

Rochester Institute of Technology

RIT Digital Institutional Repository

Theses

2005

M2MI service discovery middleware framework

Joel Donado

Follow this and additional works at: <https://repository.rit.edu/theses>

Recommended Citation

Donado, Joel, "M2MI service discovery middleware framework" (2005). Thesis. Rochester Institute of Technology. Accessed from

This Master's Project is brought to you for free and open access by the RIT Libraries. For more information, please contact repository@rit.edu.

ROCHESTER INSTITUTE OF TECHNOLOGY

M2MI SERVICE DISCOVERY MIDDLEWARE FRAMEWORK

MS Project Pre-Proposal

By JOEL VARELA DONADO
jxv0462@cs.rit.edu

M2MI has been thought of as a paradigm for applications running on ad hoc networks of fixed and mobile wireless proximal devices. For this to function, so far, we have relied on the human configuration of the devices sharing the same application. This includes, setting up the M2MI layer through Java, and starting the M2MI application. M2MI has worked fine under this scheme. A more general approach is to join a network and automatically discover any services available, in the case we have the M2MI layer already started. Afterwards, we can choose the service that we want to use. For this, a service discovery protocol and middleware would be in demand, applied in the same fashion by all devices implementing M2MI technology. Thus, this service middleware should be included into the M2MI library.

OBJECTIVE

The main objective of this project will be to develop the service discovery middleware as described above, with the capabilities of recognizing any available service on the M2MI layer of the devices which are being part of the proximal network.

For all devices which are running the M2MI objects, they will be able to talk in the same language when discovering a service. If a new device comes in, then it will send a message to all other devices out there requesting information on available services. The devices which are part of the ad hoc network will in consequence respond to this call, returning messages with information about the services they provide. This could be printing, faxing, chat, etc. This message should include just the basic information that the new host needs to know what services are out there for him to use, like name, description, and importantly, who is providing it. From this initial feedback, we must highlight the service description and provider. A service description may be nothing more than the application ID on the M2MI layer, with a string description of its name and a brief description of its capabilities. For this purpose, fields could be standardized on the M2MI layer and library. The later piece of information, the provider, I consider of particular importance due to the fact that after knowing which services are out there, we must want to use them. To use them, we must know where to get them. Following this thought, we will request the device providing this service for more information in order to start using it. The information provided must depend now on the application and the actual configuration of the device. If it is a printer, the printer driver may be provided, or if it is a chat application, then information on logon, users and chat rooms (if any) may be

provided. In either case, every device should be able to locate and obtain the necessary objects to use the desired service.

Along with the middleware described, good examples will be developed to show the functionality of it. We may be able to use the already developed MFS File System, or Collaborative Groupware applications.

Finally, this service discovery middleware could be compared to one or some of its cousins or related similar services, such as Jini's service discovery. This comparison would include architecture and functionality.

IMPORTANCE

This service discovery middleware framework will provide all M2MI applications with an invaluable tool to enhance the open functionality that M2MI was first designed to have. When the service discovery middleware is implemented, we won't have to worry about how to deploy our applications to our ad hoc network neighbors, or how they will interact with new devices. In a similar manner, devices entering an ad hoc network will easily be able to retrieve their proximal peers' services with minimal user intervention. The user will not have to download and separately compile and run the same software to use the services in the ad hoc network, rather he will just choose what he wants to do and will start interacting with the rest of the network. The framework is intended to work universally with every application developed for M2MI.

DELIVERABLES

1. M2MI Service Discovery middleware java classes and documentation in javadoc format
2. Technical Report on M2MI Service Discovery middleware (includes comparisons with different ad hoc networks service discovery middlewares)
3. Technical Report on the M2MI Service Discovery applications tests