

ManIoT: Uma Plataforma para Gerenciamento de Dispositivos da Internet das Coisas

**Josué B. Antunes^{1,2}, István L. Dénes¹, Marconi Santos¹,
Tiago O. Castro¹, Daniel F. Macedo¹, Aldri L. dos Santos³**

¹UFMG, Departamento de Ciência da Computação, Belo Horizonte, Brasil

²IFNMG, Núcleo de Informática, Araçuaí, Brasil

³UFPR, Departamento de Informática, Curitiba, Brasil

{marchs, tiago.oliveira, damacedo}@dcc.ufmg.br

{josue-jba, istvan}@ufmg.br, aldri@inf.ufpr.br

Abstract. *The Internet of Things (IoT) is a paradigm where devices have Internet connections and can interact with each other. One of the challenges of IoT is related to the heterogeneity of the devices comprising the network, where each object can have different processing capabilities or employ different communication standards. This requires a dynamic and context-aware configuration management system, that meets the requirements of networks and their different features. This paper presents a platform for managing devices on the Internet of Things, called ManIoT, which operates in local and global levels, and considers the perception of context. A prototype was implemented, and the evaluation shows that ManIoT provides the infrastructure to create applications and services, dealing with the integration of heterogeneous devices and applications.*

Resumo. *A Internet das Coisas (IoT) é um paradigma onde os dispositivos possuem conexões com a Internet e podem interagir uns com os outros. Um dos desafios da IoT está relacionado à heterogeneidade dos dispositivos que fazem parte da rede, onde cada objeto pode ter capacidade de processamento ou padrão de comunicação diferente. Isso demanda um sistema de gerenciamento de configuração dinâmica e ciente de contexto, que atenda aos requisitos das redes e suas diferentes funcionalidades. Este trabalho apresenta uma plataforma para o gerenciamento dos dispositivos em Internet das Coisas, chamada ManIoT, que atua em âmbito local e global, e considera a percepção do contexto. Um protótipo da plataforma foi implementado, e experimentos mostraram que ManIoT fornece a estrutura necessária para criação de aplicações e serviços, tratando a integração entre dispositivos, aplicações e dispositivos heterogêneos.*

1. Introdução

A Internet das Coisas (IoT - *Internet of Things*) é um paradigma onde uma variedade de dispositivos, como Tags e leitoras RFID (*Radio-Frequency IDentification*), nós sensores, telefones celulares, e objetos cotidianos (como lâmpadas, geladeiras e outros), são habilitados para interagir entre si para atingir objetivos comuns [Atzori et al. 2010]. A IoT traz, entre outros benefícios, a coleta de dados dos dispositivos e das suas condições operacionais em tempo real. Com esses dados, é possível automatizar tarefas domésticas e

melhorar a tomada de decisão nas empresas a partir de uma melhor visão dos processos que ocorrem no ambiente onde as redes IoT estão inseridas.

Os serviços oferecidos pela IoT podem beneficiar diversas áreas. Entre as mais promissoras estão os serviços de apoio à saúde, infraestrutura e serviços do setor público. O monitoramento remoto em saúde, por exemplo, proporciona uma grande diferença na vida de pessoas com doenças crônicas e, simultaneamente, diminui os custos de cuidados com a saúde desses pacientes. A automação residencial usando IoT permite a dispositivos como termostatos inteligentes adequar a temperatura do ambiente, e que lâmpadas sejam controladas à distância para tarefas de segurança e economia de energia. No entanto, a IoT apresenta muitos desafios no seu gerenciamento até que se propicie serviços inteligentes e integrados a qualquer momento e em qualquer lugar.

Um dos desafios mais importantes no gerenciamento da IoT consiste na heterogeneidade dos dispositivos que fazem parte da rede. Além dos diferentes protocolos de comunicação, capacidade de memória e processamento, os dispositivos podem empregar diferentes formatos e tipos de dados [Atzori et al. 2010]. Finalmente, fatores como a imprecisão dos dados produzidos (sistemas RFID podem gerar entre 60 e 70% de dados incorretos), o grande conjunto de dados produzidos em tempo real e a semântica implícita impõem desafios na configuração dos ambientes da IoT [Meng et al. 2013].

O gerenciamento de IoT é mais complexo que o gerenciamento de redes de sensores sem fio (RSSF) ou de redes IP. As redes IP, apesar de tratarem dispositivos com *hardware* e software heterogêneos, empregam mecanismos de comunicação homogêneos devido ao uso do protocolo IP. As RSSF devem gerenciar as falhas frequentes de comunicação e a baixa segurança dos enlaces sem fio (por exemplo, a arquitetura MANNA [Ruiz et al. 2003]), e este gerenciamento deve ser ciente do contexto. Entretanto, os dispositivos de uma RSSF em geral tendem a ser mais homogêneos em configuração que na IoT. Nas redes IoT, além dos desafios acima, é necessário suportar aplicações e serviços em rede que envolvem: (i) o uso de dispositivos de características diferentes; (ii) a interação entre redes IoT, necessitando gerenciamento local (por exemplo o dono da casa) e global (a concessionária de energia, que procura reduzir a demanda em horas de pico), ambos cientes do contexto. As arquiteturas de gerenciamento em IoT existentes, entretanto, atendem parcialmente a estes requisitos [Pires et al. 2015].

Este trabalho propõe uma plataforma para o gerenciamento de dispositivos em Internet das Coisas, chamada ManIoT (*Management for Internet of Things*). Esta plataforma integra e gerencia as funcionalidades individuais dos dispositivos em uma rede IoT e permite a criação de novos serviços cientes do contexto. A ManIoT define uma estrutura de gerenciamento em dois escopos: gerenciamento local, onde a aplicação é executada no mesmo ambiente que os dispositivos, e gerenciamento global, onde as aplicações controlam os dispositivos remotamente. A estrutura da plataforma é expansível, permitindo a adição de novos tipos de dispositivos. Além disso, a plataforma prevê serviços genéricos, tais como descoberta de nós, armazenamento de dados e autenticação, que são blocos básicos para a construção de aplicações IoT.

O artigo está organizado como segue. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 descreve a plataforma ManIoT, e a Seção 4 discute o protótipo e os resultados dos testes. Por fim, a Seção 5 conclui o artigo e relaciona os trabalhos futuros.

2. Trabalhos Relacionados

Na literatura há diversas plataformas de gerenciamento para IoT. Essas plataformas são comumente denominadas *middleware*, pois são camadas de *software* que ocultam dos desenvolvedores as complexidades e heterogeneidades referentes ao *hardware*, aos protocolos de rede, às plataformas e dependências do sistema operacional. Elas também facilitam o gerenciamento de recursos e aumentam a previsibilidade da execução de aplicações. No entanto, tais plataformas em geral tratam parcialmente de aspectos como adaptação dinâmica, ciência de contexto e a interação entre redes IoT [Pires et al. 2015].

A interoperabilidade entre os dispositivos é um dos principais requisitos das plataformas de gerenciamento. As plataformas RestThing [Qin et al. 2011], EcoDiF [Delicato et al. 2013], SmartThings [SmartThings 2015], Xively [LogMeIn 2015], Carriots [Carriots 2015] e IFTTT [IFTTT 2016] abstraem a heterogeneidade dos usuários e aplicações. A plataforma RestThing visa permitir que desenvolvedores criem aplicações usando REST (*REpresentational State Transfer*), combinando recursos físicos e *Web*, de modo que dispositivos e informações *Web* são manipulados por uma *interface REST*. A plataforma EcoDiF (Ecossistema *Web* de Dispositivos Físicos) integra dispositivos físicos heterogêneos e os conecta à Internet, fornecendo funcionalidades de controle, visualização, processamento e armazenamento de dados em tempo real. A plataforma *open source* SmartThings permite que os usuários criem aplicativos e os conectem aos dispositivos, às ações e aos serviços oferecidos pela plataforma. SmartThings permite ainda a integração de novos dispositivos e fornece suporte para os aplicativos (*SmartApps*) na comunicação com serviços *Web* externos através do envio de notificações *Push*, SMS e da apresentação do seu terminal REST. Xively e Carriots utilizam serviços de nuvem para gerenciar os dados providos pelos dispositivos. A plataforma IFTTT (*If This, Then That*) trabalha com sistemas *Web*, Android, iOS e dispositivos físicos. Ele combina duas contas diferentes de sites ou redes sociais para interagirem e funcionarem juntas.

Além da integração, o desempenho, a tolerância a falhas e o uso de protocolos customizados foram características destacadas pelos autores em [Guiping 2013] e [Bin et al. 2011]. Em [Guiping 2013], o autor propõe a criação de um protocolo para gerenciamento de dispositivos chamado TMP - *Things Management Protocol*. O TMP interconecta dispositivos entre si e com as aplicações. Em [Bin et al. 2011], os autores propõem um sistema de gerenciamento de dispositivos de edifícios chamado BEIOT - *Building Equipment Internet of Things*. Esse sistema permite acessar e enviar comandos para dispositivos, como ar condicionado e lâmpadas. Além disso, BEIOT permite a localização de pessoas e a otimização do consumo de energia, que é feita tomando como base a quantidade de pessoas presentes em cada ambiente.

O uso de protocolos conhecidos visa facilitar a integração da plataforma de IoT com outros sistemas. Em [Ning et al. 2007], os autores propuseram um sistema de gerenciamento de leitores RFID. Os autores utilizaram o protocolo SNMP para controlar o servidor que lê as etiquetas. Eles também criaram o protocolo RFID-MP (*Radio Frequency IDentification - Managing Protocol*), que gerencia a comunicação com o leitor RFID. Em [Sehgal et al. 2012], foi investigado o uso de tecnologias de gerenciamento IP (SNMP e NETCONF) em dispositivos com recursos limitados. Os autores observaram que SNMP faz uso mais eficiente dos recursos, respondendo um pedido de processamento até dez vezes mais rápido que NETCONF.

Diferentemente das plataformas mencionadas, a plataforma proposta neste artigo, chamada ManIoT, procura tratar todos os pontos levantados ao suportar tanto o gerenciamento baseado no contexto quanto possibilitar um controle no escopo de gerenciamento Local e Global.

3. ManIoT - Gerenciamento de Dispositivos em Internet das Coisas

Esta seção descreve a plataforma ManIoT (*Management for Internet of Things*) para o gerenciamento dos dispositivos que compõem ambientes da IoT. Um ambiente corresponde a um domínio de aplicações, bem como os sensores instalados fisicamente neste ambiente. As aplicações implementam um cenário beneficiado por uma infraestrutura de IoT, como o controle de segurança de uma residência, o gerenciamento de iluminação de uma sala, a assistência a pessoas com dificuldade de locomoção, entre outros.

A plataforma ManIoT estabelece dois escopos de gerenciamento, Local e Global/Remoto. O gerente local atua dentro de um cenário, gerenciando os dispositivos que compõem este cenário a partir de informações sobre o contexto. Desta forma, por exemplo, o gerente local pode controlar os eventos que uma aplicação ou usuário pode realizar, como ligar ou desligar uma lâmpada. Já o gerente global/remoto procura uniformizar as ações realizadas em diferentes cenários a partir de diretrizes de alto nível. Assim, uma concessionária de energia através do gerente global, por exemplo, poderia definir cotas máximas de consumo por área ou residência em períodos de potenciais *blackouts*. A plataforma especifica um modelo de dados e de informação com o objetivo de padronizar o formato dos dados utilizados na comunicação entre aplicações, serviços e dispositivos. O estado dos dispositivos (ligado/desligado) e o id (identificação do dispositivo) são exemplos de características utilizadas no modelo de informação. Ainda, visando a extensibilidade e integração com outros sistemas, a plataforma faz uso de protocolos e padrões populares da indústria para modelos de dados, como o XML (*eXtensible Markup Language*) e o REST.

A plataforma ManIoT também leva em conta a heterogeneidade dos dispositivos ou Coisas. Assim, ManIoT não requer grandes modificações ou instalações de *softwares* adicionais nos dispositivos da rede ou nos dispositivos dos usuários, à exceção aos dispositivos *Smartphone*, devido o uso de sistema operacional de propósito geral. O acesso às aplicações da plataforma ManIoT é realizado através de uma *interface Web*. O acesso é restrinido por contas de usuários, e um administrador define as aplicações e recursos dos dispositivos que esses usuários podem acessar, conforme será detalhado na Seção 3.4.

3.1. Gerenciamento Local e Global

As aplicações IoT podem exigir dois ou mais escopos de gerenciamento. Esses diferentes escopos são importantes em IoT porque políticas de alto nível podem orientar a gestão de uma sub-rede. Por exemplo, na gestão da energia elétrica, um gerente global rodando na concessionária de energia poderia definir horas de baixa e de alta demanda, e solicitar aos consumidores evitar o uso de alguns dispositivos de alto consumo em horários de pico. Os gerentes locais iriam gerenciar o uso de energia em cada casa, baseados nas regras da concessionária e nas preferências de cada usuário.

Assim, ManIoT emprega uma arquitetura de gerenciamento em dois níveis. O gerente global define diretrizes para o gerenciamento de alto nível, enquanto o gerente

local efetiva os comandos recebidos do gerente global e administra as rotinas locais. Cada gerente local roda em um servidor de rede ou dispositivo embarcado de maior poder computacional e sempre ligado à rede elétrica (por exemplo um roteador doméstico, uma televisão) dentro de um cenário. A comunicação entre o gerente local e o gerente global se dá via uma conexão TCP/IP.

A separação em dois escopos de gerenciamento traz alguns benefícios: (1) pode-se implementar políticas de gerenciamento e tomar decisões locais. Cada gerente local trabalha com as características de um cenário específico sem preocupar com detalhes de outros cenários; (2) O gerente global toma as decisões mais importantes usando informações providas por cada gerente local e não precisa tratar detalhes específicos de cada cenário. A próxima subseção detalha os componentes do gerente local.

3.2. Componentes de Software do Gerente Local

Os *softwares* que compõem a plataforma ManIoT no gerente local, Figura 1, possuem as camadas de aplicação, de serviços, de adaptação, de comunicação, de Coisa/dispositivos bem definidas e são detalhadas abaixo.

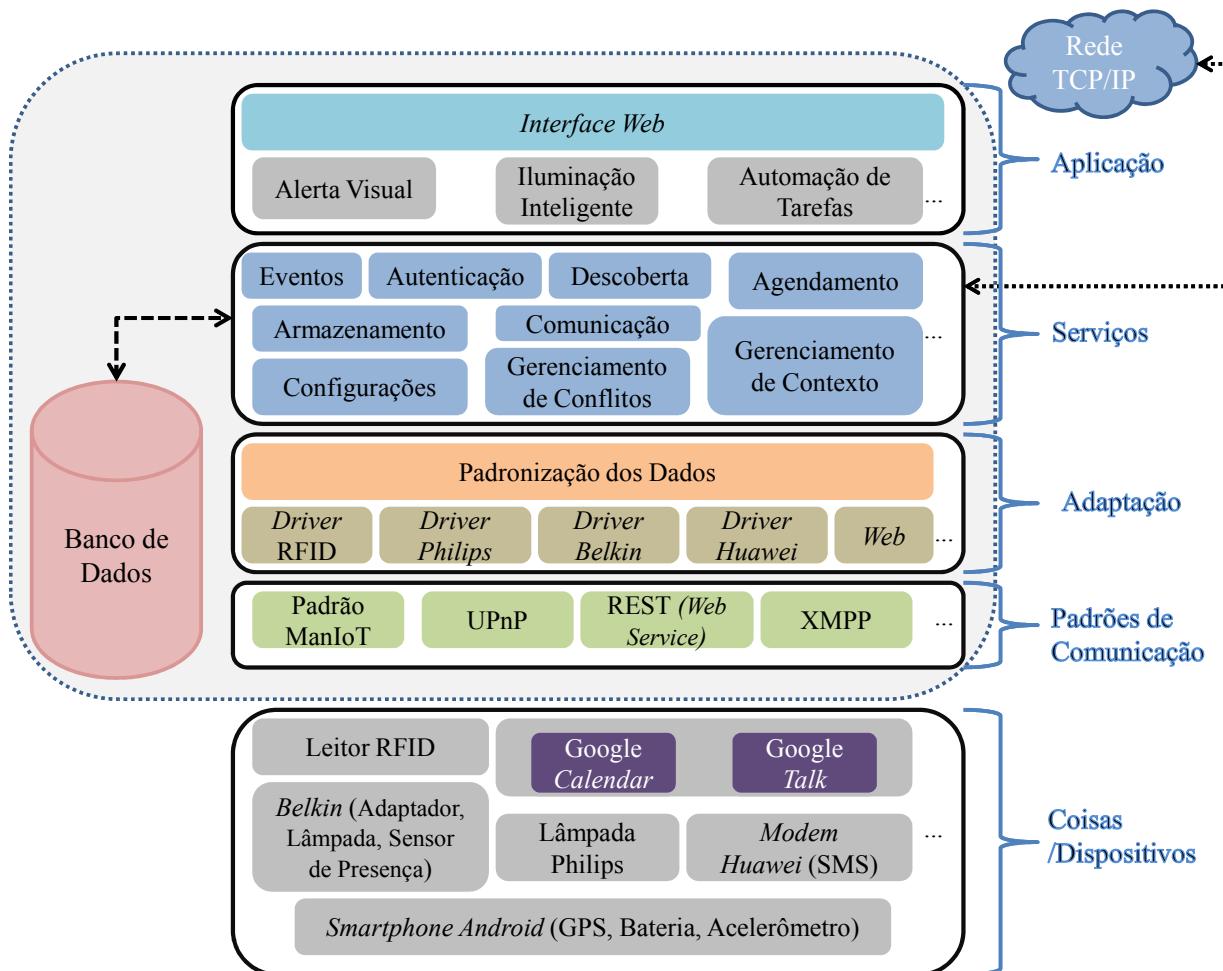


Figura 1. plataforma ManIoT: gerente local.

Camada de aplicação. A primeira camada é composta pelas aplicações. Cada aplicação usa os dados providos por um ou vários dispositivos, bem como os serviços da plataforma. Os usuários da rede acessam as aplicações através de uma *interface web* e essas aplicações, por sua vez, interagem com ManIoT utilizando chamadas de função. Cada aplicação solicita à plataforma a execução de ações sobre os sensores tendo em vista o cenário implementado. Por exemplo, uma aplicação de gerenciamento de energia solicita desligar ou ligar um ar condicionado tendo em vista a redução do consumo.

Camada de serviços. A segunda camada é formada pelos serviços. Os serviços dão suporte às aplicações e utilizam as abstrações implementadas pelos *drivers* para realizar a comunicação com os dispositivos. Entre os itens dessa camada tem-se:

- **Armazenamento.** Mantém um histórico dos dados coletados, eventos capturados pelos dispositivos e informações sobre as características e funcionalidades de cada dispositivo e de cada cenário;
- **Agendamento.** Programa ações futuras ou que se repetem periodicamente, tais como consultas a sensores ou alterações no estado de um dispositivo;
- **Autenticação.** Realiza a identificação dos usuários (utilizando *login* e senha) e dos dispositivos (utilizando um identificador único);
- **Descoberta.** Realiza a descoberta de novos dispositivos (com *drivers* previamente cadastrados), identificando os serviços oferecidos por esse dispositivo;
- **Configurações.** Gerencia as configurações básicas de cada dispositivo e fornece uma interface para as aplicações acessarem essas configurações;
- **Comunicação.** controla a comunicação entre os gerentes locais e o gerente global;
- **Eventos.** Permite que as aplicações sejam notificadas de eventos disparados pelas Coisas, como detecção de movimento pelo sensor de presença, condições da temperatura ambiente, localização de um dispositivo, etc;
- **Gerenciamento de Conflitos.** Diante de operações conflitantes entre duas aplicações, como “desligar” e “ligar” uma lâmpada, determina qual usuário ou aplicação possui prioridade na execução;
- **Gerenciamento de Contexto.** Realiza a percepção de informações contextuais, como localização e tempo (data e hora) para prover funções relevantes aos usuários e outros serviços, como agendamento e gestão de conflitos.

Camada de adaptação. A terceira camada é dividida em duas partes, sendo a primeira responsável pela padronização dos dados e a segunda por tratar as especificidades de cada dispositivo. Cada tipo de dispositivo possui um *driver* específico que abstrai as especificidades do acesso aos seus sensores e atuadores, o que permite o gerenciamento por parte dos serviços de forma integrada.

Camada de comunicação. A quarta camada é composta pelos diferentes protocolos de acesso aos dispositivos. Como mencionado anteriormente, a rede será composta por dispositivos que podem utilizar protocolos de aplicação (por exemplo, UPnP ou um protocolo proprietário) e de redes (ZigBee, WiFi) diferentes. Por exemplo, o padrão REST é utilizado pelo “dispositivo Virtual” Google *Calendar*, já dispositivos WeMo, fabricados pela Belkin, utilizam UPnP (*Universal Plug and Play*).

Camada de Coisas/dispositivos. A última camada é formada pelas Coisas. Existem dois tipos diferentes de dispositivos, a saber, os dispositivos reais e os dispositivos virtuais. Os dispositivos reais são sensores e atuadores físicos, por exemplo, uma lâmpada

inteligente (atuador), um sensor de pressão (sensor). Já os dispositivos virtuais capturam informações de um servidor conectado a uma rede TCP/IP. Por exemplo, um serviço de calendário ou de *e-mail*, ou um servidor de redes sociais.

3.3. Componentes de Software do Gerente Global

O gerente global procura uniformizar as ações realizadas em diferentes cenários a partir de diretivas de alto nível. Assim, o gerente global possui duas camadas - Aplicação e Serviços, como mostra a Figura 2.

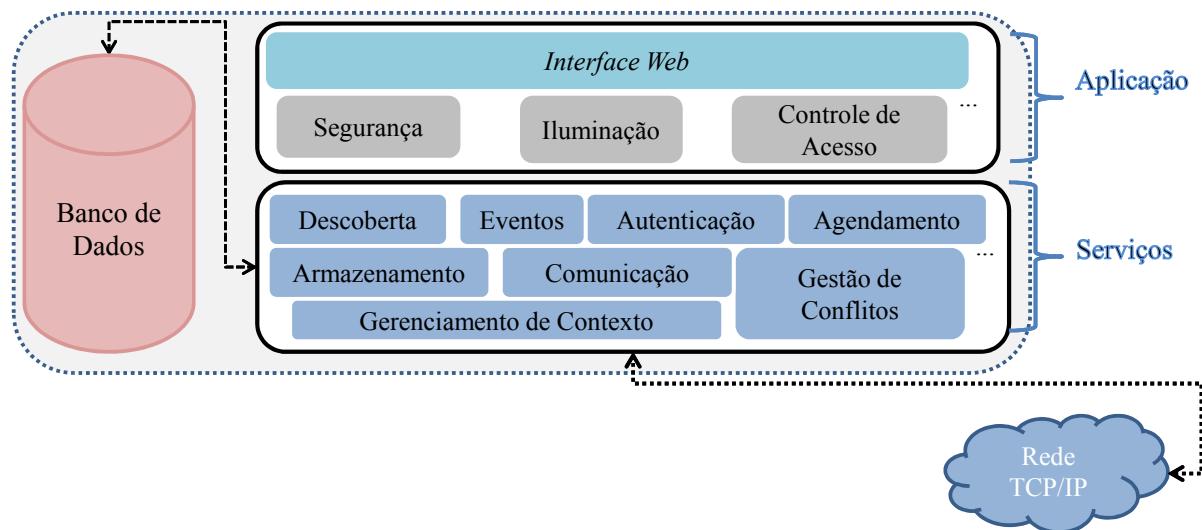


Figura 2. plataforma ManIoT: gerente global/remoto.

Os serviços implementados no escopo global possuem as mesmas funções daqueles implementados no escopo local, segunda camada da Figura 1 e Figura 2 . Os serviços do escopo global tratam conjuntos de dados maiores e fornecem suporte para aplicações mais abrangentes. Por exemplo, no contexto de gerenciamento elétrico, o gerente global deve ter a capacidade de gerenciar possíveis faltas de energia em diversas residências de um bairro. As ações definidas pelos serviços globais são enviadas e executadas nos dispositivos dos respectivos gerentes locais, para tanto empregando uma conexão TCP/IP. Por exemplo, na iminência de um “apagão”, a companhia que fornece energia elétrica (através do gerente global) pode enviar uma solicitação para o gerente local desligar temporariamente aqueles dispositivos que consomem mais energia.

3.4. Gerenciamento de Conflitos

Um aspecto importante em uma rede com múltiplos serviços se refere ao gerenciamento de conflitos. Estes podem ocorrer quando duas ou mais aplicações modificam a configuração dos dispositivos de forma conflitante, como por exemplo, quando uma aplicação solicita apagar uma lâmpada, e uma segunda solicita que esta fique ligada. Este gerenciamento é implementado pelo serviço Gerenciamento de Conflitos e Gerenciamento de Contexto.

Para realizar o gerenciamento de conflitos, cada dispositivo possui um usuário administrador. Os administradores possuem controle total e irrestrito sobre esse dispositivo,

podendo delegar permissões a outros usuários de forma permanente, ou temporária, baseado no contexto (a localização do usuário). A localização poderia ser fornecida por dispositivos como leitores RFID, *smartphones* ou sensores de presença. Como exemplo, uma lâmpada inteligente na porta de uma casa. Alice, um usuário comum, não poderia desligar a lâmpada do quarto de Maria, outra usuária comum, se Maria estiver no seu quarto. Joana, a mãe e administradora da lâmpada, poderia desligar ou ligar a lâmpada a qualquer momento, sobrepondo as ações de Maria e Alice.

Algorithm 1 Gerenciamento de operações conflitantes no ManIoT.

```

1: procedure Conflito(Recurso, Operacao, idUsuarios[])
2:   idUsuario ← 0
3:   if Operacao = IC then idUsuario ← BuscarMaisProximo(idUsuarios)
4:     return idUsuario
5:   else
6:     for i ← 1, n do
7:       if idUsuarios[i].Tipo = Administrador then
8:         idUsuario ← idUsuarios[i]
9:       return idUsuario
10:      if idUsuario.Tipo ≠ Administrador then
11:        idUsuario ← BuscarPermissao(idUsuarios, Recurso)
12:      return idUsuario

```

O Algoritmo 1 descreve a operação do serviço de gerenciamento de conflitos. O serviço recebe a lista de usuários que enviaram comandos ao dispositivo, verifica se o dispositivo (que contém o recurso) está habilitado para ser gerenciado no Modo Contexto (IC) e retorna o usuário mais apto (l.3). Se o modo IC não estiver habilitado, a lista de usuários candidatos deverá ser verificada e escolhido aquele que possui permissão de administrador sobre o recurso (l.7). Por fim, se não existir um administrador entre os candidatos, cada usuário candidato poderá solicitar essa permissão diretamente ao administrador do recurso (l.11).

4. Desenvolvimento e Avaliação da Plataforma ManIoT

A plataforma ManIoT foi implementada em um protótipo referente ao subsistema do gerente local e de seus componentes. O desempenho da plataforma é avaliado ao empregá-lo no gerenciamento de dois cenários comuns de IoT: um cenário de “iluminação residencial” e um para “automação de tarefas”. Foi implementado o gerente local pois ele é o módulo mais importante da plataforma, ao integrar os dispositivos de forma lógica e física. Além disso, os serviços previstos no gerente global possuem o mesmo princípio funcional dos serviços locais.

Os componentes implementados no protótipo foram os padrões de comunicação (REST, UPnP, entre outros); os *drivers* da Camada de Adaptação, responsáveis pela comunicação com os seis dispositivos utilizados; o banco de dados local; e os serviços, como Armazenamento e Eventos. A implementação foi feita em Java e o banco de dados escolhido foi o MySQL. Analisando as características dos dispositivos e o fato que muitas ações são assíncronas, o software foi desenvolvido empregando-se uma abordagem baseada em eventos. O protótipo realiza a comunicação entre componentes via banco

de dados. Assim, os sensores geram novas entradas no banco de dados, e as consultas são implementadas como leituras aos dados inseridos. Por exemplo, quando o sensor de luminosidade faz a leitura de um novo valor, a plataforma insere esse valor no banco de dados. A aplicação, por sua vez, pode ler esse valor e usá-lo para definir seus parâmetros de operação.

As tabelas do banco de dados do ManIoT armazenam informações sobre usuários, aplicações, dispositivos, dados, recursos, permissões, entre outras. Como exemplo, as Tabelas 1 e 2 definem os campos para armazenar as informações dos dados coletados pelos recursos dos dispositivos. Na Tabela 1 o campo idRECURSO identifica cada recurso de forma única, os campos idCOISA e idADM identificam os dispositivos e o administrador do recurso, respectivamente. Tem-se ainda o campo “nome”, que armazena o nome dos respectivos recursos.

Tabela 1. Tabela RECURSO - Banco de Dados ManIoT

RECURSO			
idRECURSO	idCOISA	nome	idADM
01	01	GPS	000
02	01	Acelerômetro	001
03	02	Tag1	000

Como pode ser observado na Tabela 2, além da identificação do recurso, o protótipo armazena os valores dos dados capturados pelos dispositivos e o momento da coleta no campo *time*. Com esse formato qualquer tipo de dado (texto, inteiro ou real) pode ser armazenado no banco. Nos exemplos de entradas, encontra-se na 1, as informações sobre os recursos GPS, acelerômetro e Etiqueta, e na tabela 2, os valores coletados pelos recursos (01 e 02) com data e hora de coleta.

Tabela 2. Tabela DADOS - Banco de Dados ManIoT

DADOS		
idRECURSO	valor	time
01	“12.002343, 23.456837”	01/05/2015 22:45:32
02	“8882321123233”	10/03/2015 02:45:32

Dois cenários de testes foram implementados para avaliar as funcionalidades e o desempenho da plataforma. O cenário **Iluminação Inteligente** controla a iluminação nos ambientes de uma residência, ligando/desligando e ajustando a luminosidade das lâmpadas de acordo com a existência de pessoas presentes no recinto e da iluminação natural disponível. Já o cenário **Automação de Tarefas** realiza o acionamento de equipamentos de acordo com a localização do usuário.

Foram empregados os seguintes dispositivos nos cenários avaliados:

- Um PC Linux (i3-4160 CPU @ 3.60GHz e 16GB de RAM) rodando o software do ManIoT;
- Três lâmpadas *Philips Hue* (permitem ajuste de cor e intensidade de luz);
- Um *tablet* Samsung Galaxy Tab 2 7.0, rodando Android 4.0 Ice Cream Sandwich (faz a leitura da localização via GPS e também da luminosidade no ambiente)

- Uma tomada Belkin WeMo (mede consumo de energia e liga/desliga a saída de energia na tomada).

Como o objetivo da avaliação foi identificar o desempenho da plataforma, as métricas medidas foram o consumo de recursos no gerente local, a largura de banda empregada pela plataforma e o tempo de reação a eventos. O uso de recursos computacionais foi avaliado pois permite determinar se a plataforma pode ser executada em dispositivos de baixo custo. Já a largura de banda e o tempo de reação medem o desempenho da plataforma.

4.1. Resultados e Análise

Os resultados foram obtidos com os testes realizados nos cenários Iluminação Inteligente e Automação de Tarefas e empregando métricas adequadas para mostrar a capacidade da ManIoT em prover aos ambientes uma adaptação dinâmica e ciência de contexto.

Cenário Iluminação Inteligente

Neste cenário, a iluminação de um ambiente é ajustada de acordo com a presença de pessoas e com a existência de luz natural. As lâmpadas são ligadas somente quando há uma pessoa no ambiente. Já a intensidade da luz é inversamente proporcional à quantidade de luz natural. Foi empregadas as lâmpadas Philips Hue, bem como um *tablet*. O *tablet* fornece, via GPS, a localização da pessoa, e o seu sensor de luminosidade indica a quantidade de luz natural no ambiente.

Foi medido o tempo de reação entre a detecção da variação de luminosidade e a atuação da plataforma sobre a intensidade de emissão de luz da lâmpada. Como não é possível inserir código nas lâmpadas inteligentes para notificar a modificação da sua luminosidade, este teste foi feito visualmente, onde tampou-se o sensor de luminosidade do *tablet*, e verificou-se que o ajuste da intensidade da lâmpada ocorreu após menos de um segundo, o que considerou-se aceitável para a aplicação. Vale ressaltar que o sensor de luminosidade atualiza seus dados quando ocorre uma diferença de 50 *lúmens* para mais ou para menos, de acordo com o último valor lido. A Figura 3 apresenta uma amostra do uso da CPU no computador rodando o ManIoT. Esse percentual é variável, se mantendo entre 0 e 6%, com uma média de 1.78%. Alguns picos de processamento são observados. Eles são causados por solicitações da plataforma para que a lâmpada modifique a sua intensidade.

A Figura 4 apresenta uma amostra das taxas de troca de dados entre a plataforma ManIoT e os dispositivos a cada segundo. Os valores de *Download* são os dados enviados dos dispositivos para a arquitetura e o *Upload* são dados enviados da arquitetura ManIoT para os dispositivos. Os dados recebidos dos sensores variam entre 0 e 50kbps, com uma média de 25kbps, enquanto os dados enviados variam entre 0 e 32kbps, com uma média de 15kbps. As trocas de dados são mais intensas quando o gerente local notifica as lâmpadas para uma troca de intensidade luminosa. Mesmo considerando o uso do UPNP, que é extremamente verboso pois emprega XML-RPC, o volume de dados trocado é relativamente baixo. Pode-se considerar ainda o fato que as redes Ethernet e WiFi aceitam 100 e 54mbps, respectivamente, o protótipo ManIoT consome aproximadamente 0.05% da banda dessas redes no pior caso.

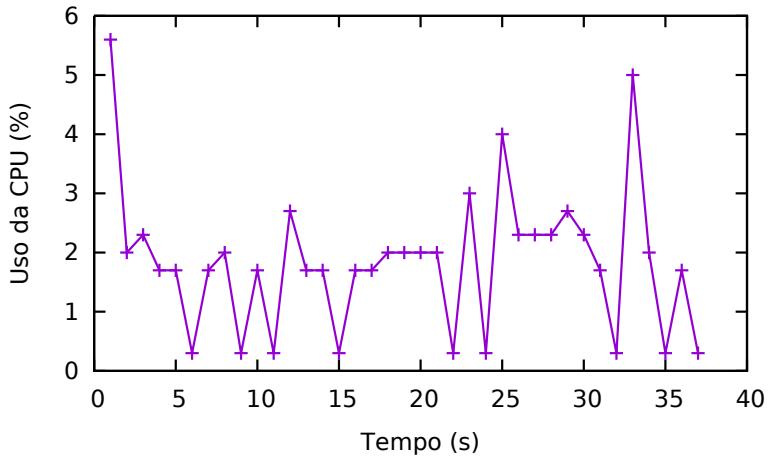


Figura 3. Percentual de consumo de CPU.

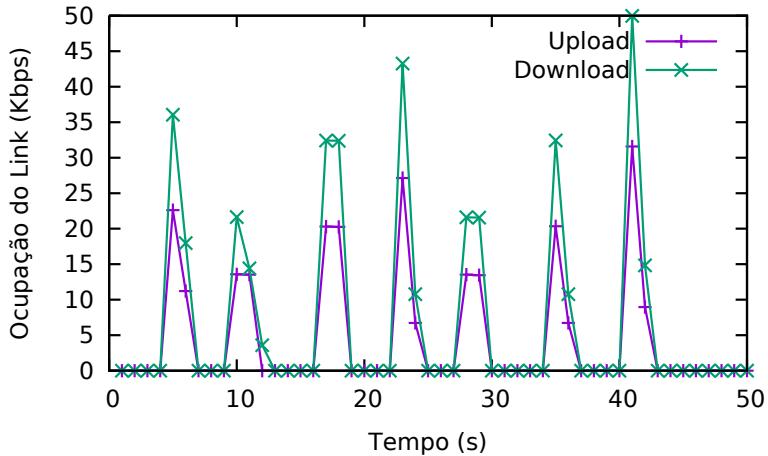


Figura 4. Troca de dados entre ManIoT e os dispositivos.

Já o uso de memória RAM do gerente local variou entre 3 e 7MB, mantendo a média de 3.5MB. Esses valores são justificados devido à pequena quantidade de dados trocados entre o protótipo do gerente local e os dispositivos, reforçando assim o uso mínimo dos recursos de *hardware*.

Cenário Automação de Tarefas

Neste cenário realizou-se a ativação automática de dispositivos elétricos de acordo com a localização do usuário. Foram realizados os testes utilizando a função GPS do *tablet*. Caso o usuário esteja em uma região geográfica bem definida, ativa-se a tomada WeMo. Quando o usuário sai dessa região, a tomada é desligada. Este cenário simula, por exemplo, uma cafeteira que é ativada quando o usuário chega na sua casa.

No teste realizado, classificou-se a localização do usuário em três zonas: “Casa”, “Trabalho” ou “Outro”. Ao detectar a proximidade ou entrada em uma dessas zonas, a plataforma dispara determinada tarefa. As zonas são definidas como um ponto central

(a casa ou local de trabalho do usuário) e um raio. Com isso o protótipo dispara as ações no momento que o usuário cruza essa fronteira virtual. Foi simulada a saída ou chegada de uma pessoa na sua residência. No momento de saída do usuário, o sensor de tomada *Switch WeMo Belkin* era ativado (ligando o dispositivo conectado no mesmo), e na chegada, o sensor era desativado. Para tanto, foi utilizado um *software* chamado *Fake GPS*¹ para definir a localização do usuário.

Mediu-se o tempo de reação, que é o tempo decorrido entre a ultrapassagem da fronteira virtual e o acionamento do dispositivo conectado ao sensor de tomada. A Tabela 3 apresenta dados dos testes onde houve variação no tempo de resposta com média de 0.158 segundos e desvio padrão de 0.0459. Este tempo de reação considera o tempo entre o envio da notificação pelo *tablet* até a ativação da tomada WeMo. A medição foi realizada através da diferença entre os tempos dos *logs* de envio e recebimento de uma notificação.

Tabela 3. Reação da plataforma - Automação de Tarefas

Número de Sequência	Tempo de Reação (Segundos)
1	0.141
2	0.179
3	0.149
4	0.125
5	0.219
6	0.125
7	0.227
8	0.101
Média	0.158

Já o consumo de memória RAM se manteve entre 3 e 3.75MB, com média de 3.4MB. Portanto, assim como no cenário iluminação inteligente, o protótipo faz uso moderado do recurso memória. Observou-se ainda que os valores médios também foram próximos daqueles encontrados no primeiro cenário, mas com uso máximo menor (igual a 3.75MB). No uso de CPU, constata-se que o protótipo fez uso moderado. A Figura 5 apresenta uma amostra de execução onde o valor máximo aproximou-se de 6%, mas a média (considerando somente o efetivo processamento) ficou em 3%. Por fim, vale ressaltar que os valores de uso de CPU também foram próximos aos valores percebidos no cenário iluminação inteligente. Considerando a quantidade de memória e de CPU empregada pelo protótipo, a que acredita-se que a plataforma ManIoT poderia ser executada em dispositivos menos potentes, como televisores ou *gateways IoT*.

Mensurou-se o consumo de banda para transmissão de dados entre o gerente local e os dispositivos. A Figura 6 apresenta uma amostra dos testes onde percebe-se os valores de 4.5kbps para envio e 1.5 a 2kbps para recebimento de dados. Essa troca de dados ocorre nos momentos em que o *tablet* cruza a fronteira virtual. Neste cenário os valores apresentados foram ainda menores que no cenário anterior, chegando a taxas de 10% dos valores percebidos no cenário iluminação inteligente. Essa diminuição ocorre devido o fato do protótipo, neste cenário, utilizar uma menor quantidade de eventos.

¹Fake GPS, disponível em <https://play.google.com>

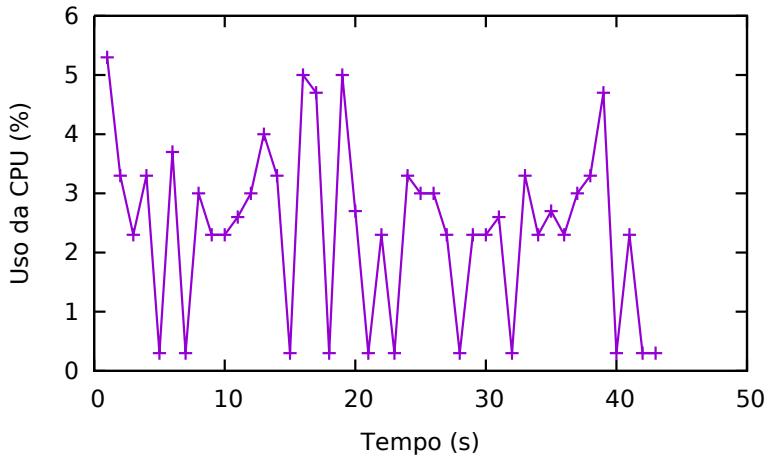


Figura 5. Percentual de consumo de CPU.

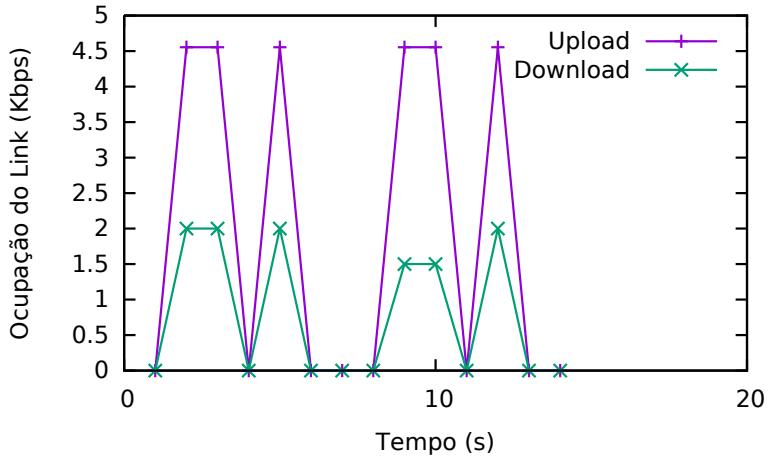


Figura 6. Troca de dados entre ManIoT e os dispositivos.

5. Conclusões e Trabalhos Futuros

Este trabalho apresentou uma plataforma para o gerenciamento dos dispositivos em Internet das Coisas, chamada ManIoT, que promove a integração de dispositivos IoT heterogêneos. ManIoT é genérica e pode ser utilizada em vários cenários, visto que ela emprega uma estrutura em dois níveis de gerenciamento, que permite a inclusão de novos dispositivos e novos serviços de gerenciamento em IoTs. Além disso, ManIoT faz uso dos dados de contexto e cria possibilidades para expansão dos serviços oferecidos pela rede e para especificação de novos serviços.

Um protótipo da plataforma foi implementado, e experimentos foram realizados considerando um ambiente residencial. Os resultados mostraram que ManIoT oferece bons tempos de resposta e requer baixo poder computacional, indicando que o software pode ser executado em dispositivos domésticos de baixo poder computacional. Como trabalhos futuros, estão a implementação do nível de gerência global, o uso da ManIoT em cenários distintos em paralelo, e a avaliação dos mecanismos de autenticação do ManIoT.

Agradecimentos

Os autores gostariam de agradecer a Fapemig, CNPq, CAPES e PBQS-IFNMG pelo suporte ao desenvolvimento deste trabalho.

Referências

- Atzori, L., Lera, A., and Morabito, G. (2010). The Internet of things: A survey. *Computer Network*, pages 2787–2805.
- Bin, S., Guiqing, Z., Shaolin, W., and Dong, W. (2011). The development of management system for building equipment Internet of things. In *3rd International Conference on Communication Software and Networks (ICCSN), 2011*, pages 423–427.
- Carriots, C. S. (2015). Carriots: Carrying the internet of things. <https://www.carriots.com/>. Acessado em: 20-10-2015.
- Delicato, F. C., Pires, P. F., and Batista, T. (2013). *Middleware solutions for the Internet of Things*. Springer.
- Guiping, D. (2013). Design and implementation on a things management protocol for Internet of things. In *Proceedings of the 32nd Chinese Control Conference (CCC)*, pages 7361–7364.
- IFTTT (2016). Connect the apps you love. <https://ifttt.com/>. Acessado em: 03-01-2016.
- LogMeIn (2015). Xively connected product management. <https://xively.com/>. Acessado em: 21-10-2015.
- Meng, M., Wang, P., and Chao-Hsien, C. (2013). Data management for internet of things: Challenges, approaches and opportunities. In *IEEE International Conference on and IEEE Cyber, Physical and Social Computing (CPSCom)*, pages 1144–1151.
- Ning, H., Ning, N., Qu, S., Zhang, Y., and Yang, H. (2007). Layered structure and management in Internet of things. In *Future Generation Communication and Networking (FGCN 2007)*, volume 2, pages 386–389.
- Pires, P., Delicato, F., Batista, T. V., Avila, T., Cavalcante, E., and Pitanga, M. (2015). *Capítulo 3: Plataformas para a Internet das Coisas*. SBRC 2015.
- Qin, W., Li, Q., Sun, L., Zhu, H., and Liu, Y. (2011). Restthing: A restful web service infrastructure for mash-up physical and web resources. In *Embedded and Ubiquitous Computing (EUC), 2011 IFIP 9th International Conference on*, pages 197–204.
- Ruiz, L., Nogueira, J., and Loureiro, A. (2003). Manna: a management architecture for wireless sensor networks. *Communications Magazine, IEEE*, 41(2):116–125.
- Sehgal, A., Perelman, V., Kuryla, S., and Schonwalder, J. (2012). Management of resource constrained devices in the Internet of things. *IEEE Communications Magazine*, 50(12):144–149.
- SmartThings (2015). *Developer Documentation: Release 1.0*. <https://media.readthedocs.org/pdf/smartthings/latest/smartthings.pdf>. Acessado em: 15-12-2015.