O Middleware E2BS Aplicado ao Acionamento de Unidades Móveis em Cidades Inteligentes para Situações de Emergência

Sediane C. L. Hernandes[1], Marcelo Eduardo Pellenz[2], Alcides Calsavara[2]

[1]Universidade Tecnológica Federal do Paraná (UTFPR)
80.053-525 – Guarapuava – PR – Brasil

[2]Programa de Pós-Graduação em Informática (PPGIa)
Pontifícia Universidade Católica do Parana (PUC-PR)
80.215-901 – Curitiba – PR – Brasil

sedianec@utfpr.edu.br, {marcelo, alcides}@ppgia.pucpr.br

Abstract. *This article describes the implementation of the middleware Event to Best Subscribers (E2BS), which presents an event model in which only the most appropriate subscribers are selected to receive messages. The middleware was designed for Smart Cities scenarios and was used to activation of mobile units in emergency situations. In this way, the different architectures and implementation approaches of the E2BS middleware are described, as well as the results of its simulation.*

Resumo. *Este artigo descreve a implementação do middleware E2BS (Event to Best Subscribers), o qual apresenta um modelo de eventos em que um subconjunto de assinantes mais adequados são selecionados para o recebimento das mensagens. O middleware foi projetado para cenários de Cidades Inteligentes e foi utilizado para o acionamento de unidades móveis em situações de emergência. Desta forma, as diferentes arquiteturas e abordagens de implementação do middleware E2BS são descritas, bem como os resultados de sua simulação.*

1. Introdução

Atualmente, objetos do dia a dia, que normalmente não são considerados computadores, apresentam capacidade computacional e conectividade de rede. Com isso, esses dispositivos geram, trocam e consomem dados com o mínimo de interação humana (ROSE; ELDRIDGE; LYMAN, 2015). Esse cenário de refere a Internet das Coisas (*Internet of Things - IoT*). Em Internet das Coisas uma arquitetura poderia ser definida como um sistema descentralizado, fracamente acoplado de objetos inteligentes (*smart objects*).

IoT permite que objetos inteligentes se conectem a Internet. Para (GYRARD; SERRANO; PLATFORMS, 2015), Internet das Coisas é uma rede de objetos físicos que contêm tecnologia embutida para comunicação e sensoriamento ou interação com seus estados internos ou o ambiente externo. O interessante mesmo seria chamar Internet das Coisas de "Internet dos Objetos" ou "Internet dos Objetos Inteligentes". Estes objetos conectados à Internet poderiam ser controlados remotamente e ser

provedores e consumidores de serviços de/e para outros objetos ou aplicações, proverem dados para análise e reagirem a eventos. Tudo isso sendo possível porque esses objetos são endereçados unicamente.

Desta forma, com o uso dos recursos desses objetos (e. g., capacidade de processamento e comunicação) várias aplicações para Cidades Inteligentes surgem como, por exemplo, aplicações para monitoração da saúde estrutural de construções, para a gerência de lixo inteligente, para a mobilidade inteligente, para o transporte inteligente, para a monitoração de ruído e da qualidade do ar, para o atendimento a situações de emergência. IoT pode aumentar o desenvolvimento dessas aplicações que reagem a eventos e geram uma grande quantidade e variedade de dados.

Além disso, em aplicações IoT, interações entre objetos podem acontecer quando um objeto entra na faixa de comunicação de outros objetos levando a geração espontânea de eventos (RAZZAQUE et al., 2016). Por exemplo, um usuário de um smartphone pode vir a entrar em contato com uma TV ou refrigerador ou uma máquina de lavar e gerar eventos sem o envolvimento do usuário. Normalmente, em IoT, uma interação com um objeto significa que um evento é gerado e é "empurrado" para o sistema sem muita atenção humana. Em IoT milhares de objetos inteligentes podem interagir em uma escala muito maior do que os sistemas em rede tradicionais (RAZZAQUE et al., 2016). Interação espontânea de eventos entre um número muito grande de dispositivos pode produzir uma grande quantidade de eventos que precisam ser gerenciados de alguma forma.

Consequentemente, um middleware a ser utilizado no contexto de Cidades Inteligentes pode ser utilizado para gerenciar o grande número de eventos que os objetos inteligentes, presentes no ambiente urbano, geram espontaneamente. Uma rede de objetos inteligentes pode ser formada nesse cenário e um middleware ofereceria serviços comuns para as aplicações e facilitaria seu desenvolvimento, especialmente por gerenciar esses eventos. Um dos serviços que se beneficiaria do middleware seria o atendimento a situações de emergência. Um serviço de emergência quando acionado deve selecionar as unidades móveis mais apropriadas para o atendimento.

Vários middlewares *publish-subscribe* foram pesquisados (e. g., EMMA (MUSOLESI; MASCOLO; HAILES, 2006), Hermes (PIETZUCH; BACON, 2002), Mires (SOUTO et al., 2006), PSWare (LAI; CAO; ZHENG, 2009), PRISMA (SILVA et al., 2014), RUNES (COSTA et al., 2006), SensorBus (ADMILSON et al., 2005), Siena (CARZANIGA; ROSENBLUM; WOLF, 2001), Steam (MEIER; CAHILL, 2002) e TinyDDS (BOONMA; SUZUKI, 2011)) visando encontrar um que se adequasse ao contexto de atendimento a situações de emergência em Cidades Inteligentes. Uma análise mais detalhada desses middlewares pode ser encontrada em (HERNANDES; CALSAVARA; LIMA JR., 2017a). Entretanto, os middlewares se mostraram pouco flexíveis em relação ao modelo de envio de mensagens que implementam, no qual todos os assinantes subscritos recebem os eventos, o que não é apropriado para o cenário pretendido. Desta forma, o middleware E2Bs (*Event to Best Subscribers*) foi proposto visando o envio de eventos somente aos assinantes mais adequados. Assim, o objetivo deste artigo é apresentar o middleware E2BS (HERNANDES; CALSAVARA; LIMA JR., 2017a), seu objetivo, fundamentos, arquitetura, estratégias de alocação de objetos inteligentes, abordagens de implementação e, principalmente, os principais resultados de

sua implementação e validação no atendimento a situações de emergência. O artigo é dividido como segue. A seção 2 apresenta Internet das Coisas e Cidades Inteligentes. Na seção 3 descreve-se o middleware E2BS. O atendimento a situações de emergência com o uso do middleware E2BS é contemplado na seção 4 e algumas considerações finais são descritas na seção 5.

2. Internet das Coisas e Cidades Inteligentes

Internet das Coisas é um paradigma de comunicação no qual os objetos da vida diária podem ser equipados com microcontroladores e rádio transmissores para comunicação digital e pilhas de protocolos adequados que irá torná-los capazes de se comunicar uns com os outros e com os usuários, tornando-se parte integrante da Internet. Esses objetos inteligentes podem ser objetos que possuem um chip RFID, sensores, elementos atuadores e outras tecnologias. Um objeto inteligente é um objeto (físico ou digital) com capacidade de sensoriamento, processamento e rede (KORTUEM et al., 2010). Objetos inteligentes carregam parte da lógica da aplicação, eles podem "sentir", registrar e interpretar o que está ocorrendo dentro de si mesmos e do mundo, agir por conta própria, se intercomunicando uns com os outros e trocando informações com as pessoas (KORTUEM et al., 2010). Exemplos de objetos inteligentes incluem eletrodomésticos, câmeras de segurança, sensores de monitoramento, veículos, entre outros.

Assim, IoT permite pessoas e coisas estarem conectadas a qualquer hora, qualquer lugar, com qualquer coisa e qualquer um, idealmente utilizando qualquer rede e qualquer serviço (VERMESAN et al., 2009). IoT é simplesmente uma infraestrutura baseada em TCP/IP que conecta dispositivos (objetos inteligentes) para uma aplicação específica através de um protocolo (JALOUDI, 2015). O protocolo transfere dados do dispositivo sobre TCP/IP baseado em redes tais como a Internet, a Intranet, WiFi, WiMax, para uma aplicação em um centro de controle. Sensores sem fio, por exemplo, podem transferir dados de veículos em rodovias utilizando o protocolo *Constrained Application Protocol* (CoAP) para a gerência de tráfego inteligente. A tecnologia RFID pode ser utilizada para o monitoramento da saúde de um paciente e eventualmente chamar uma assistência remota (AMENDOLA et al., 2014).

Cidades Inteligentes e IoT possuem ideologias similares, mas com diferentes origens. Cidades Inteligentes surgiram para resolver os problemas das cidades modernas ou grandes cidades, enquanto que IoT surgiu inicialmente para conectar fisicamente dispositivos a Internet (VERMESAN et al., 2009). IoT é conduzida por avanços tecnológicos, não por aplicações ou necessidades dos usuários. IoT vai em direção as necessidades dos usuários quando é utilizado em um contexto específico, como a gerência do lixo em uma grande cidade, por exemplo. Em resumo, Internet das Coisas e Cidades Inteligentes são conceitos estreitamente relacionados. Cidades Inteligentes não existiriam sem IoT e, da mesma forma, IoT não teria sua devida importância sem seu uso em um contexto urbano. Assim, serviços de emergência podem se beneficiar do uso de IoT em um contexto urbano: *drones* poderiam supervisionar um atendimento de emergência; objetos inteligentes poderiam ser espalhados na cidade para detectar situações de emergência ou realizar o próprio acionamento das unidades móveis, ou; unidades móveis poderiam ser equipadas com sensores para captar dados do ambiente.

3. O middleware E2BS

O middleware E2BS (*Event to Best Subscribers)* (HERNANDES; CALSAVARA; LIMA JR., 2017a) implementa um modelo de eventos em que cada evento é enviado a um subconjunto de assinantes mais apropriados. Assinantes se inscrevem com o middleware e este envia os eventos, através de *broker(s),* as unidades móveis adequadas. Uma unidade móvel, ou seja, um assinante, é um automóvel utilitário que atende as situações de emergência possuindo equipamentos de emergência, pessoal qualificado e materiais específicos para isso. Faz uso do middleware E2BS para emissão de subscrições, confirmação de atendimento a situações de emergência, entre outros. O *broker*, por outro lado, é um processo de software integrante do middleware que toma a decisão sobre quais unidades móveis devem se deslocar para atender a um chamado de emergência, além de realizar outras funções relacionadas a esta. Cada *broker* pode ser instalado, sendo desejável, em um objeto inteligente com capacidade de processamento, memória e comunicação.

3.1 Arquitetura do middleware

A arquitetura do middleware E2BS pode assumir uma organização centralizada ou descentralizada, sendo composta de unidades móveis e de *brokers*. Na organização centralizada, existe um único *broker* que é responsável pela tomada de decisão sobre quais unidades móveis devem ser acionadas, enquanto que na organização descentralizada, os *brokers* são distribuídos pela cidade e trabalham em conjunto para essa tomada de decisão. A Figura 1 apresenta as duas formas de organização da arquitetura do middleware E2BS.

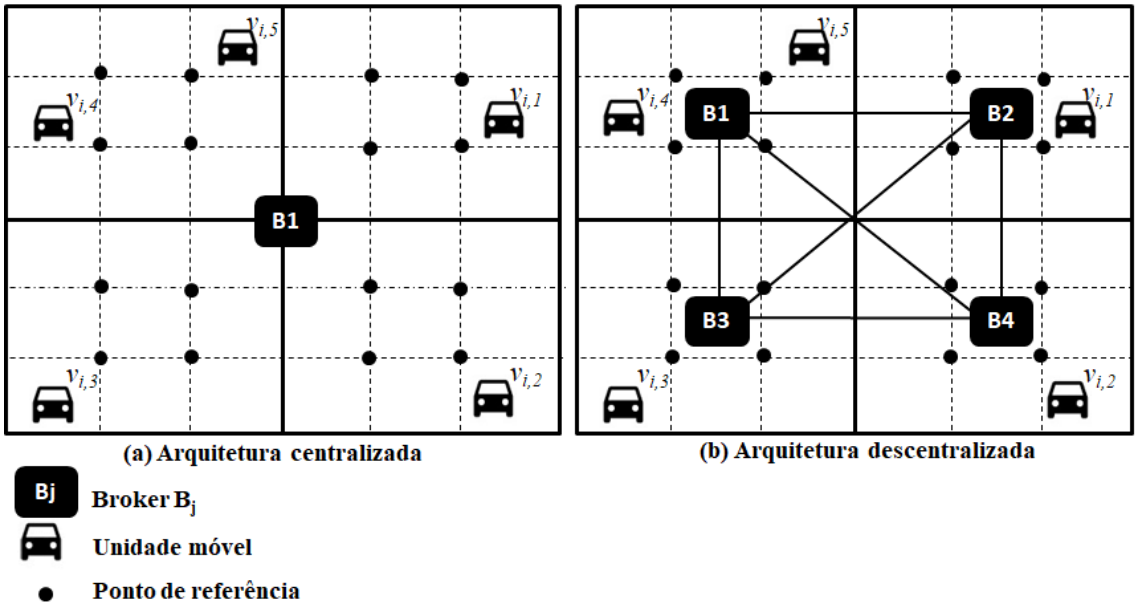

Figura 1 – Arquitetura centralizada (a) e descentralizada (b) do middleware.

Na arquitetura centralizada, o *broker* pode ficar de maneira centralizada na cidade. Já na arquitetura descentralizada, os *brokers* devem ser alocados de duas formas: alocação distribuída por regiões da cidade (um *borker* para cada região) ou alocação baseada em *cluster* (os *broker* são alocados em uma datacenter de serviço de

Computação em Nuvem). A Figura 2 ilustra mais claramente o cenário dessas arquiteturas, sendo uma antena de comunicação por região da cidade. A arquitetura descentralizada com alocação de *brokers* em cluster é parecida com a segunda arquitetura apresentada na Figura 2, sendo que os *brokers* ficam em um local único (datacenter) ligados por uma rede de alta velocidade.

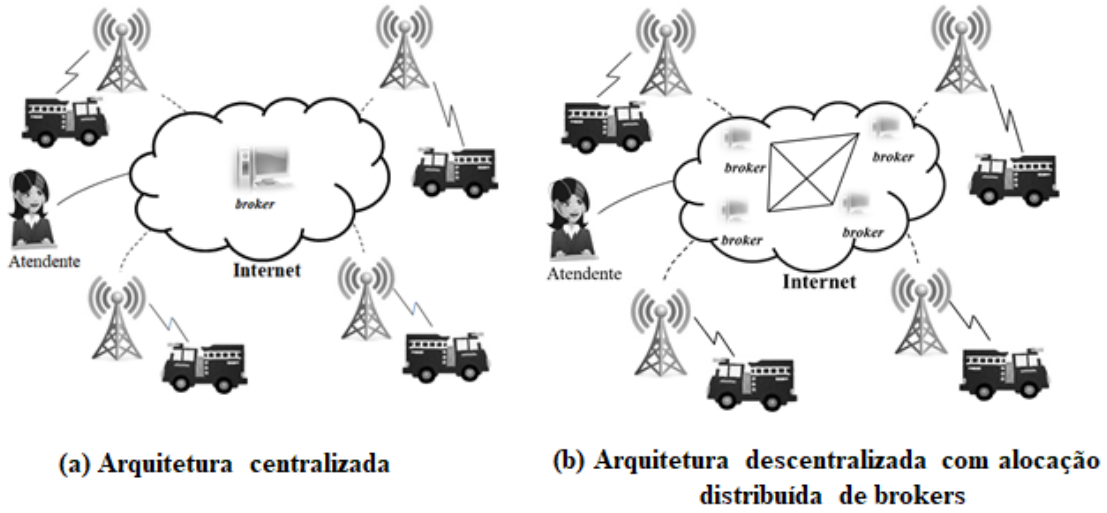

Figura 2 – Cenário das arquiteturas centralizada e descentralizada.

Quanto a forma de comunicação, tanto na arquitetura centralizada quanto na arquitetura descentralizada a comunicação segue o padrão de comunicação *Device-to-Cloud* para comunicação de objetos inteligentes em IoT (IETF, 2015), (TSCHOFENIG et al., 2015). Nesse modelo de comunicação, os dispositivos se conectam diretamente a um serviço de nuvem na Internet, semelhante a um provedor de serviço de aplicação para troca de dados e controle do tráfego das mensagens. Essa abordagem possui como vantagem o fato da existência de um mecanismo de comunicação como Ethernet com fio ou conexões Wi-fi para estabelecer uma conexão entre o dispositivo e a rede IP, a qual se conecta, posteriormente, a um serviço de nuvem.

3.2 Atualização dos níveis de adequação das unidades móveis

Cada *broker* mantém os níveis de adequação das unidades móveis de sua região para posterior acionamento das mesmas. O nível de adequação da unidade móvel diz respeito a um valor numérico (calculado com base na disponibilidade e localização da unidade móvel e das condições de trânsito entre o percurso da unidade móvel a cada ponto de referência da cidade) que indica quão "preparada" ou "adequada" a unidade móvel está para atender à situação de emergência. Detalhes de como os níveis de adequação são calculados, bem como outras especificações do protocolo de troca de mensagens e dos algoritmos utilizados para tomada de decisão pelos *brokers* podem ser encontrados em Hernandes, Calsavara e Lima Jr. (2017b). Quanto maior esse valor, mais apropriada ela se encontra para responder a um chamado. Por consequente, é importante para o *broker* manter atualizado os níveis de adequação das unidades móveis de sua região para melhorar a precisão do resultado do acionamento das mesmas.

Desta forma, os níveis de adequação nos *brokers* podem ser atualizados seguindo a abordagem reativa ou pró-ativa. Na abordagem reativa, os níveis são

atualizados somente no momento em que ocorre a situação de emergência. Os níveis de adequação das unidades móveis têm que ser calculados a partir deste momento. Na abordagem pró-ativa, os níveis de adequação são atualizados constantemente facilitando, desta forma, o acionamento das unidades móveis quando uma situação de emergência acontece. Os níveis de adequação das unidades móveis já estão à disposição para que o *broker* escolha adequadamente as unidades móveis necessárias.

3.3 Princípio do middleware E2BS

O princípio do middleware E2BS constitui-se da criação de um conjunto de redes overlay sobre a rede física que conecta *broker(s)* e unidades móveis. Cada rede overlay é representada como uma árvore lógica de disseminação de eventos. Para cada ponto de referência e tipo de serviço (par (p, s)), uma rede overlay é criada. Se a abordagem centralizada for utilizada para um único serviço, como no exemplo da Figura 1 (a), serão criadas 16 redes overlay, uma para cada ponto de referência, pois há somente um serviço disponível. Essa rede será formada pelo *broker* B_1 e pelas unidades móveis que se encontram na sua região ($v_{i,1}$, $v_{i,2}$, $v_{i,3}$, $v_{i,4}$, $v_{i,5}$). Entretanto, se a abordagem descentralizada for utilizada, como no exemplo da Figura 1 (b), o mesmo número de redes overlay será criado; contudo, uma rede overlay para cada ponto de referência pertencente ao *broker*. Logo, cada *broker* terá 4 redes overlay. A Figura 3(b) apresenta uma árvore lógica de disseminação de eventos para um ponto de referência de B_1.

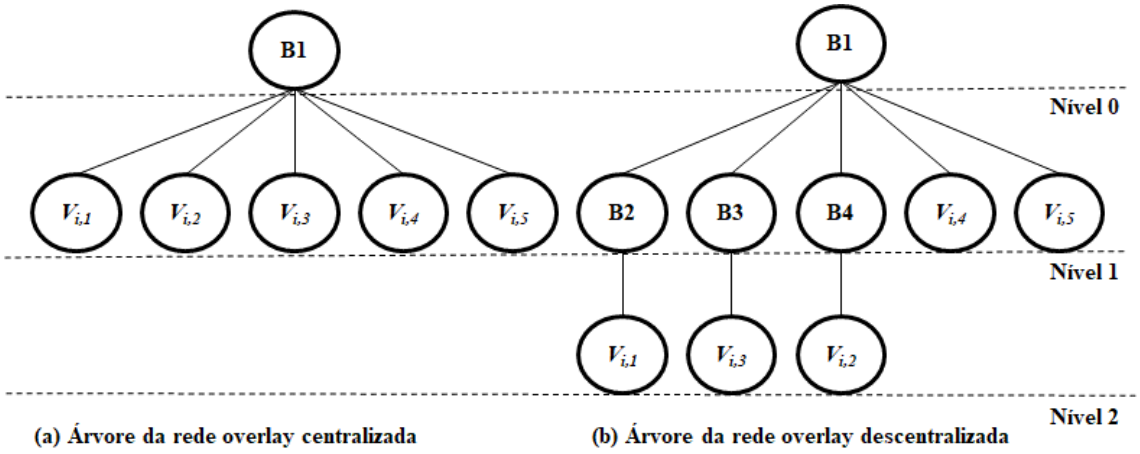

Figura 3 – (a) Árvore lógica de disseminação de eventos de B_1 (arquitetura centralizada); (b) Árvore lógica de disseminação de eventos de B_1 (arquitetura descentralizada).

A árvore lógica de disseminação de eventos, a qual representa a topologia da rede overlay, é utilizada para que o evento chegue as unidades móveis que precisam ser acionadas. Na árvore lógica da Figura 3(a), B_1 é o *broker* raiz. Da mesma forma na Figura 3 (b), B_1 é o *broker* raiz e B_2, B_3 e B_4 são os *brokers* intermediários. A topologia das redes overlay muda a medida que as unidades móveis se movimentam pela cidade. A mobilidade é tratada como parte do middleware.

4. Atendimento a situações de emergência com o uso do middleware E2BS

O middleware E2BS foi implementado em 6 (seis) versões considerando as arquiteturas centralizada e descentralizada e as abordagens de atualização de estado de

forma reativa e pró-ativa. Para a validação do middleware foi utilizado o simulador Sinalgo (*Simulator for Network Algorithms*).

4.1 Pacotes principais do middleware

Dois pacotes principais foram desenvolvidos no middleware: o pacote de mensagens e o pacote de nós. O pacote de mensagens contém as mensagens que são enviadas entre os nós e precisam ser tratadas (e.g., mensagens de subscrição e de notificação de eventos). O pacote de nós contém a implementação dos nós propriamente ditos (e.g., os *brokers* e as unidades móveis que enviam, recebem e tratam as mensagens recebidas). Além disso, foram implementados um modelo de mobilidade entre os nós, um modelo de transmissão de mensagens, modelos de distribuição dos nós na área de simulação, modelos de conectividade entre os nós, além de *timers* para o disparo de eventos.

As mensagens enviadas por um *broker* a(s) unidade(s) móvel(is) são:

- *invite*: mensagem de convite enviada em *broadcast* por um *broker* para todas as unidades móveis sob sua área de cobertura. Os campos dessa mensagem são: número de sequência e identificação do *broker*.
- *notification*: mensagem de acionamento para atendimento a situação de emergência. Os campos dessa mensagem são: número de sequência, identificação do *broker* que envia a mensagem, identificação da unidade móvel que irá atender a emergência, serviço e ponto da situação de emergência.

As mensagens enviadas por uma unidade móvel para um *broker* são:

- *subscribe*: mensagem de subscrição para um serviço respondendo a uma mensagem *invite*. Os campos dessa mensagem são: número de sequência, identificação do *broker* que enviou *invite*, identificação da unidade móvel que está enviando *subscribe*, ponto em que a mesma se encontra, serviço e disponibilidade.
- *bind*: mensagem de vínculo emitida periodicamente informando que a unidade móvel mantém-se na região de abrangência do *broker*. Enviada somente se a mensagem *subscribe* for recebida pela unidade móvel. Os campos dessa mensagem são: número de sequência, identificação do *broker,* identificação da unidade móvel, ponto em que a mesma se encontra, serviço e disponibilidade.
- *bye-bye*: mensagem que sinaliza que a unidade móvel está saindo da região de abrangência do *broker*. Os campos dessa mensagem são: identificação do *broker* que a unidade móvel está se "desligando" e identificação da unidade móvel.
- *ack_notification*: mensagem de confirmação de acionamento em resposta a uma mensagem *notification.* Os campos dessa mensagem são: número de sequência, identificação do *broker* que enviou *notification*, identificação da unidade móvel acionada e serviço.

As mensagens enviadas entre *brokers* são:

- *publish*: mensagem enviada por um *broker* raiz a um *broker* intermediário solicitando ao *broker* intermediário que acione uma de suas unidades móveis para atendimento. Os campos dessa mensagem são: número de sequência,

identificação do *broker* raiz, identificação do *broker* intermediário, serviço e ponto de referência da situação de emergência.

- *ack_publish:* mensagem de confirmação de acionamento de unidade móvel enviada por um *broker* intermediário a um *broker* raiz, em resposta a uma mensagem *publish.* Os campos dessa mensagem são: número de sequência, identificação do *broker* que emitiu *publish*, identificação do *broker* que está confirmando a mensagem de *publish*, serviço e número de unidades móveis acionadas (1 –uma acionada ou 0 – nenhuma acionada).
- *update*: mensagem enviada por um *broker* intermediário a um *broker* raiz contendo os níveis de adequação atualizados das unidades móveis presentes em sua área de cobertura. Os níveis de adequação enviados dizem respeito aos maiores valores entre todas as unidades móveis de um *broker* intermediário para cada ponto de referência pertencente ao *broker* raiz. Os campos dessa mensagem são: identificação do *broker* recebedor, serviço e níveis de adequação atualizados.

Outro tipo de mensagem que é enviada por um agente externo (atendente da situação de emergência) a um *broker* raiz (responsável pela região em que acontece a situação de emergência) é a mensagem *publish_activation.* Essa é uma mensagem de ativação de situação de emergência. Ela indica que o *broker* deve acionar uma quantidade *n* de unidades móveis, pertencentes a determinado serviço, de sua árvore lógica de disseminação de eventos para atender a emergência em um ponto específico. A cada unidade móvel acionada, o atendente que disparou a mensagem recebe uma mensagem de confirmação *ack_publish_activation*. O agente, desta forma, controla se o pedido é cumprido. Enquanto o pedido não é cumprido, o agente envia novas mensagens de *publish_activation* de 2 em 2 minutos durante o período de 1 hora. Novas mensagens de *publish_activation* são geradas em intervalos aleatórios de tempo. Assim, se durante o reenvio de um pedido não atendido, uma nova mensagem é gerada, a mesma é colocada em uma fila para posterior envio.

4.2 Parâmetros de simulação

Para as simulações alguns parâmetros de configuração do simulador e do cenário precisaram ser definidos. Como parâmetros de configuração do simulador ficou definido: (1) Simulação do tipo síncrona, ou seja, com mobilidade: os *brokers* seguem o modelo de mobilidade *NoMobility* e as unidade móveis uma extensão do *PerfectRWP*, no qual as unidades móveis selecionam um ponto aleatório da área de simulação (ponto da situação de emergência) e se movimentam a uma velocidade constante (*NodeSpeed=1*) até ele; (2) Envio de mensagens enquanto o nó estiver recebendo-as; (3) Atualização dos nós a cada *round* (rodada) do simulador; (3) Modelo de transmissão de mensagens do tipo *ConstantTime* (*ConstantTime* = 1.0), ou seja, as mensagens enviadas são transmitidas no próximo *round* do simulador; (4) Distribuição dos *brokers* na área de simulação (x=1000 e y=1000) em grade e das unidades móveis de forma randômica.

Para as simulações do cenário escolhido, os parâmetros foram: (i) Cada *round* do simulador equivalendo a 1 segundo; (ii) Cada simulação correspondendo a 1 dia (86.400 segundos ou *rounds*); (iii) Área de cobertura dos *brokers* em 100% da área urbana; (iv) Tipo de serviço simulado como sendo o bombeiro (unidades móveis possuem um posto

de atendimento e se movimentam até a situação a de emergência quando são acionadas, retornando posteriormente ao seu respectivo posto de atendimento); (v) Intervalo de acionamento do serviço: variável aleatória com distribuição entre 30 minutos a 2 horas*; (vi) Quantidade de unidades móveis para atendimento as emergências: variável aleatória entre 1 e 3*; (vii) Tempo de deslocamento dependente da distância entre a unidade móvel e o ponto da situação de emergência e das condições de trânsito (leve, moderado e intenso): depois de obtida a distância da unidade móvel à situação de emergência, esta é dividida por 60, pois para cada minuto foi determinado que a unidade móvel deveria andar 1Km com trânsito leve; entretanto, 500 metros com trânsito moderado e 250 metros com trânsito intenso*; (viii) Tempo de atendimento da unidade móvel: variável aleatória com distribuição entre 20 minutos a 8 horas*; (ix) Trânsito considerado leve, moderado ou intenso, dependente dos horários do dia, para o cálculo dos níveis de adequação; (x) Número de pontos de referência por *broker* definido em 4; (xi) Ponto da situação de emergência escolhido aleatoriamente no *broker* acionado com *brokers* acionados de forma circular; (xii) Tempo de envio de mensagens de *invite* pelo *broker* as unidades móveis em intervalos de 1 minuto (para as abordagens pró-ativas); (xiii) Tempo de envio de mensagens de *bind* pelas unidades móveis ao *broker* em intervalos de 5 minutos (para as abordagens pró-ativas); (xiv) Oito simulações para cada abordagem implementada; (xv) Atraso no envio de pacotes em 2 segundos, exceto no envio de pacotes entre *brokers* na abordagem descentralizada com alocação de *brokers* em *cluster*, o qual foi considerado como sendo em 0 (zero) segundos; (xvi) Representação do tempo na forma ***hora:minuto:segundo***. Para a definição dos parâmetros dos cenários (os quais são marcados com asterisco), a Corporação XYZ e a Corporação ZWV (nomes fictícios devido a termo de confidencialidade firmado) forneceram um resumo dos dados reais das situações de emergência do ano de 2016 em setembro de 2017.

4.3 Simulações

Quarenta e oito simulações foram executadas para a coleta de dados, oito para cada arquitetura e abordagem. Para o tempo de tomada de decisão em cada *broker* foi considerado o tempo necessário, em segundos, para a escolha das unidades móveis que devem atender a situação de emergência. Para o tempo de resposta de um pedido ao atendente foi considerado o intervalo de tempo entre o envio de *publishactivation* e o recebimento da última mensagem de confirmação de acionamento de unidade móvel pelo *broker* acionado pelo atendente. O atraso foi utilizado para demonstrar a diferença entre as abordagens pró-ativa e reativa. Para ambas as abordagens, para os cálculos dessas médias, foi considerado somente acionamentos concluídos com sucesso na primeira tentativa de acionamento pelo atendente. Acionamentos concluídos com sucesso significam que a situação de emergência foi atendida plenamente, ou seja, a unidade móvel se deslocou e prestou o serviço necessário voltando posteriormente ao seu posto de atendimento. O método de validação utilizado foi a comparação de cenários e as métricas utilizadas foram o percentual de sucesso no atendimento a eventos relacionado ao número de unidades móveis, o número de unidades móveis e sua relação quanto ao número de tentativas para conclusão dos acionamentos com sucesso, as médias do tempo de tomada de decisão pelo *brokers* e as médias do tempo de resposta dos pedidos ao atendente.

4.3.1 Simulação da alocação centralizada

Os dados obtidos da simulação com alocação centralizada do *broker*, abordagem pró-ativa, encontram-se na Tabela 1.

De acordo com os dados das simulações da Tabela 1, pode-se observar que normalmente quanto maior o número de unidades móveis, maior é o percentual de sucesso no atendimento a eventos. Isso ocorre principalmente devido a média do tempo de atendimento e deslocamento das unidades móveis e do tempo médio entre um acionamento e outro. Se o intervalo de tempo de acionamento ou as médias aumentassem é provável que essa afirmação não seria possível. Ainda de acordo com a Tabela 1, pode-se observar que um atendimento com um tempo maior faz com que mais tentativas sejam realizadas para que o acionamento seja concluído. Esse fator irá se fazer presente em todas as simulações. Se, por exemplo, todas as unidades móveis acionadas precisarem de 8 horas de atendimento, mesmo com um número maior de unidades móveis a disposição, o percentual de sucesso no atendimento dos eventos e outras variáveis relacionadas irão sofrem aumento (e.g., número de tentativas para conclusão dos acionamentos) ou redução em seus valores (e.g., número de unidades móveis acionadas). O destaque dessa abordagem é que o tempo médio de tomada de decisão pelo *broker* (não mostrado na Tabela 1) foi de 0 (zero) segundos porque os níveis de adequação das unidades móveis já estavam calculados. O *broker* precisa somente visitar as árvores lógicas de disseminação de eventos para descobrir qual o maior nível de adequação e acionar as respectivas unidades móveis.

Tabela 1– Abordagem pró-ativa (centralizada).

	Sim. 1	Sim. 2	Sim. 3	Sim. 4	Sim. 5	Sim. 6	Sim. 7	Sim. 8
Número de unidades móveis	1	2	3	4	5	6	7	8
Tempo médio entre um acionamento e outro	01:17:56	01:09:03	01:11:33	01:03:34	01:02:56	01:02:56	01:11:43	01:11:43
Total de mensagens de acionamentos	18	20	19	23	22	20	19	23
Número de unidades móveis solicitadas	26	39	43	48	40	34	37	47
Número de unidades móveis acionadas	9	19	22	34	35	32	35	47
Percentual de sucesso no atendimento a eventos	35%	49%	51%	71%	88%	94%	95%	100%
Média do tempo de resposta dos pedidos ao atendente	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08
Média do tempo de deslocamento	00:12:56	00:15:09	00:16:49	00:11:10	00:16:59	00:14:06	00:11:55	00:12:25
Média do tempo de atendimento	02:03:04	01:54:37	02:53:16	02:27:22	02:28:27	02:21:34	02:41:08	02:40:24
Total de tentativas realizadas para conclusão dos acionamentos	335	441	395	315	227	51	125	27

Os dados obtidos da simulação com alocação centralizada do *broker*, abordagem reativa, encontram-se na Tabela 2.

Tabela 2 – Abordagem reativa (centralizada).

	Sim. 1	Sim. 2	Sim. 3	Sim. 4	Sim. 5	Sim. 6	Sim. 7	Sim. 8
Número de unidades móveis	1	2	3	4	5	6	7	8
Tempo médio entre um acionamento e outro	01:11:08	00:53:54	01:10:28	01:06:45	01:08:11	01:11:06	00:57:17	01:10:49
Total de mensagens de acionamentos	20	26	20	21	21	20	25	20
Número de unidades móveis solicitadas	34	56	40	38	42	31	45	44
Número de unidades móveis acionadas	8	16	18	32	35	29	38	44
Percentual de sucesso no atendimento a eventos	24%	29%	45%	84%	83%	94%	84%	100%
Média do tempo de tomada de decisão pelo broker	00:00:04	00:00:04	00:00:04	00:00:04	00:00:04	00:00:04	00:00:04	00:00:04
Média do tempo de resposta dos pedidos ao atendente	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08	00:00:08
Média do tempo de deslocamento	00:11:21	00:15:24	00:14:24	00:15:31	00:17:25	00:13:45	00:17:58	00:12:31
Média do tempo de atendimento	02:58:52	02:46:40	03:04:21	02:29:29	02:20:26	02:47:05	02:59:43	02:08:22
Total de tentativas realizadas para conclusão dos acionamentos	492	549	437	270	192	86	227	47

Comparando-se os dados da Tabela 1 com a Tabela 2, pode-se perceber que a média do tempo de tomada de decisão pelo *broker* (não mostrada na Tabela 1) aumenta na abordagem reativa, uma vez que o *broker* acionado não possui informações dos níveis de adequação das unidades móveis para a tomada de decisão sobre quais unidades móveis precisam ser acionadas. Como consequência, o *broker* envia mensagens de *invite* para que as unidades móveis se inscrevam com ele para esta escolha. Por isso, a média do tempo de tomada de decisão pelo *broker* é maior na abordagem reativa.

4.3.2 Simulação da alocação distribuída por regiões da cidade

Os dados obtidos da simulação com alocação distribuída de *brokers* por regiões da cidade, abordagem pró-ativa encontram-se na Tabela 3.

Tabela 3 - Abordagem pró-ativa (alocação distribuída de *brokers*).

	Sim. 1	Sim. 2	Sim. 3	Sim. 4	Sim. 5	Sim. 6	Sim. 7	Sim. 8
Número de unidades móveis	1	2	3	4	5	6	7	8
Tempo médio entre um acionamento e outro	01:13:45	01:04:36	01:11:06	01:07:16	01:13:46	01:13:02	01:10:24	01:10:24
Total de mensagens de acionamentos	20	22	20	21	19	19	20	20
Número de unidades móveis solicitadas	40	47	44	43	35	33	37	37
Número de unidades móveis acionadas	6	15	28	29	33	33	37	37
Percentual de sucesso no atendimento a eventos	15%	32%	64%	67%	94%	100%	100%	100%
Média do tempo de resposta dos pedidos ao atendente	00:00:10	00:00:12	00:00:11	00:00:11	00:00:11	00:00:11	00:00:11	00:00:10
Média do tempo de deslocamento	00:14:52	00:17:58	00:15:45	00:16:05	00:12:36	00:13:33	00:15:57	00:10:13
Média do tempo de atendimento	02:52:07	02:38:22	02:11:18	02:13:04	01:51:54	02:29:36	02:44:16	02:20:20
Total de tentativas realizadas para conclusão dos acionamentos	431	477	332	305	43	56	0	0

Os dados obtidos da simulação com alocação distribuída de *brokers* por regiões da cidade, abordagem reativa, encontram-se Tabela 4.

Tabela 4 – Abordagem reativa (alocação distribuída de *brokers*).

	Sim. 1	Sim. 2	Sim. 3	Sim. 4	Sim. 5	Sim. 6	Sim. 7	Sim. 8
Número de unidades móveis	1	2	3	4	5	6	7	8
Tempo médio entre um acionamento e outro	01:06:24	01:14:38	01:04:51	01:11:39	01:03:44	01:17:10	01:05:40	01:05:40
Total de mensagens de acionamentos	21	20	22	20	22	18	22	21
Número de unidades móveis solicitadas	43	48	51	37	40	37	38	39
Número de unidades móveis acionadas	7	14	25	31	34	31	36	39
Percentual de sucesso no atendimento a eventos	16%	29%	49%	84%	85%	84%	95%	100%
Média do tempo de tomada de decisão pelos brokers	00:00:08	00:00:08	00:00:08	00:00:07	00:00:07	00:00:07	00:00:05	00:00:06
Média do tempo de resposta dos pedidos ao atendente	00:00:12	00:00:12	00:00:12	00:00:11	00:00:11	00:00:11	00:00:09	00:00:10
Média do tempo de deslocamento	00:07:43	00:11:29	00:15:24	00:12:10	00:17:43	00:16:14	00:11:31	00:14:36
Média do tempo de atendimento	02:53:26	03:19:57	02:19:43	02:12:06	02:01:39	02:43:16	02:30:21	02:08:13
Total de tentativas realizadas para conclusão dos acionamentos	569	490	438	197	210	120	87	28

Observando os dados da Tabela 3 e da Tabela 4, os dados demonstram basicamente o mesmo comportamento da arquitetura centralizada. Entretanto, os valores das médias dos tempos de resposta dos pedidos ao atendente são maiores na arquitetura descentralizada. Esse aumento dos segundos se deve ao fato de que existe um nível a mais na árvore lógica de disseminação de eventos que é formada na arquitetura descentralizada com alocação distribuída de *brokers*. Algumas vezes um *broker* raiz tem que acionar um *broker* intermediário e este uma ou mais unidades móveis para que a situação de emergência seja atendida. Isso faz com que mais mensagens sejam enviadas e o tempo de resposta aumente.

4.3.3 Simulação da alocação baseada em *cluster*

Os dados obtidos da simulação com alocação de *brokers* em *cluster*, abordagem pró-ativa encontram-se na Tabela 5.

Tabela 5 – Abordagem pró-ativa (alocação de *brokers* em *cluster*).

	Sim. 1	Sim. 2	Sim. 3	Sim. 4	Sim. 5	Sim. 6	Sim. 7	Sim. 8
Número de unidades móveis	1	2	3	4	5	6	7	8
Tempo médio entre um acionamento e outro	01:12:48	01:08:57	01:07:04	01:20:15	01:11:15	01:13:57	01:13:57	01:11:28
Total de mensagens de acionamentos	19	21	22	18	19	19	19	20
Número de unidades móveis solicitadas	41	45	50	37	43	35	38	41
Número de unidades móveis acionadas	9	18	22	30	37	35	37	41
Percentual de sucesso no atendimento a eventos	22%	40%	44%	81%	86%	100%	97%	100%
Média do tempo de resposta dos pedidos ao atendente	00:00:08	00:00:09	00:00:09	00:00:09	00:00:09	00:00:09	00:00:09	00:00:09
Média do tempo de deslocamento	00:12:56	00:15:52	00:15:40	00:14:58	00:15:13	00:10:55	00:14:21	00:12:16
Média do tempo de atendimento	02:18:59	02:18:46	03:02:06	02:24:23	02:08:27	02:09:02	02:53:47	02:31:29
Total de tentativas realizadas para conclusão dos acionamentos	456	486	404	251	218	3	53	1

Os dados obtidos da simulação com alocação de *brokers* em *cluster*, abordagem reativa, encontram-se na Tabela 6.

Tabela 6 – Abordagem reativa (alocação de *brokers* em *cluster*).

	Sim. 1	Sim. 2	Sim. 3	Sim. 4	Sim. 5	Sim. 6	Sim. 7	Sim. 8
Número de unidades móveis	1	2	3	4	5	6	7	8
Tempo médio entre um acionamento e outro	01:05:08	01:08:57	01:07:04	01:05:05	01:06:41	01:12:39	01:12:43	01:13:15
Total de mensagens de acionamentos	21	18	19	22	21	19	19	20
Número de unidades móveis solicitadas	40	34	40	46	41	40	41	48
Número de unidades móveis acionadas	9	16	24	38	36	37	40	48
Percentual de sucesso no atendimento a eventos	23%	47%	60%	83%	88%	93%	98%	100%
Média do tempo de tomada de decisão pelos brokers	00:00:05	00:00:05	00:00:04	00:00:05	00:00:05	00:00:05	00:00:05	00:00:05
Média do tempo de resposta dos pedidos ao atendente	00:00:09	00:00:09	00:00:08	00:00:09	00:00:09	00:00:09	00:00:09	00:00:09
Média do tempo de deslocamento	00:14:00	00:17:40	00:14:57	00:16:16	00:15:29	00:14:25	00:14:40	00:12:50
Média do tempo de atendimento	01:39:57	02:38:19	02:39:04	02:04:59	02:42:59	03:00:41	02:27:31	02:40:32
Total de tentativas realizadas para conclusão dos acionamentos	516	401	387	371	223	140	74	64

Observando os dados das Tabela 5 e Tabela 6, pode-se perceber em relação a abordagem descentralizada com alocação distribuída de *brokers* que as médias dos tempos de tomada de decisão pelos *brokers* (não mostrado na Tabela 5) e as médias dos tempos de resposta dos pedidos ao atendente ficou menor. Isso porque os *brokers* estão em um *cluster* e o atraso de comunicação foi considerado em 0 (zero) segundos. Em relação à abordagem centralizada, os valores das médias ficaram muito próximos porque na arquitetura centralizada a árvore lógica de disseminação de eventos possui somente dois níveis (*broker* raiz e unidades móveis). O percentual de acionamentos com sucesso aumenta à medida que o número de unidades móveis também aumenta. Foram realizados testes com 8 e 16 pontos de referência em todas as abordagens com 1, 4 e 8 unidades móveis e as médias dos tempos de tomada de decisão pelos *brokers* e de resposta dos pedidos ao atendente apresentaram o mesmo comportamento de 4 pontos de referência.

Analisando qualitativa e quantitativamente todas as implementações das arquiteturas e abordagens do middleware E2BS, pode-se avaliar as vantagens e desvantagens apontadas em Hernandes, Calsavara e Lima Jr. (2017a). Constatou-se que a implementação da arquitetura centralizada apresentou baixa complexidade, a arquitetura descentralizada com alocação distribuída de *brokers*, alta complexidade e a arquitetura descentralizada com alocação de *brokers* em *cluster*, média complexidade. Nas abordagens reativas o número de mensagens foi consideravelmente menor do que

nas abordagens pró-ativas. A decisão de escolha das unidades móveis foi mais rápida (levando em conta a média do tempo de tomada de decisão pelos *brokers*) nas abordagens pró-ativas do que nas abordagens reativas. Essa diferença entre uma abordagem e outra foi de segundos. Mas, em uma situação de emergência, como um afogamento, por exemplo, uma vida pode ser salva. Além disso, o middleware se mostrou escalável, com exceção da arquitetura centralizada, pelo fato da existência de um único *broker* que se vier a falhar compromete o funcionamento de todo o sistema.

Portanto, as arquiteturas que tendem a ser as mais indicadas, no caso de Cidades Inteligentes, são as arquiteturas descentralizadas com alocação de *brokers* em *cluster* e com alocação distribuída de *brokers* por regiões da cidade. Levando-se em consideração a média de tomada de decisão pelos *brokers*, as abordagens pró-ativas são as mais indicadas, uma vez que o fator tempo é crucial no salvamento de vidas.

5. Considerações Finais

Neste artigo, descreveu-se a implementação e validação do middleware E2BS, o qual foi utilizado para o acionamento de unidades móveis em situações de emergência no contexto de Cidades Inteligentes. As diferentes arquiteturas do middleware e as duas formas de atualização dos níveis de adequação das unidades móveis foram avaliadas com o uso de um simulador. Não foram realizadas comparação com trabalhos existentes na literatura porque os objetivos divergiam e os middlewares estudados não enviavam eventos somente a um subconjunto de assinantes mais adequados, conforme proposto neste trabalho. Em resumo, a abordagem reativa acrescenta um atraso ao processo de escolha das unidades móveis em relação à abordagem pró-ativa. Por outro lado, a abordagem pró-ativa consome mais recursos do sistema para manter as informações atualizadas, incluindo processamento e comunicação. Quanto as arquiteturas, as descentralizadas se mostraram mais adequadas para Cidades Inteligentes e em relação as abordagens, as pró-ativas são mais indicadas. Como desafio, pretende-se melhorar o método de validação e utilizar diferentes métricas de comparação entre as arquiteturas e as abordagens, além de demonstrar que o envio aos assinantes mais adequados é melhor do que uma escolha aleatória. Futuramente, o trabalho pode ser estendido para outros cenários como a coleta de lixo inteligente e o serviço de cuidados médicos, entre outros.

6. Agradecimento

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela bolsa auxílio mensalidade concedida a primeira autora do artigo.

Referências

ADMILSON, R. L. et al. SensorBus : A Middleware Model for Wireless Sensor Networks. n. c, p. 1–9, 2005.

AMENDOLA, S. et al. RFID Technology for IoT-based Personal Healthcare in SmartSpaces. **IEEE Internet of Things Journal**, v. PP, n. 99, p. 1–1, 2014.

BOONMA, P.; SUZUKI, J. TinyDDS: An Interoperable and Configurable Publish/Subscribe Middleware for Wireless Sensor Networks. **Wireless Technologies: Concepts, Methodologies, Tools and Applications**, p. 819–846, 2011.

CARZANIGA, A.; ROSENBLUM, D. S.; WOLF, A. L. Design and Evaluation of a Wide-Area

Event Notification Service. v. 19, n. 3, p. 332–383, 2001.

COSTA, P. et al. The RUNES middleware for networked embedded systems and its application in a disaster management scenario. **Proceedings - Fifth Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2007**, p. 69–78, 2006.

GYRARD, A.; SERRANO, M.; PLATFORMS, B. S. C. A Unified Semantic Engine for Internet of Things and Smart Cities : From Sensor Data to End-Users Applications. 2015.

HERNANDES, S. C. L.; CALSAVARA, A.; LIMA JR., L. P. Acionamento Inteligente de Unidades Móveis em Cidades. **I Workshop de Computação Urbana (COURB 2017)**. Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 35 (SBRC 2017). Belém do Pará – PA, p. 58-71, 2017a.

HERNANDES, S. C. L.; CALSAVARA, A.; LIMA JR., L. P. Serviços de Emergência em Cidades Inteligentes: o Problema de Acionamento de Unidades Móveis. **Revista Eletrônica de Sistemas de Informação**, vol. 16, n.2, mai-ago 2017b.

IETF. **RFC 7452 - Architectural Considerations in Smart Object Networking**. [s.l: s.n.].

JALOUDI, S. Open source software of smart city protocols current status and challenges. **Open Source Software Computing (OSSCOM), 2015 International Conference on**, p. 1–6, 2015.

KORTUEM, G. et al. Smart objects as building blocks for the Internet of things. **Internet Computing, IEEE**, v. 14, n. 1, p. 44–51, 2010.

LAI, S.; CAO, J.; ZHENG, Y. PSWare: A publish / subscribe middleware supporting composite event in wireless sensor network. **7th Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2009**, 2009.

MEIER, R.; CAHILL, V. STEAM: Event-based middleware for wireless ad hoc networks. **Proceedings - International Conference on Distributed Computing Systems**, v. 2002–Janua, p. 639–644, 2002.

MUSOLESI, M.; MASCOLO, C.; HAILES, S. EMMA: Epidemic Messaging Middleware for Ad hoc networks. **Personal and Ubiquitous Computing**, v. 10, n. 1, p. 28–36, 2006.

PIETZUCH, P. R.; BACON, J. M. Hermes: A distributed event-based middleware architecture. **Proceedings - International Conference on Distributed Computing Systems**, v. 2002–Janua, p. 611–618, 2002.

RAZZAQUE, M. A. et al. Middleware for Internet of Things : A Survey. v. 3, n. 1, p. 70–95, 2016.

ROSE, K.; ELDRIDGE, S.; LYMAN, C. The internet of things: an overview. **Internet Society**, n. October, p. 53, 2015.

SILVA, J. R. et al. PRISMA: A publish-subscribe and resource-oriented middleware for wireless sensor networks. **Advanced International Conference on Telecommunications, AICT**, v. 2014–July, n. July, p. 87–97, 2014.

SOUTO, E. et al. Mires: A publish/subscribe middleware for sensor networks. **Personal and Ubiquitous Computing**, v. 10, n. 1, p. 37–44, 2006.

TSCHOFENIG, H. et al. Architectural Considerations in Smart Object Networking. **Rfc 7452**, v. 1, p. 1–24, 2015.

VERMESAN, O. et al. Internet of Things Strategic Research Roadmap. **Internet of Things Strategic Research Roadmap**, p. 9–52, 2009.