DESENVOLVIMENTO DE UM CONTROLADOR DE DISPOSITIVOS PARA AUTOMAÇÃO RESIDENCIAL UTILIZANDO TECNOLOGIA DE REDE ZIGBEE

Paulo Feitosa Júnior[1], Lucas Francisco[1], Kleber Manrique Trevisani[2]

[1]Faculdade de Informática da Universidade do Oeste Paulista (UNOESTE), Presidente Prudente, SP. [2]Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP). e-mail: kleber@ifsp.edu.br

RESUMO
O HMS (*House Management System*) é uma arquitetura projetada para a automação residencial, que define a integração de programas de computadores para o controle de dispositivos eletrônicos, manipulados por dispositivos móveis. Feitosa Jr et al. (2010) desenvolveram um software para esta arquitetura, com o objetivo de provar sua viabilidade. Este software apresentou alguns problemas, como a dificuldade de adicionar novos controladores de dispositivos de diferentes tecnologias de rede. Francisco e Trevisani (2010) realizaram modificações na arquitetura e no software desenvolvido, com o intuito de facilitar a adição de novos controladores de dispositivos de diferentes tecnologias de rede. Assim este trabalho foi concebido com o objetivo de desenvolver um controlador de dispositivos de uma tecnologia de rede diferente das já existentes no software, seguindo as modificações realizadas por Francisco e Trevisani (2010).
Palavras chaves: Automação residencial, sistemas distribuídos, ZigBee.

DEVELOPMENT OF A DEVICE CONTROLLER FOR THE RESIDENTIAL AUTOMATION USING THE NETWORK TECHNOLOGY ZIGBEE

ABSTRACT
The HMS (*House Management System*) is an architecture projected for the residential automation, which define the integration of computer programs to control electronic devices, that are controlled by mobile devices. Feitosa Jr et al. (2010) developed a software for this architecture. This software showed some problems, like the difficult to add new device controllers that has a different network technology. Francisco e Trevisani (2010) did some modifications on both, architecture and software developed, thinking of make the add of new device controllers of different network technology easier. In this sense this project has the objective of develop a new device controller, which a different network technology, following the modifications that Francisco e Trevisani (2010) did.
Keywords: automation, distributed systems, ZigBee.

1. INTRODUÇÃO

Atualmente as pessoas já estão percebendo como a tecnologia esta presente em suas vidas. Desde que acordamos com o alarme do aparelho celular, até quando saímos de casa pelo portão eletrônico estamos tendo um contato direto com soluções desenvolvidas para facilitar a vida das pessoas. À cerca de 20 anos atrás, WEISER (1991) criava o termo computação ubíqua para se referir a estes sistemas computacionais, que estão presentes nos mais triviais objetos do dia a dia, de maneira transparente ao usuário. “As tecnologias mais profundas são aquelas que desaparecem: são costuradas no tecido da vida cotidiana até ficarem indistinguíveis dela” (WEISER, 1991).

Para que a computação ubíqua fosse possível, muitos avanços foram necessários. A miniaturização dos componentes de hardware, como microprocessadores e memórias RAM, a diminuição dos custos dos componentes computacionais, impulsionado pelos avanços na área, e a evolução das redes de computadores e outros fatores, permitiram que as idéias de WEISER (1991) fossem enfim possíveis.

Com estes avanços, a automação residencial passou a ser algo comum para as pessoas. Dispositivos como portões eletrônicos e lâmpadas com sensores são alguns exemplos dela. Apesar de toda a evolução, sistemas mais complexos que controlem vários dispositivos de um ambiente, são ainda raros e caros.

O artigo esta dividido em seções. A seção 2 apresenta a arquitetura HMS e seus componentes. A seção 3 trata das alterações feitas na arquitetura e no software por Francisco e Trevisani (2010). A seção 4 mostra os objetivos e a justificativa do trabalho. A seção 5 trás os materiais e métodos utilizados no trabalho. A seção 6 revela os detalhes de implementação. A seção 7 mostra os resultados e discussão, enquanto a seção 8 apresenta as considerações finais.

2. ARQUITETURA HMS

A arquitetura foi inicialmente projetada por Bentlin e Trevisani (2008), como ilustrado na Figura 1, para permitir o controle de dispositivos eletrônicos instalados em um local (residência, escritório ou sala de cirurgia, por exemplo) a partir de dispositivos móveis conectados a Internet.

No trabalho de Bentlin e Trevisani (2008) o foco não era o projeto de uma arquitetura de software, mas a implementação do software. Assim, foi implementado um servidor, que recebia mensagens via *socket* (TANENBAUM; STEEN, 2007), enviadas por um dispositivo móvel (telefone celular) através de uma aplicação no dispositivo. O servidor se comunicava com uma placa controladora de lâmpadas através de uma porta paralela. Os comandos recebidos pelo servidor eram enviados a placa controladora para realizar as funções (Liga/Desliga) das lâmpadas. É importante destacar que a comunicação com novos dispositivos demandaria alterações diretamente no código fonte do servidor, o que pode ser considerado uma desvantagem.

Para solucionar tal problema, Feitosa Jr et al. (2010) adequaram a arquitetura inicialmente desenvolvida por Bentlin e Trevisani (2008) de modo a criar diversos componentes de software para diminuir o acoplamento. Assim, foi constatada a necessidade de padronizar a comunicação entre os componentes através de protocolos de comunicação. Tais protocolos foram projetados por Atalla e Trevisani (2009).

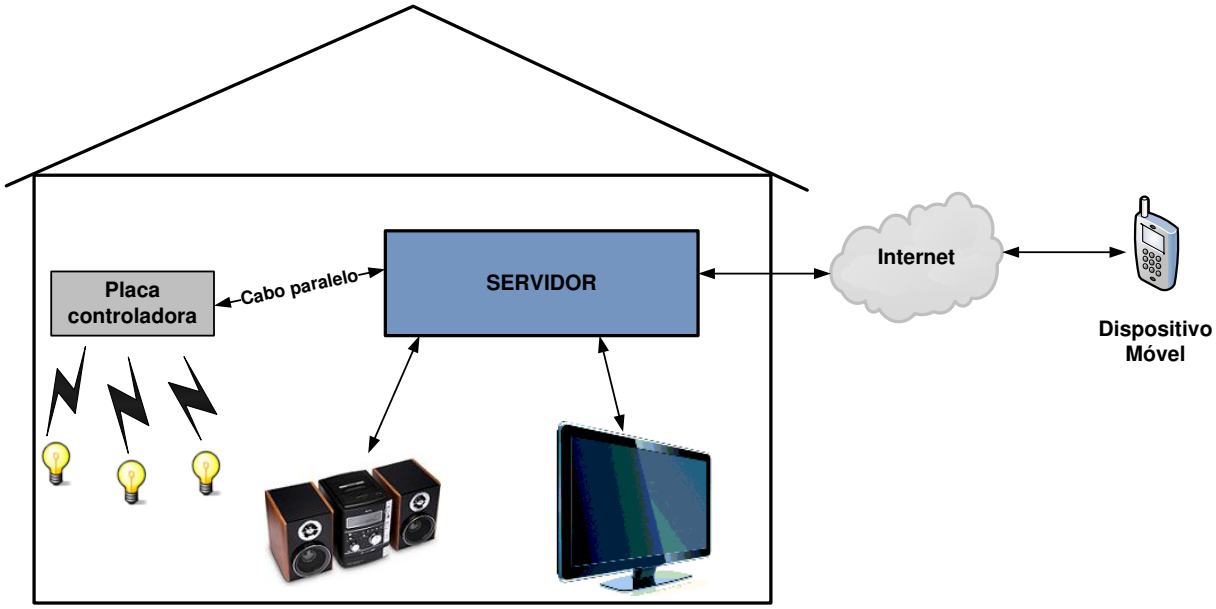

Figura 1. Visão Geral da arquitetura.

A placa controladora, que era totalmente implementada em hardware, foi substituída por *Device Controllers* (DC). Os *Device Controllers* são programas desenvolvidos para intermediar a comunicação entre o servidor e os dispositivos eletrônicos. Desta forma as alterações no software servidor são minimizadas ao se adicionar um novo dispositivo.

A arquitetura de Feitosa Jr et al. (2010), ilustrada pela Figura 2, é composta por componentes que se comunicam através de protocolos. Um dos componentes é a aplicação *Mobile House Control* (MHC), que é instalado no dispositivo móvel do usuário. A partir dela o usuário pode controlar remotamente dispositivos de um local, como uma residência por exemplo. Outro componente, o *House Server* (HS), é o servidor responsável por receber as requisições de manipulação do MHC e enviá-las aos controladores de dispositivo. Também é função dele comunicar-se com a Base de Dados que armazena as informações utilizadas pelo HS em suas operações. Os *Device Controllers* (DC) são programas embarcados responsáveis pelo controle dos dispositivos eletrônicos, como TVs e lâmpadas por exemplo. Após a requisição ser enviada do dispositivo móvel (MHC) para o servidor (HS), ela é encaminha pelo HS para o DC que efetivará a manipulação. Existe também uma interface com o usuário, o *House Control Center* (HCC), que possui as funcionalidades do MHC, e é responsável pelo gerenciamento de outras funcionalidades, como o gerenciamento de usuários e grupos, definição de permissões de acesso, agendamento de tarefas e definições de perfis de ambiente.

No desenvolvimento do trabalho de Feitosa Jr et al. (2010) constatou–se alguns problemas, como a necessidade de aumentar o nível de abertura da arquitetura para possibilitar a integração com diferentes tecnologias de uma forma mais simples, principalmente com relação a diferentes tecnologias de dispositivos eletrônicos. Na forma como a arquitetura foi definida, os controladores de dispositivos deveriam ser desenvolvidos estritamente de acordo com as diretrizes da arquitetura HMS, dificultando a integração de dispositivos já existentes e o desenvolvimento de novos controladores de dispositivos.

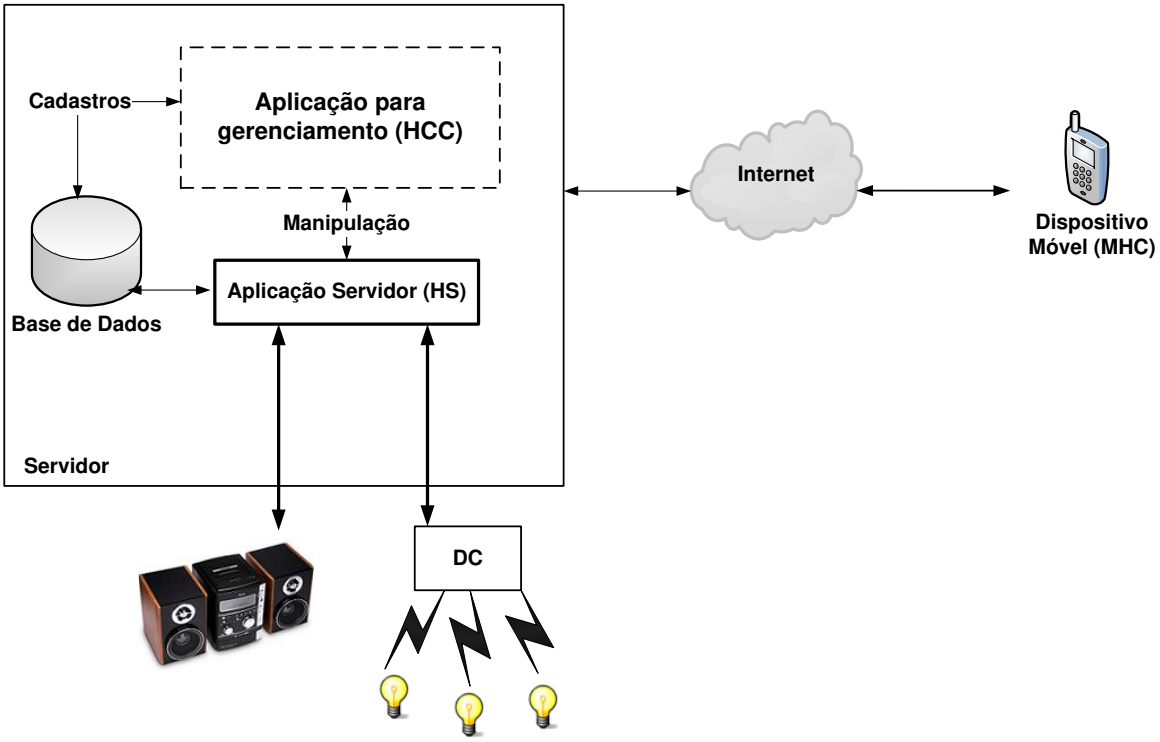

Figura 2. Visão detalhada da arquitetura inicial.

3. ALTERAÇÔES REALIZADAS NA ARQUITETURA

Para alcançar um nível de abertura satisfatório para a integração de controladores de redes existentes e novos controladores, foram necessárias algumas alterações na estrutura da arquitetura HMS, como pode ser visualizado na Figura 6.

Um fator que limita a arquitetura projetada por Feitosa Jr et al. (2010) é a dificuldade de trabalhar com tecnologias de rede heterogêneas, como redes ZigBee (SINEM). A integração com novas tecnologias de rede era possível, porém demandava um esforço de implementação que as inviabilizava, já que todos os pacotes enviados pelo servidor utilizavam endereços IPv4.

Para solucionar este problema foram realizadas modificações na arquitetura para retirar do HS a parte do código responsável pelo envio e recebimento das mensagens de rede. Essa responsabilidade passou a ser delegada a componentes de software específicos, chamados controladores de rede. É necessário um controlador para cada tecnologia utilizada (FRANCISCO; TREVISANI, 2010).

O HS passou a realizar uma nova funcionalidade. Ele agora passa a selecionar o controlador de rede para qual a mensagem será enviada. Essa seleção depende da identificação da rede em que o dispositivo eletrônico se encontra. Essa identificação se encontra na base de dados. Com estas alterações, facilitou-se a integração de novas tecnologias de rede, pois basta desenvolver e integrar o novo componente controlador com implementações específicas da tecnologia desejada, para que a comunicação seja feita. Além do controlador de rede, o servidor deve possuir uma interface física (placa controladora) (TANENBAUM, 2003; KUROSE; ROSS, 2006; FRANCISCO; TREVISANI, 2010).

Outro problema a ser solucionado é a forma como a arquitetura desenvolvida por Feitosa Jr et al. (2010) lida com a integração de novos dispositivos. Todo novo controlador de dispositivo deve ser baseado nos padrões da arquitetura, para garantir a compatibilidade com o servidor. Normalmente os controladores de

dispositivos possuem programas de computador embarcados. Nem todos os dispositivos eletrônicos têm capacidade suficiente para suportar estes programas. Outro ponto negativo da abordagem de Feitosa Jr et al. (2010) é que os fabricantes dificilmente adaptariam seus produtos para que eles fossem compatíveis com a arquitetura HMS (FRANCISCO; TREVISANI, 2010).

A solução selecionada para resolver estes problemas foi utilizar tradutores de tecnologia. Os tradutores possuem a função de padronizar a comunicação entre o HS e os DCs. Eles devem agregar funções específicas para a tradução. Foi adotada a abordagem de utilização de *plugins,* ilustrado na Figura 3 (FRANCISCO; TREVISANI, 2010).

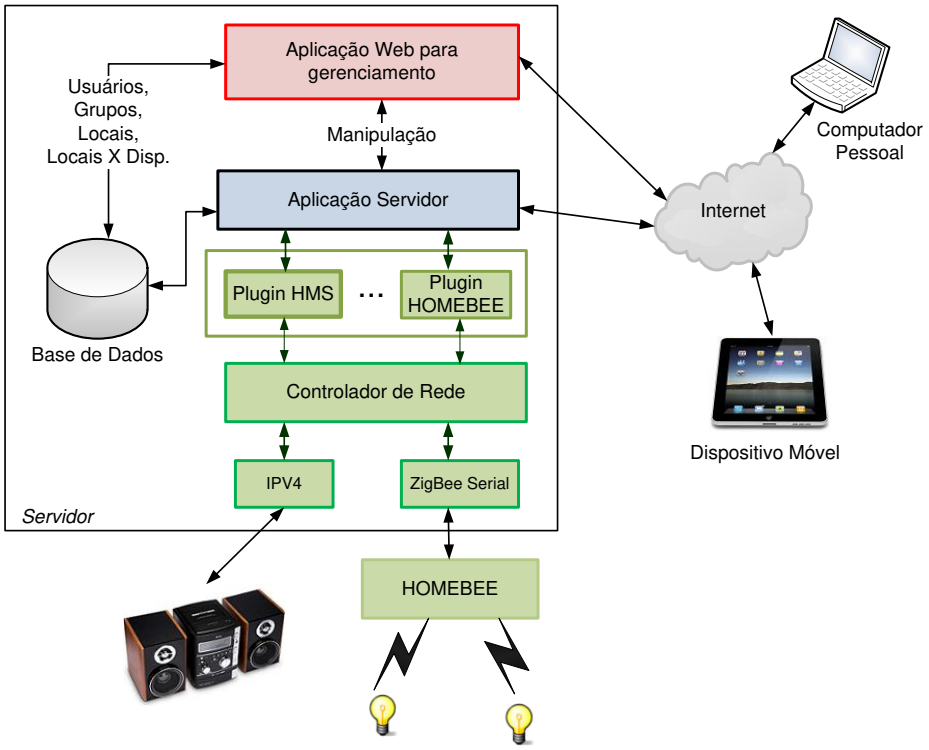

Figura 3. Visão geral da arquitetura após alterações.

4. OBJETIVO DO TRABALHO E JUSTIFICATIVA

Este trabalho tem como objetivo o desenvolvimento de um controlador de dispositivos para automação residencial, fazendo uso da tecnologia de rede ZigBee (SINEM). Este controlador de dispositivos deve ser compatível com o software desenvolvido por Feitosa Jr et al. 2010 e as alterações realizadas por Francisco e Trevisani (2010).

Ao final do trabalho pretende-se demonstrar que as alterações na arquitetura HMS propostas por Francisco e Trevisani (2010) facilitaram a integração de dispositivos eletrônicos e tecnologias de comunicação em redes distintas no software desenvolvido baseado na arquitetura HMS.

5. MATERIAS E METODOS

Para o desenvolvimento do projeto foi feito o uso de materiais específicos. Foram utilizados duas placas HOMEBEE (MANUAL DA PLACA RCON-HOMEBEE), três micro-controladores XBee Pro (MANUAL DA PLACA RCON-HOMEBEE), uma placa COM-USBEE (MANUAL DA PLACA RCON-HOMEBEE), além

de uma maquete de uma residência, vista na Figura 4, com o intuito de ilustrar como o sistema se comportaria em uma residência real.



Figura 4. Maquete.

Primeiramente foi feito um estudo com a tecnologia ZigBee (SINEM). Neste estudo as várias características desta tecnologia foram conhecidas e analisadas. A tecnologia foi fruto da união entre a ZigBee Alliance e o comitê IEEE 802.15.4. Ela foi concebida para atender a demanda por controladores que possuíssem um baixo custo, consumo e taxa de transferência de dados. Tais características a tornam perfeita para a proposta do projeto, uma vez que o baixo custo e baixo consumo a tornam uma opção atrativa e sua taxa de transferência, apesar de baixa, é adequada.

Os dispositivos que implementam a tecnologia ZigBee possuem quatro camadas, sendo duas, PHY e MAC, desenvolvidas pelo comitê IEE 802.15.4 e as outras, Network e Application, pela ZigBee Alliance. Enquanto as camadas PHY e MAC fazem as conversões e enviam os dados no *spectrum* de rádio, as duas outras camadas cuidam da criação da rede e dos dados das aplicações que trafegam nela (SINEM).

As redes da tecnologia ZigBee podem se organizar de três formas (Topologias): Star, Mesh e Cluster Tree. Os elementos desta rede podem ser de três tipos: PAN Coordinator, FFD (*Full Funcition Device*) e RFD (*Reduced Funciton Device*) (SINEM).

Um elemento PAN Coordinator faz o controle da rede, ou seja, ele cria a rede, adiciona os elementos a ela e envia os pacotes da aplicação. Em toda rede um elemento deve realizar o papel de PAN Coordinator. Ele usa os

FFDs para o envio dos pacotes a elementos que estejam fora de seu alcance. Diferente dos FFDs os RFDs não podem enviar os pacotes para outros elementos, eles servem apenas como dispositivos finais que recebem um pacote o executam e depois transmitem sua resposta (SINEM), como visto na Figura 5.

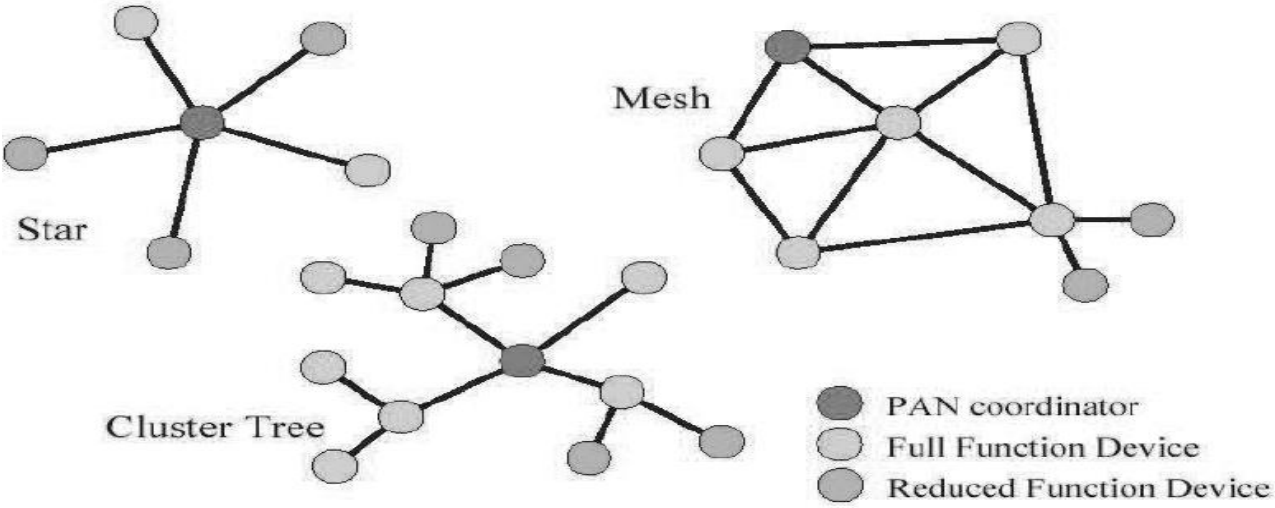

Figura 5. Topologias da Rede ZigBee (SINEM, 2004).

Após os estudos da tecnologia ZigBee, foram feitos estudos sobre a comunicação serial em Java. A biblioteca RXTX (JARVI) permite o acesso à porta serial e o envio de dados por ela. Assim, os comandos das placas HOMEBEE (MANUAL DA PLACA RCON-HOMEBEE) podem ser enviados ao micro controlador acoplado a placa COM-USBEE (MANUAL DA PLACA RCON-HOMEBEE) através da porta serial.

Estudos posteriores sobre a arquitetura HMS e os elementos que a compõem permitiram o desenvolvimento dos componentes necessários para o funcionamento do controlador de dispositivos. Definiu-se que os dispositivos a serem controlados seriam duas lâmpadas incandescentes. As funcionalidades que o controlador de dispositivos implementaria seriam as de liga e desliga. Para tanto foram utilizados os dois relês da placa HOMEBEE, como ilustrado na Figura 6.

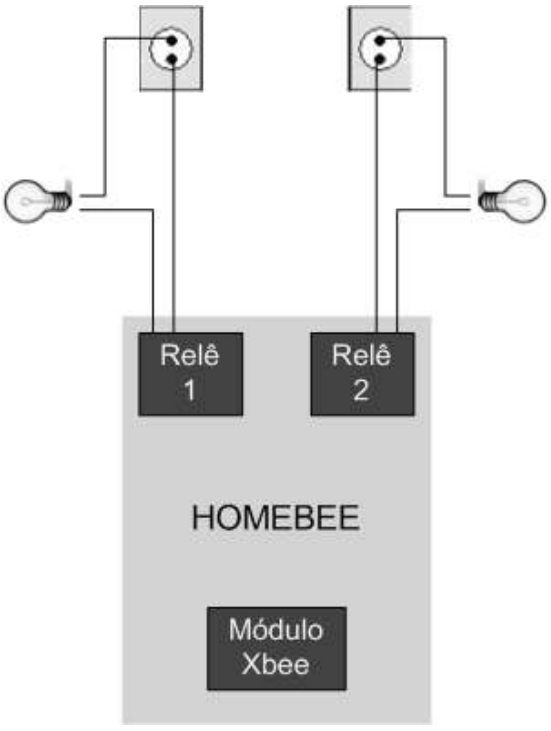

Figura 6. Controlador de Dispositivos.

6. DETALHES DE IMPLEMENTAÇÃO

Foram desenvolvidos dois artefatos de

software necessários para a comunicação do controlador de dispositivos desenvolvido e o software para a arquitetura HMS. O primeiro artefato desenvolvido foi um *plugin* (FRANCISCO; TREVISANI, 2010). Ele faz a "tradução" do pacote vindo do HS para o controlador de dispositivos desenvolvido. O ID do dispositivo dentro do pacote é transformado em um numero em hexadecimal, que identifica o relê em que o dispositivo está instalado na placa HOMEBEE. A função e o status são também transformados em números hexadecimais que os representam no controlador de dispositivos. Esta tradução é difícil, uma vez que a placa HOMEBEE não possui todas as funcionalidades previstas no software. Assim alguns dispositivos e algumas funções não estão disponíveis para o controlador de dispositivos.

Outro artefato desenvolvido foi um controlador para redes seriais, com foco no hardware COM-USBEE e suas particularidades. Este controlador de rede faz a "ponte" entre o HS, *plugin* e controlador de dispositivos. É ele que se comunica com o controlador de dispositivos e o HS, recebendo e enviando seus pacotes. Todos estes pacotes são enviados primeiro ao *plugin,* para o processo de tradução, como explicado no parágrafo acima. A Figura 7 ilustra os dois artefatos.

A placa HOMEBEE aceita dois tipos de comandos em hexadecimal: 0x7B e 0x7C (MANUAL DA PLACA RCON-HOMEBEE), sendo o primeiro a requisição de leitura de seus componentes e o segundo o comando de alteração dos status dos componentes. Um exemplo da tradução pode ser dado pela seguinte situação: Uma requisição de desligamento da luz é enviado para o controlador de rede. Ele é traduzido para o comando em hexadecimal 0x7B 0x01. Ele é então enviado para o DC que responde com o comando 0x57 0x01, que é traduzido para uma mensagem que o HS entenda, no caso luz desligada. O exemplo é ilustrado na Figura 8.

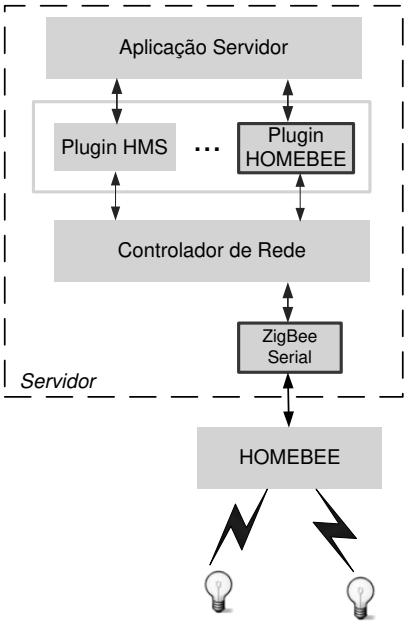

Figura 7. Artefatos em destaque (borda escura).

6. RESULTADOS E DISCUSSÃO

Após o desenvolvimento do controlador de dispositivos, visto na figura 12, utilizando a tecnologia ZigBee foi possível controlar através do software desenvolvido para a arquitetura HMS, duas lâmpadas incandescentes que estavam posicionadas na maquete. Ambas as funções programadas foram executadas sem problemas.

A arquitetura foi capaz de transmitir ao usuário a sensação de transparência quanto a qual rede o dispositivo requerido se encontra. O controlador de dispositivos se comportou de maneira estável, mantendo os dispositivos em um estado consistente com a base de dados da arquitetura. Nenhuma falha foi detectada na realização do teste em ambiente real.

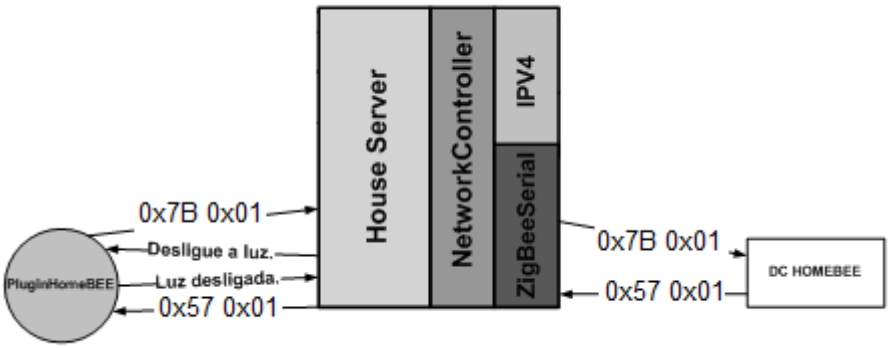

Figura 8. Processo de Tradução.

Observou-se que caso uma falha no fornecimento de energia ao controlador de dispositivos ocorresse, a base de dados da arquitetura poderia ficar inconsistente. Isto ocorreria pelo fato das placas utilizadas no desenvolvimento do controlador de dispositivos não controlarem este tipo de falha, não religando os relês que fossem desligados (Figura 9).

Observou-se ainda que as placas utilizadas no desenvolvimento do controlador de dispositivos eram limitadas para permitir o envio de eventos. Isto ocorre porque as placas utilizadas são totalmente passivas, respondendo apenas a pacotes enviados e nunca enviando um pacote originado de um evento em uma de suas entradas. Assim um sensor, por exemplo, teria que ter seu status lido constantemente pela arquitetura o que dificulta sua implementação.

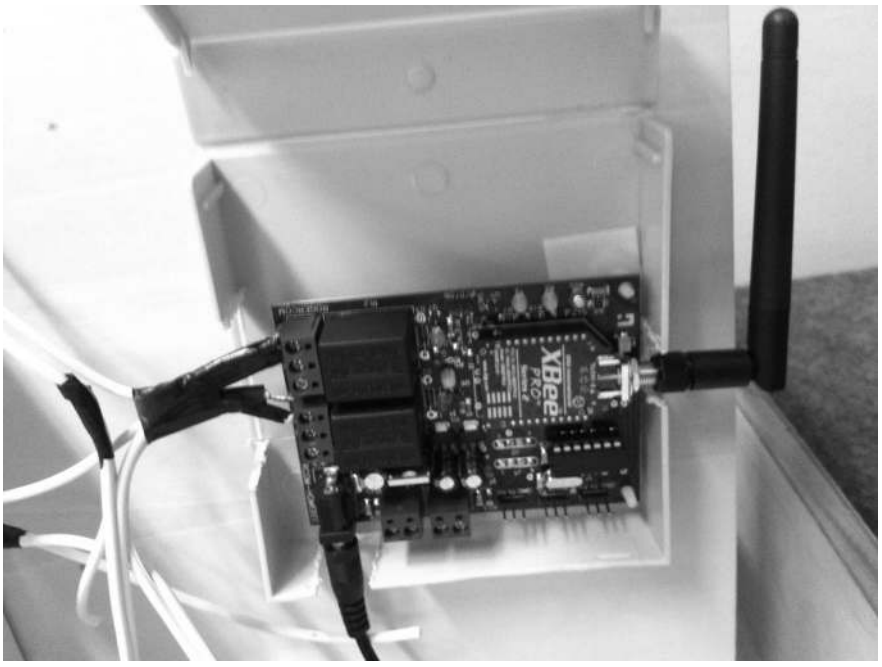

Figura 9. Controlador de Dispositivo Físico.

8. CONSIDERAÇÕES FINAIS

Com o teste em ambiente real foi possível provar a viabilidade da arquitetura. Antes a arquitetura havia sido testada apenas utilizando controladores de dispositivos lógicos (Virtuais), não sabendo ao certo seu comportamento no mundo real.

Conclui-se também que as placas

utilizadas para o desenvolvimento do controlador de dispositivos erram muito limitadas para operar outro tipo de dispositivo diferente de lâmpadas incandescentes. Um dispositivo com mais funções como uma TV, por exemplo, necessitaria de um hardware muito mais complexo que as placas utilizadas no desenvolvimento do controlador de dispositivos para as lâmpadas incandescentes.

Para trabalhos futuros sugeriu-se o desenvolvimento de novos controladores de dispositivos de outras tecnologias de redes diferentes de IPV4 e ZigBee foi proposto (Ex. Bluetooth, IrDA). Foi proposto também o desenvolvimento de um controle de estado para garantir a consistência dos dados do banco de dados da arquitetura, já que uma falha resultaria em uma base de dados inconsistente para o usuário da arquitetura. A utilização das entradas digitais presentes nas placas HOMEBEE adquiridas também foi proposto, uma vez que elas permitirão o controle dos dispositivos localmente, bem como a adição de novos dispositivos ao controlador de dispositivos desenvolvido por este trabalho.

REFERÊNCIAS

ATALLA, A. D.; TREVISANI, K. M. **Uma Extensão do Protocolo do Sistema MHC para Comunicação Segura.** In: XIV ENAPI - Encontro Anual de Pesquisa Institucional e Iniciação Científica, 2009, Presidente Prudente. ANAIS do XIV ENAPI, 2009.

BENTLIN, E.; TREVISANI, K. **MHC: Um sistema para controle de locais utilizando dispositivos móveis.** 2008**.** Trabalho de conclusão de curso (Bacharelado em Ciência da Computação) – Faculdade de Informática de Presidente Prudente – Unoeste, Presidente Prudente.

CORELI ERGEN SINEM. **ZigBee/IEEE 802.15.4 Summary.** Disponível em:< http://pages.cs.wisc.edu/~suman/courses/838/papers/zigbee.pdf>. Acesso em: 15 dez. 2011.

FEITOSA Jr. et al. **HMS: Um sistema aberto para automação residencial.** Presidente Prudente: Colloquium Exactarum, v.2 n.2, 2010.

FRANCISCO, L. V.; TREVISANI, K. M. **Um Protocolo de Comunicação para Manipulação e Reconhecimento de Dispositivos para o MHC.** In: XV ENAPI - Encontro Anual de Pesquisa Instituicional e Iniciação Científica, 2010, Presidente Prudente. Anais do Encontro de Ensino, Pesquisa e Extensão. Presidente Prudente : Unoeste, 2010. p.50.

KUROSE, J.; ROSS, K. **Redes de Computadores e a Internet: Uma abordagem Top-down**. 3ªed. São Paulo: Pearson Addison Wesley, 2006.

ROGERCOM. **Manual da Placa RCON-HOMEBEE.**
Disponível em: <http://www.rogercom.com/ManualHomeBee.pdf> Acesso em: 15 dez. 2011.

KEANE JARVI. **RXTX Wiki**. Disponível em: <http://rxtx.qbang.org/wiki/index.php/Main_Page>. Acesso em: 15 dez. 2011.

TANENBAUM, A; STEEN, M. **Distributed systems**. New York: Prentice Hall, 2007.

TANENBAUM, A. **Redes de computadores**. 4ªed. Rio de Janeiro: Campus, 2003.

WEISER, M. **The computer for the 21st century.** Scientific American, vol. 265, no.3, pp. 66-75, Setembro, 1991. http://dx.doi.org/10.1038/scientificamerican0991-94

Recebido em: 05/06/2013

Revisado em: 25/06/2013

Aceito em: 01/07/2013