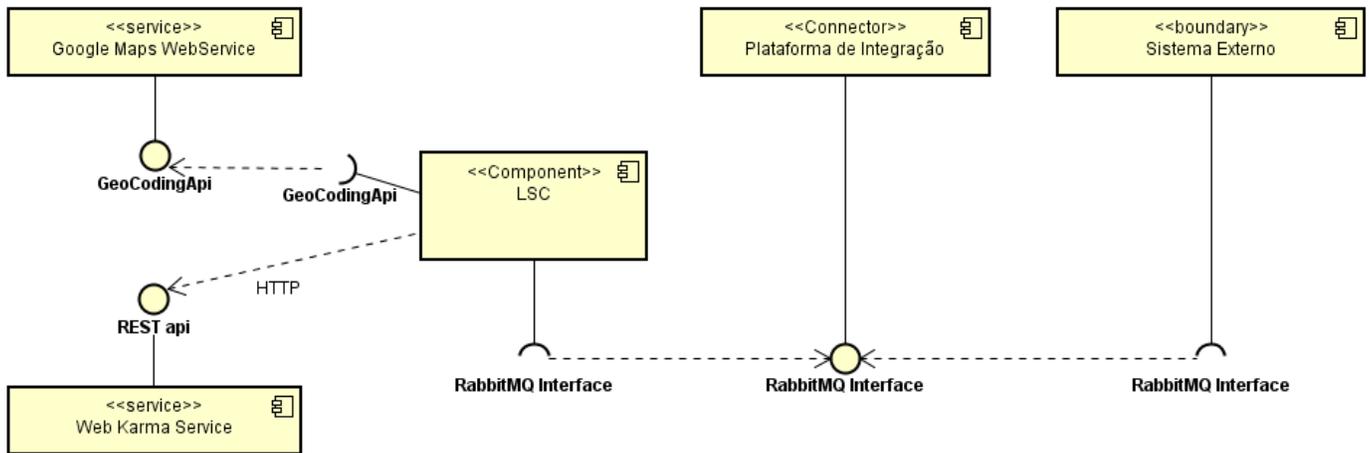


Figura 4. Diagrama de Componente do LSC

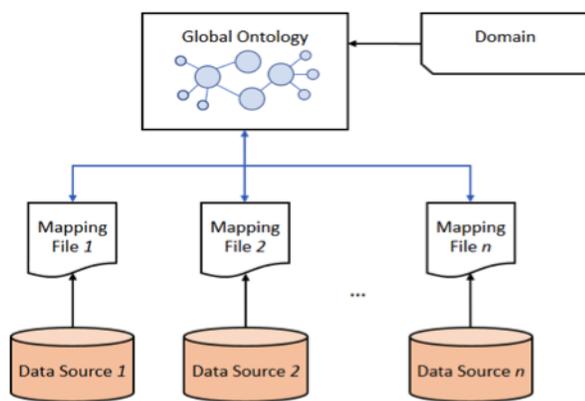


Figura 5. Visão das fontes de dados e cada arquivo de mapeamento referente a fonte de dados.

de dados. Ela oferece suporte a troca de dados (por exemplo ela transforma os dados do esquema de origem para o formato RDF¹⁴).

O formato JSON-LD¹⁵ [22], um dos formatos de serialização RDF, foi escolhido para ser o formato semântico dos dados entre os sistemas legados e o RESCUER. O JSON-LD foi escolhido por apresentar um formato de documento JSON (formato comumente utilizado em aplicações *web*) e ter conceitos definidos pelo RDF para dados "linkados". O Karma foi escolhido para dar o suporte a tradução dos esquemas de origens durante a fase de desenvolvimento do componente LSC e durante a fase de implantação do LSC.

O Karma permite ao usuário mapear as fontes de dados e traduzir automaticamente a fonte de dados utilizando o respectivo arquivo de mapeamento. Durante a fase de desenvolvimento do componente LSC foi utilizado o *web* Karma

para criar os arquivos de mapeamento dos esquemas de fontes da dados do CECOCO, CDD e do RESCUER.

A ferramenta Karma é flexível quanto aos formatos que podem ser importados (como JSON, CSV, XML e esquemas de bancos de dados SQL). Além de possuir um aplicação *web* intuitiva que ajuda no mapeamento, o Karma é uma ferramenta de código aberto.

Existe uma funcionalidade dentro do Karma que permite que o usuário utilize-se de programação de códigos na linguagem de programação Python¹⁶. Esta funcionalidade é o módulo do *pytransform* que pode ser utilizada para recuperar os devidos valores com funções e módulos que uma linguagem de scripts como o Python possa proporcionar.

Durante a fase de desenvolvimento do componente LSC foram efetuados uma série de passos para criar os arquivos de mapeamento de cada fonte de dados. Primeiramente o estudo das fontes de dados para identificação do conceito de cada tupla¹⁷ para ser representada na semântica da ontologia EDXL-RESCUER.

No segundo passo nós enviamos o suíte de ontologias EDXL-RESCUER para a ferramenta Karma para que a ferramenta possa interpretar os conceitos da ontologia e em seguida colocamos as fontes de dados. Com a ontologia e os dados, identificamos como fazer o mapeamento das fontes de dados dentro da ferramenta Karma.

Depois de mapear os dados para os valores da ontologia é gerado o arquivo de mapeamento que será colocado na versão de implantação do componente LSC. Os arquivos de mapeamento foram baseados na convenção do R2RML¹⁸ (*Relational to RDF Mapping Language*), uma linguagem para expressar mapeamentos personalizados de um banco de dados para um conjunto de dados em RDF.

O arquivo de mapeamento, apesar de ser baseado no R2RML, tem a particularidade de características definidas na

¹⁴No contexto de *web* semântica uma ontologia é expressa em linguagens como o RDF (*Resource Description Framework*). O RDF é um *framework* para definição ampla de recursos na *web*. Neste sentido o RDF pode ser escrito em diversos formatos de serialização.

¹⁵<https://json-ld.org/>

¹⁶<https://www.python.org/>

¹⁷Por se tratar de arquivos CSV o conceito de tupla aqui utilizado é o mesmo conceito de banco de dados, onde cada registro é uma tupla

¹⁸<https://www.w3.org/TR/r2rml/>

KR2RML [23]. O KR2RML é uma variação do R2RML implementado dentro da ferramenta Karma para adicionar alguns requisitos, como proporcionar múltiplos formatos de entrada e saída, adicionar extensibilidade, transformações e escalabilidade em relação aos dados.

O componente LSC também utilizou-se da interface provida pelo serviço do Google Maps [24]. Esta comunicação estabelecida com o serviço de geocodificação tem o propósito de receber a posição em latitude e longitude de um determinado endereço conforme pesquisa feita dentro do serviço do Google Maps.

Outra interface utilizada dentro do LSC é a interface da plataforma de integração do RESCUER. O RabbitMQ¹⁹, um *message broker*, tem a finalidade de construção de um serviço centralizado de comunicação entre diversos processos e aplicações.

O RabbitMQ foi especificado pela equipe responsável pela plataforma de integração conforme *Deliverable 1.5.1* [25]. Partindo da premissa que os sistemas legados utilizarão parte da solução da plataforma de integração para comunicação com o RESCUER *backend*, utilizamos o RabbitMQ para nos auxiliar com tal tarefa de comunicação.

O RabbitMQ oferece uma abordagem baseada em mensagens. A mensageria, um jargão computacional proveniente da tradução literal do termo inglês *Messaging*, consiste na utilização de mensagens para estabelecer a comunicação síncrona ou assíncrona entre aplicações. Uma mensagem pode ser definida como uma estrutura de dados composta por metadados como host de origem/destino, fila de destino, além de dados fornecidos pela aplicação.

A arquitetura do ambiente RESCUER tem o RabbitMQ como componente gerenciador de filas e de mensagens onde são criadas as filas que serão utilizadas para as aplicações enviarem ou pegarem mensagens. Ele também se encarrega de gerenciar as camadas mais baixas de comunicação como TCP/IP, ou outros protocolos.

A analogia utilizada pela equipe do RabbitMQ é que o mesmo funciona como se fosse uma agência dos correios e o carteiro ao mesmo tempo. O RabbitMQ recebe a mensagem do remetente (sistema/processo produtor da informação) e também é responsável por postar a mensagem no destinatário (sistema/processo consumidor da informação).

No RabbitMQ o processo posta uma mensagem do tipo BLOB (*Binary Large Object*) em uma fila e a partir do mecanismo interno do RabbitMQ o consumidor cadastrado para consumir mensagens de uma determinada fila recebe as mensagens desta fila. O RabbitMQ é responsável por aceitar, armazenar e encaminhar os dados BLOBs.

O RabbitMQ oferece funcionalidades como o encaminhamento flexível das mensagens, agrupamento de instâncias de servidores, as filas podem ser espelhadas entre diferentes *hardwares* em um *cluster* certificando a segurança das mensagens mesmo ocorrendo falha de *hardware*. O RabbitMQ possui uma grande comunidade de apoio e também possui suporte comercial.

IV. AVALIAÇÃO

Foram realizados testes de unidade para testar tal integração de sistemas externos com o RESCUER. Como citado na seção III, não foi possível efetuar testes de integração com sistemas que estejam em produção e ajudando as forças operacionais com respostas de emergência.

Porém a equipe do LSC construiu uma aplicação que simula o envio dos dados colhidos. Foram realizados testes de unidades e de comunicação com outros componentes do RESCUER. As especificações técnicas de cada componente do RESCUER são estabelecidas em seus respectivos *Deliverables*.

Foram definidos os formatos e os dados imprescindíveis para a comunicação dos componentes e acompanhamento do incidente. Um dado imprescindível para o RESCUER é a posição da coordenada geográfica do relato. O componente ERTK baseia-se na posição de latitude e longitude para demonstração no mapa do centro de comando e controle do RESCUER.

Porém as duas fontes de dados do CECOCO e da base canadense (CDD) estão com a localização baseadas em endereços postais. Para a conversão de endereços em coordenadas geográficas utilizamos o serviço do Google Maps de geocodificação [24].

O serviço de geocodificação oferecido pelo Google Maps possui limites de conversão. Todos os serviços Google Maps API (*Application Programming Interface*) exigem autenticação. Para autenticação da requisição foi utilizada uma chave de desenvolvedor padrão gerada somente para meios de teste de comunicação com o serviço de geocodificação.

Esse tipo de chave padrão permite que a aplicação faça, no máximo, 2500 (duas mil e quinhentas) consultas por dia. Cada linha dos arquivos lidos são equivalentes a um relato produzido por cada sistema. Então tivemos o cuidado durante a simulação enviar somente o número menor ou igual a 2500 (dois mil e quinhentos) relatos de incidentes por dia durante o teste do componente.

Nesta seção de avaliação estamos denominando o sistema que simula o envio das informações das fontes de dados como sistema legado. Tal aplicação efetua a tradução da informação para o vocabulário da ontologia EDXL-RESCUER e a publica no RabbitMQ.

A. Enviando informações para Sistema legado

Este é o cenário de envio da informação de um incidente com *status* confirmado para o sistema legado. Este cenário não foi totalmente explorado e identificado nenhum problema por não apresentar particularidades de um SGE externo. Não foi reconhecido como seria uma ação do sistema legado ou da força tarefa em relação ao recebimento da informação de um incidente enviado pelo RESCUER.

A Figura 6, um diagrama de sequência, demonstra os passos em sequência da comunicação efetuada do RESCUER para o sistema legado. Em detalhes estes passos descrevem uma ordem:

- 1 PublicarIncidente(): O ESB (*Emergency State Builder*) é o módulo responsável pela junção dos variados relatos relacionados a um incidente. Quando o

¹⁹<https://www.rabbitmq.com/>

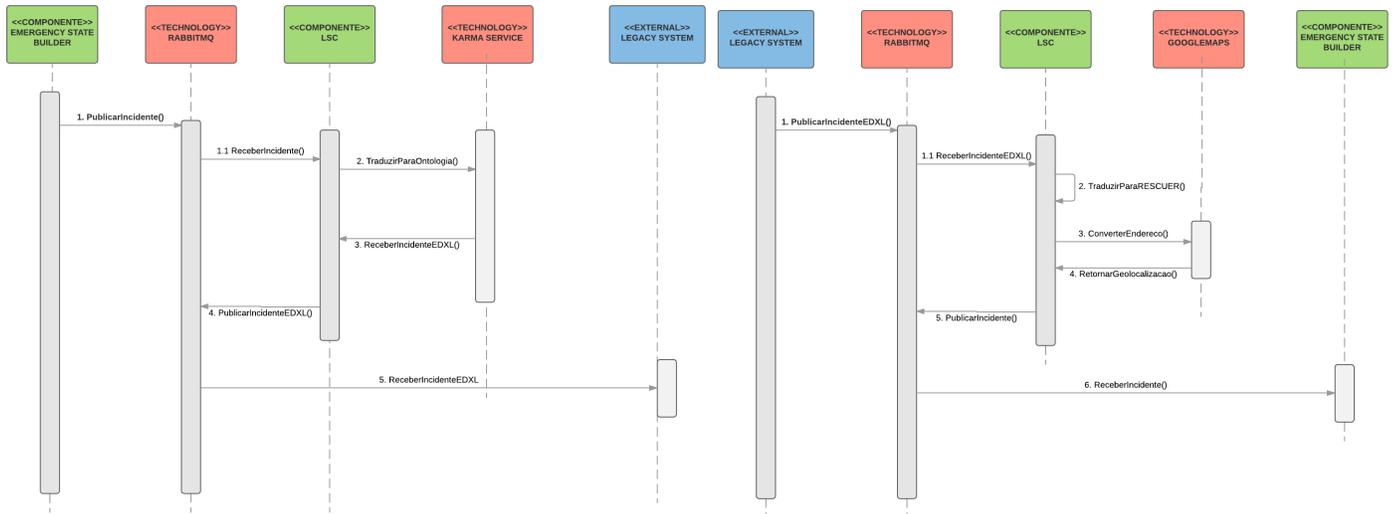


Figura 6. Envio da informação de um incidente para o sistema legado

incidente é confirmado o ESB publica informações do incidente no RabbitMQ para os componentes interessados na informação do incidente.

- 1.1 ReceberIncidente(): Através do RabbitMQ o LSC recebe a mensagem do ESB com informações do incidente confirmado.
- 2 TraduzirParaOntologia(): O componente LSC recebe a informação no formato JSON e envia a informação do incidente e o arquivo de mapeamento associado para o serviço do Karma.
- 3 ReceberIncidenteEDXL(): O serviço do Karma retorna a informação do incidente no vocabulário da Ontologia EDXL-RESCUER.
- 4 PublicarIncidenteEDXL(): O LSC publica o novo incidente na fila do RabbitMQ.
- 5 ReceberIncidenteEDXL(): O sistema legado consome o novo incidente via RabbitMQ.

B. Enviando informações para o RESCUER

Este é o cenário de envio da informação de um incidente com *status* confirmado para o RESCUER. Neste cenário podemos identificar alguns problemas e explorar a comunicação entre os sistemas de respostas a emergência.

Os passos descritos no diagrama de sequência 7 mostram a troca de mensagens entre o sistema legado e o RESCUER. Em detalhes estes passos descrevem uma ordem:

- 1 PublicarIncidenteEDXL(): A partir da leitura dos dados, o sistema legado publica a informação de um incidente confirmado no RabbitMQ. O sistema legado publica a informação de incidente já traduzida para o vocabulário da ontologia EDXL-RESCUER.
- 1.1 ReceberIncidenteEDXL(): O LSC recebe a mensagem do sistema legado através do RabbitMQ

Figura 7. Envio da informação de um incidente para o RESCUER

- 2 TraduzirParaRESCUER(): É papel do LSC fazer a tradução da informação do incidente no vocabulário da ontologia EDXL-RESCUER para o formato do JSON. O componente LSC traduz o incidente em JSON-LD.
- 3 ConverterEndereco(): A coordenada geográfica, como citado anteriormente, é imprescindível para que o ESB possa fazer uma correta leitura da informação do incidente. Sendo assim, O LSC consulta uma eventual coordenada geográfica que seja referente ao endereço passado.
- 4 RetornarGeolocalizacao(): O serviço de geocodificação do Google Maps Api retorna lista de possíveis coordenadas geográficas referentes ao endereço passado durante a consulta.
- 5 PublicarIncidente(): O componente LSC é responsável por publicar o incidente no vocabulário que o ESB possa distribuir para os outros componentes.

Durante os testes de avaliação realizados no componente e na comunicação bilateral do RESCUER e do simulador do sistema legado foram verificados alguns problemas. A fonte de dados proveniente da STELECOM, o sistema CECOCO, estava com alguns problemas em relação a escrita dos dados.

Pudemos notar que haviam muitas irregularidades nos relatos de incidentes. Tais irregularidades foram notadas como duplicidade de informações ou até mesmo a falta delas. Em alguns relatos podemos verificar a presença de nomes de bairro em duplicidade e até mesmo o endereço postal não digitado corretamente.

Tais irregularidades nos dados do CECOCO acarretaram em problemas de tradução nos relatos pois o serviço do Google Maps não conseguia encontrar uma coordenada geográfica daquele endereço com irregularidade. Além da avaliação utilizada pela equipe do LSC, o projeto RESCUER também passou por uma avaliação.

O projeto RESCUER passou por uma avaliação de corretores e avaliadores que testaram a solução como um todo.

O funcionamento de todos os componentes e todas as suas funcionalidades operando em conjunto. E durante esta avaliação ficou visível que o retorno de uma coordenada geográfica através de um endereço postal pode não ser o ponto exato em que o incidente está ocorrendo.

A avaliação do RESCUER aconteceu no ambiente do Polo petroquímico de Camaçari onde foi simulado um incidente dentro do COFIC (Comitê de Fomento Industrial de Camaçari). Foi utilizado o simulador de sistema legado passando o endereço postal do COFIC como relato de incidente com fogo em um local que possuía caldeiras e dutos da água.

Porém, durante a simulação, o retorno da coordenada geográfica do endereço postal retornou um ponto no início da rodovia BA-512. Percebemos que por se tratar de um endereço que não possui número e ser uma localização distante da cidade, a coordenada retornada foi distante da real coordenada do incidente.

O projeto RESCUER teve uma maneira de solucionar este problema. O componente responsável pela visualização no centro de comando, o ERTK, tem a possibilidade de agregar o posicionamento do incidente com o real posicionamento do incidente. Sendo assim o agente responsável pelo centro de comando e controle pode verificar se as informações relatadas condiz com o incidente e agregar esta informação ao local correto do incidente.

V. TRABALHOS RELACIONADOS

O sistema de suporte SoKNOS é um protótipo de sistema de gerenciamento de emergência que utiliza ontologia. As fontes de informação do SoKNOS e os serviços utilizam-se de ontologias para prever a informação correta no tempo correto. A implementação utiliza o OntoBroker²⁰ no qual converte os dados armazenados em bancos de dados relacionais no vocabulário de uma ontologia.

No trabalho de Santos, Sicília e Padrino [26] eles avaliam a utilização de ontologias como método de integração entre sistemas de gerenciamento de emergência. Inclusive eles citam a integração entre dois sistemas de emergência, Ganesha²¹ e o Sahana²².

O Ganesha e o Sahana se comunicam através de suas bases de dados onde um sistema infere seus dados e o raciocínio para automatizar situações de gerenciamento de emergência. Este raciocínio é feito entre os sistemas baseando-se nos dados agregados em uma única base de dados.

No trabalho de mestrado de Barros [27] é proposto a implementação de acesso de dados baseada em ontologias. Neste trabalho os bancos de dados usados para a interoperabilidade foram disponibilizados através de uma única interface de acesso as bases.

Barros [27] utiliza o SPARQL²³ para consultar os dados de um banco virtual das bases de dados integradas. Esta

abordagem permite a integração dos dados através do uso de *queries* SPARQL recuperando os dados armazenados nos bancos de dados conectados.

Estes trabalhos apresentados possuem uma característica baseada no acesso direto aos bancos de dados e no conceito de fusão de dados. Fusão de dados é o processo de integração de múltiplas fontes de dados para produzir informações mais consistente, exatas e úteis que podem ser fornecidas em uma única fonte de dados. Já este trabalho apresenta um meio único de comunicação onde possibilita que um ou mais sistemas (*peers*) possam se comunicar através de mensagens. O trabalho aqui apresentado também possibilita o envio de informações mais detalhadas ou com menos detalhes, dependendo das especificações de negócio entre os sistemas de gerenciamento de emergência.

VI. CONCLUSÃO

Este trabalho apresentou a experiência de troca de dados de sistemas de gerenciamento de emergência baseando-se em ontologias. Este mecanismo de comunicação viabiliza a interação através de um vocabulário rico e com seus constructos bem definidos. O foco desta integração é o sistema RESCUER pois o componente LSC é um componente que foi criado para tal integração no projeto.

Esta abordagem pode ser aplicada para qualquer SGE que queira interagir com o RESCUER e que possa fazer o mapeamento das suas fontes de dados para a ontologia EDXL-RESCUER. O desenvolvimento de um simulador de sistema legado caracteriza que o mapeamento das fontes de dados deve ser uma função individual de cada sistema de gerenciamento de emergência.

Cada sistema é responsável por gerir e identificar os dados provenientes das suas fontes de dados. Os dados de dois diferentes sistemas instanciados em dois países com diferentes culturas pode demonstrar que o uso da ontologia EDXL-RESCUER como um vocabulário comum entre eles, funciona. A ontologia EDXL-RESCUER consegue promover a interoperabilidade semântica entre sistemas de resposta à emergência.

Por questão de segurança da informação do projeto RESCUER e por trabalhar com dispositivos de terceiros optou-se por não ter um acesso direto ao banco de dados do RESCUER. Por sua vez o *Legacy System Connector* foca na troca direta dos dados sobre a informação de um incidente confirmado. Esta abordagem de troca de dados através de mensagens parece ser mais adequada para a infraestrutura apresentada do RESCUER.

Porém como uma possível atividade futura, percebemos a possibilidade de registrar as informações em um banco de dados diferente do banco de dados do RESCUER. Caso seja permitido por cada SGE os relatos serão publicados e funcionarão como uma base de dados aberta e pública para consulta dos relatos de diversas emergências e diversos sistemas como o S2ID.

Esta nova abordagem permite a integração dos SGEs através da fusão de dados. Esta nova atividade pode utilizar dos arquivos de mapeamento gerados durante a fase de criação do componente. Os arquivos de mapeamento seriam utilizados

²⁰<http://www.semafora-systems.com/en/products/ontobroker/>

²¹<https://code.google.com/archive/p/ganesha/>

²²<https://sahanafoundation.org/>

²³O SPARQL é uma linguagem de query para realizar consultas em dados estruturados usando o padrão RDF. O termo SPARQL é um acrônimo recursivo que significa SPARQL Protocol and RDF Query Language

pela plataforma de integração que forneceria uma *interface* comum para consulta dos dados através de *queries* SPARQL.

APÊNDICE A FORMATOS DAS FONTES DE DADOS

A. Formato da informação CECOCO

```
id;tipo;data_inicio_operacao;
  hora_inicio_operacao;descricao;
  endereco;bairro;ponto_referencia;
  municipio;recurso;estado_final_recurso;
  data_alocado;hora_alocado;
  data_deslocamento;hora_deslocamento;
  data_atendimento;hora_atendimento;
  data_finalizado;hora_finalizado
2806553;EXPLOSÃO;09/12/2015;11:31:19;
  RELATA QUE HOVE UMA EXPLOSÃO DE GÁS
  NO BLOCO 1. DIZ AINDA QUE TEM UMA
  PESSOA COM ESCORIAÇÕES;SUSSUARANA, RUA
  DIRETA, Ulysses Guimarães, CONJUNTO
  SUMMERVILLE;Nova Sussuarana;PX. ESCOLA
  \coma NA ENTRADA DE UM POSTO DE
  GASOLINA;SALVADOR;Salvar 39;Em
  atendimento
;00/00/0000;00:00:00;00/00/0000;00:00:00;
00/00/0000;00:00:00;00/00/0000;00:00:00
```

B. Formato da informação CDD

```
id;category;group;subgroup;type;place;
  start_date;comments;fatalities;
  injured_infected;evacuated;
  estimated_total_cost;
  normalized_total_cost;end_date;
  federal_dfaa_payments;
  provincial_dfaa_payments;
  provincial_department_payments;
  municipal_costs;ogd_costs;
  insurance_payments;ngo_payments;
  utility_people_affected;magnitude
1;Incident;Technology;Fire;Non-Residential
;Ottawa ON and Hull QC;26/04/1900
00:00;Ottawa ON and Hull QC, April 26,
1900. A small chimney fire in Hull
quickly spread to neighbouring
buildings and then engulfed the Eddy
lumberyard. A strong wind carried the
fire to the Ottawa side. By the time
the fire ended, 3000 buildings had
been destroyed in the Ottawa-Hull
region.;7;0;15000;;;26/04/1900
00:00;;;;;0;0.0
```

C. Formato da informação JSON RESCUER

Listing 1. Informação de um incidente RESCUER

```
{
  "allStatus": [
    {
      "status": "INFORMED",
```

```
      "begin": "2016-06-14T13:16:26.314Z",
      "end": "2016-06-14T13:16:26.314Z"
    }
  ],
  "mergedIncidents": [ ],
  "aggregationTree": { },
  "reports": [ ],
  "analysedImages": [ ],
  "analysedVideos": [ ],
  "rowKey": "I1_b99e995d_FIRE",
  "timestamp": "2016-06-14T12:48:15.143Z",
  "updatedTimestampString": "2016-06-14T13:14:06.623Z",
  "keyword": "FIRE",
  "position": {
    "latitude": -12.9232658,
    "longitude": -38.3883258,
    "altitude": 0.0
  },
  "reliability": "low",
  "incidentIdentifier": "1_b99e995d",
  "analysedText": [ ],
  "merged": false,
  "containingIncidentIdentifier": null,
  "id": 1,
  "allKeywords": [ "FIRE" ],
  "analysedTexts": [ ],
  "helpReportsCount": 0,
  "workforceReports": [ ],
  "canceledReportsCount": 0,
  "reportsCount": 4,
  "ownReports": [ ]
}
```

AGRADECIMENTOS

Os autores deste trabalho gostariam de agradecer aos familiares que sempre apoiaram nesta árdua jornada do trabalho. Gostaríamos de agradecer também a toda equipe do RESCUER pelo trabalho apresentado e por administrar um projeto de alta qualidade. E por fim agradecimentos pela participação da professora Simone Amorim e pelo apoio oferecido para a apresentação deste trabalho.

REFERÊNCIAS

- [1] U. DHA, "Internationally agreed glossary of basic terms related to disaster management," *UN DHA (United Nations Department of Humanitarian Affairs)*, Geneva, 1992.
- [2] W. J. Petak, "Emergency management: A challenge for public administration," *Public Administration Review*, vol. 45, pp. 3-7, 1985.
- [3] B. S. Manoj and A. H. Baker, "Communication challenges in emergency response," *Communications of the ACM*, vol. 50, no. 3, pp. 51-53, 2007.
- [4] "Rescuer: Reliable and smart crowdsourcing solution for emergency and crisis management european," <http://www.rescuer-project.org/>, acessado em 30 de Julho de 2017.
- [5] "Emergency management planning guide 2010-2011," https://www.publicsafety.gc.ca/cnt/rsrscs/pblctns/mrgnc-mngmnt-pnngn/#section_one, acessado em 4 de Setembro de 2017.

- [6] “Deliverable 1.2.2: System architecture,” https://s3-sa-east-1.amazonaws.com/rescuer.deliverables/deliverables/second/RESCUER_D1.2.2.pdf, acessado em 24 de Agosto de 2017.
- [7] T. Vale, T. Arif, and L. Gasparin, “Software architecture challenges in distributed development settings: An experience report,” *WDES 2015*, p. 17, 2015.
- [8] T. R. Gruber *et al.*, “A translation approach to portable ontology specifications,” *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.
- [9] A. Gomez-Perez, M. Fernández-López, and O. Corcho, *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Science & Business Media, 2006.
- [10] R. Barros, P. Kislansky, L. Salvador, R. Almeida, M. Breyer, and L. Gasparin, “Edxl-rescuer ontology: Conceptual model for semantic integration,” in *Proceedings of the 12th International ISCRAM Conference*, 2015.
- [11] “Oasis: Advancing open standards for the information society,” https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=emergency#tc-tools, acessado em 30 de Julho de 2017.
- [12] “Edxl-de: Distribution element version 1.0,” http://docs.oasis-open.org/emergency/edxl-de/v1.0/EDXL-DE_Spec_v1.0.pdf, acessado em 4 de Agosto de 2017.
- [13] “Cap: Common alerting protocol version 1.2,” <http://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.pdf>, acessado em 4 de Agosto de 2017.
- [14] “Edxl-rm: Resource messaging version 1.0,” <http://docs.oasis-open.org/emergency/edxl-rm/v1.0/os/EDXL-RM-v1.0-OS.pdf>, acessado em 4 de Agosto de 2017.
- [15] “Edxl-sitrep: Situation reporting version 1.0,” <http://docs.oasis-open.org/emergency/edxl-sitrep/v1.0/edxl-sitrep-v1.0.pdf>, acessado em 4 de Agosto de 2017.
- [16] R. Prieto-Díaz, “A faceted approach to building ontologies,” in *Information Reuse and Integration, 2003. IRI 2003. IEEE International Conference on*. IEEE, 2003, pp. 458–465.
- [17] R. Barros¹², P. Kislansky, L. Salvador¹², R. Almeida, M. Breyer, L. G. Pedraza, and V. Vieira¹², “Edxl-rescuer ontology: an update based on faceted taxonomy approach,” 2015.
- [18] H.-J. Happel and S. Seedorf, “Applications of ontologies in software engineering,” in *Proc. of Workshop on Semantic Web Enabled Software Engineering (SWESE) on the ISWC*. Citeseer, 2006, pp. 5–9.
- [19] T. S. Dillon, E. Chang, and P. Wongthongtham, “Ontology-based software engineering-software engineering 2.0,” in *19th Australian Conference on Software Engineering (aswec 2008)*. IEEE, 2008, pp. 13–23.
- [20] L. Patil, D. Dutta, and R. Sriram, “Ontology-based exchange of product data semantics,” *IEEE Transactions on automation science and engineering*, vol. 2, no. 3, pp. 213–225, 2005.
- [21] I. F. Cruz and H. Xiao, “The role of ontologies in data integration,” *Engineering intelligent systems for electrical engineering and communications*, vol. 13, no. 4, p. 245, 2005.
- [22] “Rdf and json-ld usecases,” https://www.w3.org/2013/dwbp/wiki/RDF_AND_JSON-LD_UseCases, acessado em 28 de Agosto de 2017.
- [23] J. Slepicka, C. Yin, P. A. Szekeley, and C. A. Knoblock, “Kr2rml: An alternative interpretation of r2rml for heterogenous sources.” in *COLD*, 2015.
- [24] “Google maps web services: Geocodificação api,” <https://developers.google.com/maps/documentation/geocoding/intro?hl=pt-br>, acessado em 30 de Agosto de 2017.
- [25] “Deliverable 1.5.1: Integrated platform demonstrator,” https://s3-sa-east-1.amazonaws.com/rescuer.deliverables/deliverables/first/RESCUER_D1.5.1.pdf, acessado em 24 de Agosto de 2017.
- [26] L. Santos, M.-A. Sicilia, and S. Padrino, “Ontologies for emergency response: effect-based assessment as the main ontological commitment,” *Metadata and Semantic Research*, pp. 93–104, 2011.
- [27] R. Barros, “EDXL-RESCUER: Interoperabilidade Semântica Entre Sistemas de Resposta à Emergência,” Master’s thesis, Universidade Federal da Bahia, Salvador, Bahia, Brasil, 2016.