

Management of Intelligent Library Wireless Sensor Networks Based on Runtime Model

Ping Zhang, Jianzhong Wang

College of Fundamental Education, Sichuan Normal University, Chengdu, 610068, China

Email: 835148187@qq.com

Keywords: wireless sensor networks; intelligent library; runtime model; software architecture

Abstract. To quick customize, develop and extend intelligent library internet of things system more efficiently, in this paper an approach based on runtime model to managing intelligent library wireless sensor networks is proposed. Firstly, manageability of intelligent library wireless sensors is abstracted as runtime models which automatically and immediately propagate any observable runtime changes of target resources to corresponding architecture models. Then a composite model of intelligent library wireless sensors is constructed through merging their runtime models in order to manage different kinds of devices in a unified way. Finally, a customized model is constructed according to the personalized management requirement and the synchronization between the customized model and the composite model is ensured through model transformation. Thus, all the management tasks can be carried through executing operating programs on the customized model. In the stack room area library conducted experiments and compared with the traditional method, this method can be more effective management of library facilities, more energy efficient and orderly, which can reach a 10.57% energy saving.

Introduction

As the Internet of things and cloud computing etc. information technology has become increasingly mature and the intelligent library become the new hot spot. In this paper, the intelligent library is the use of the wireless sensor network intelligent library Internet of things. Intelligent library IOT management system is mainly through the RFID reader to paste in the literature and the shelves of RFID tags, RFID library card reading, with background of RFID Management System and library automation integrated system, the literature, bookshelf, readers borrow, return frame, integration of the code used and safety anti-theft detection full range management. At the same time, it is through the spread of all kinds of on library sensing equipment, gathering library stack room, office and the self-study rooms of library environment such as light, electricity, temperature, all kinds of equipment switch, personnel location information and connect to the school LAN to realize intelligent identification, location, tracking and monitoring, analysis and management, which can automatically manage the light, fan or air conditioning equipment. In intelligent library Internet of things there are different types of application scenarios, sensing devices in different application scenarios corresponding to different objective things, so the connection between objective things in the scene and the sensor device needs to be established implementing the corresponding mapping relationship, even in the same application scenario, elements such as the management needs, the deployment environment, the sensing equipment etc. may also change during the long time running. So the management services in the intelligent library Internet of things on demand to the library information management has brought certain difficulty. The core of Intelligent library is the Internet of wireless sensor network, various types of sensors providing data read in different ways, and at the same time, data from the sensing device is real-time with a large amount of data, a certain attribute of the specific device to the mapping of data, and you need to write a lot of conversion code [1], which brings the greater complexity to the access of the information. In order to be able to quickly customize, develop and extend intelligent library Internet system, according to the requirements of management, improve the versatility and extensibility of intelligent library Internet system, and reduce the complexity of the development, in this paper the runtime software

architecture model will be introduced to intelligent wireless sensor network library management process, a runtime model based intelligent library wireless sensor network management method put forward.

The second section of this paper outlines the overall framework of the intelligent library wireless sensor network management method based on the runtime model. In the third section, the construction method of the runtime model of the sensing device of the intelligent library wireless sensor network is introduced. The fourth section describes the method of extracting and merging the running time model and the method of data synchronization. In the fifth section, the paper introduces the method of the combination model to the application scene model. The sixth section describes the experimental research and compares the traditional method with the method proposed in the paper. The seventh section summarizes the full text.

Intelligent library wireless sensor network management method

The more complex in the wireless sensor network management is to write a lot of conversion code data mapping to a sensing device. In the current process of system development of Internet of things, programming work is generally close to the operating system level, requiring the programmer to have a knowledge of the relevant technology of the underlying system, and focuses on the related problems of underlying system, rather than the application logic itself [1]. Commonly using C language series to directly call the management interface provided by sensing device, programmers must be familiar with different management interface of sensing equipment and C language series, to achieve their interaction. Dozens of other programming languages have been proposed, in the hope that, in the case of guarantee efficiency, simplify the development process of Internet of things system as much as possible, for example NesC, TeenyLIME and TinyDB etc. programming languages. Although the programming language to a certain extent decreases the difficulty of the development of the Internet of things application, the node heterogeneous system, application system and the underlying problem such as tight coupling still plague the developer. And some research work [2-3] based on service-oriented architecture tries to solve the problem of node data access, encapsulating the data collected by the sensing devices, and provide access interface in the form of RESTful services. However, the work is not abstracted to node system; developers still need to face to the underlying data programming. Model-driven development method can carry on the abstraction to the underlying system, helping to solve the problem of tight coupling application system with the underlying node [4]. And some work [5-7] on the application of wireless sensor net system modeling method, the work will apply model method to the demand of application development phase and design phase, effectively improving the level of abstraction of Internet programming. However, IOT systems need to be able to adapt to the demand, the environment and its own change in the process of long running, while the traditional model methods can't modify and debug management logic in the running process of the system.

Model in software engineering is the carrier between ideal software and management knowledge. The runtime software architecture model represents the overall architecture of the system with a set of manageable unit, and through runtime information display like internal structure, status, configuration hidden in the system to be described as a standard, management perspective oriented structural view, it can effectively improve the level of abstraction and automation degree the development of IOT systems [8]. The runtime software architecture model has won the wide attention in academia and industry. A lot of research work has proved the important role of it in a different system and management mode [9]. The runtime model is now widely used in different systems to support management function such as the system self-healing, management functions and dynamic adaptive and also used in some of the sensor network management [10].

In order to be able to quickly customize, develop and extend intelligent library network system according to the requirements of management, a model based on runtime intelligent library wireless sensor network management method is proposed. Methods summary is shown in figure 1. This method first constructs runtime model structure for different sensing equipment of intelligent library, and based on the interface of equipment management in the model layer monitoring different

sensing equipment is realized, the data synchronization between runtime model and gathering information maintained. Secondly based on the runtime model the data collected by different sensing devices of intelligent library is combined, realizing the unified management for different sensing equipment by combining the models of the scene; Finally the connection between the object of objective things and sensing devices in different application scenarios of intelligent library such as, intelligent library stack room, office and the self-study rooms of library is established, realizing the mapping from a portfolio model to different application scenario models by transformation, which makes all kinds of data collected by intelligent library sensing device performed in the form of various attributes of the objects in the objective world. The method is based on the existing management interface to construct the runtime model of sensing equipment, shielding isomerism of sensing device management interface, reducing complexity; Based on distributed sensing device runtime model the combination model is constructed, blocking the distribution of the sensing device. Through model transformation the mapping from a portfolio model to the application scenario model is implemented. With the methods above, all kinds of data collected by sensing equipment can be expressed in the form of things object properties in the objective world, thus the administrator can rapidly customize, develop and extend intelligent library network system facing application scenarios.

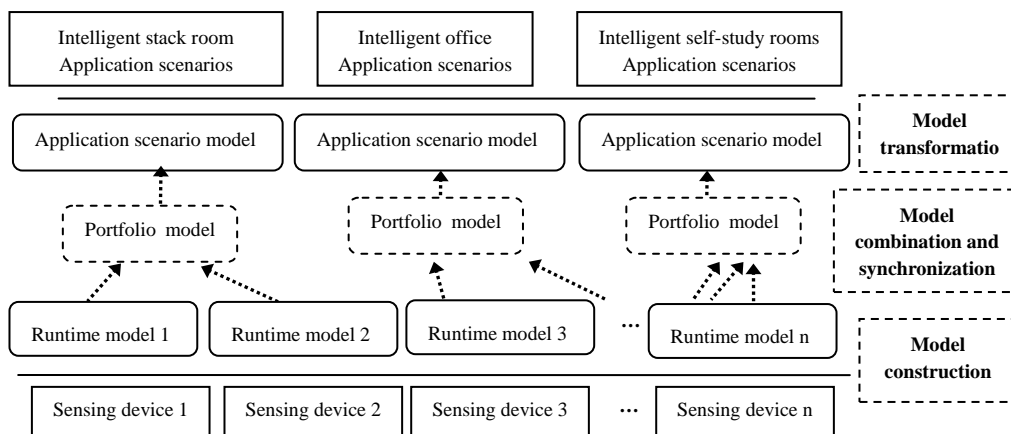


Figure 1 intelligent library wireless sensor network management method based on the runtime model

Intelligent library wireless sensor network model construction method

Intelligent library wireless sensor network is mainly to solve the information perception problems of all kinds of sensing equipment in CIOT, but it is of great difficulty and complexity to obtain information because of the diversity and heterogeneity of sensing equipment. Based on the basis of reference literature [11-12], with the tools of SM@RT [13-14] sensing equipment runtime model is constructed, the information acquired and processed in a unified way. SM@RT is a runtime software construction methods and tools driven by model proposed by the research team, including specific areas SM@RT language modeling language and SM@RT code generator. SM @ RT language code generator allows the users to define the runtime software architecture of meta-model and access model. Meta model defines the structure and manageable elements of the target system, and access model states the way of management of these elements in meta- model, that is, API.SM@RT generator of the target system is called, with meta- model and access model as input, the infrastructure maintaining runtime software architecture can be automatically generated, and then the underlying system of real-time state model is reflected in the runtime model. SM@RT source code can be downloaded from literature [15].

Intelligent library wireless sensor network model combination method

Intelligent library wireless sensor network consists of many different types of sensors, the application systems like intelligent stack room, office and the self-study rooms of library need the

unified analysis and processing of the data collected by different sensing device according to management goals. Intelligent stack room system requires monitoring and management for all the stack room lighting equipment, brightness sensors, temperature sensors and humidity sensors in library. Intelligent office systems requires analysis and processing of lighting equipment, brightness perception, temperature sensing equipment and office staff position in all office area of the library . Intelligent the self-study rooms of library system need analysis, processing and monitoring for the lighting equipment, brightness perception devices, temperature sensors and students location in the self-study rooms of library. These sensors need unified management, in order to realize the different types of complex management tasks. In this article the method of combination is adopted on the basis of sensing device model in operation, reorganization and merger in the form of a portfolio model, to realize the unified management of different types of sensors, including model extraction and merging, and data synchronization.

Extract and merge models

Combination model is that, to meet a specific management requirements in different application scenarios of intelligent library, the corresponding model fragment in the sensing device runtime model of some application scenarios of library is intercepted, which is restructured to merge into a new model - combination model. Scenario in figure 2 is about the managing of lighting equipment in a room whose Id is stack room 201, and the sensing device to collect location information and brightness information needs to be used at the same time, switch state decided according to the personnel and the brightness of stack room. Sensing device respectively provides the acquisition system of location information and the acquisition system of brightness information, and the two devices above will be combined on the basis of run time model, which forms a new portfolio model. In the specific case, the whole model of the location information acquisition system with Id for 201 and the tag element of brightness information acquisition system runtime model with Id for 201 are intercepted, which are merged into a new portfolio model specific to the management requirements. It is shown in figure 2. Student Number elements of combination model and their associated Tag elements reflect the staff within the stack room. The brightness property of stack room sensor element reflects the brightness of the stack room. When the stack room is provided for the use of a certain number of personnel and the stack room is lack of brightness, then open the stack room lighting, or conversely close the stack room lighting equipment.

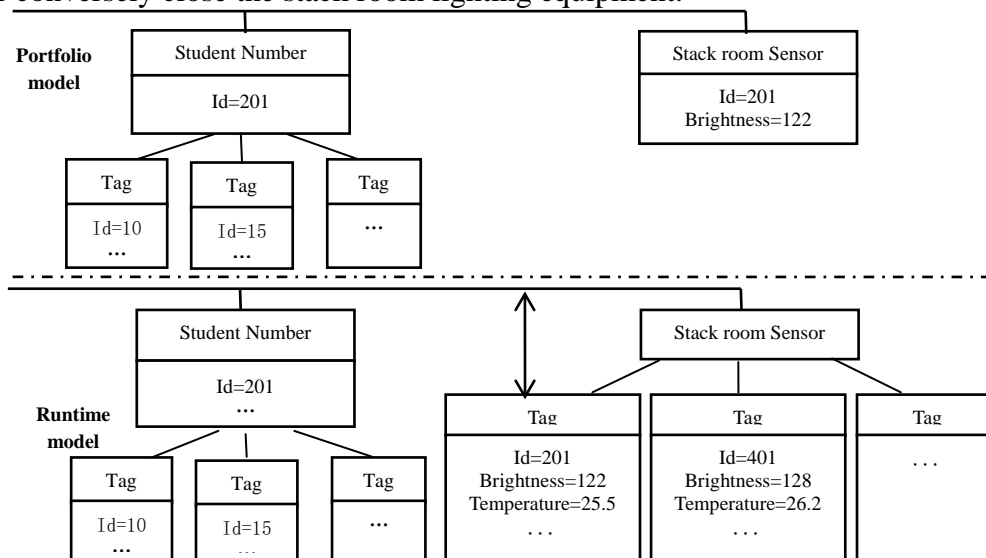


Figure 2 combination model constructions of stack room sections in intelligent library

Model pieces of combination needs to be able to obtain the information collected by the target device from a specific sensing device in the runtime model. Sensing equipment run time model and the model fragments customized by administrators are stored in the form of an XML file, each element in the model has one and only one path from the root node which can navigate to the

element. Model fragments of combination model from different sensor runtime models of intelligent library application scenario. They have not grammatical relation and can merge the root node of the each model piece as direct child nodes of the root nodes from combination model to. At the same time for element naming conflicts of each model fragment, a namespace needs maintaining, to replace and record conflict element naming.

```
<Stackroom_Sys : StackroomSystem >
  <Tags>
    <Tag id-"201"  brightness-"122"  temperatoure-"25.5"/>
    <Tag id-"401"  brightness-"128"  temperatoure-"26.2"/>
    ...
  </Tags>
</Stackroom_Sys : StackroomSystem >
```

Figure 3 Runtime model fragments of stack room sensing device in intelligent library

Data synchronization

On the basis of the combination model, the unity of the combination model of multiple systems management can be achieved only by maintaining the data synchronization between the combination model and the system running model. Portfolio model is composed of different system runtime model elements, whose data consistency is completed by the different model fragments. In the runtime system model, a copy of model fragment is deployed, and with the change of the model by polling mechanism found, the corresponding model operation automatically generated, is sent to portfolio model and executed. In all model operation types, only the Set, the Add and the Remove operations can produce the corresponding model change, these three kinds of model operation are described and illustrated in table 1.

Table 1 three basic models

Model operation	description	instructions
Set	$\exists \text{Type } n, n \text{ in condition of Constraint} \wedge \text{prop} \in n. \text{properties}$	In the model look for a certain type and constraint elements and assignment for its attribute
Add	$\exists \text{Type } s, \exists \text{Type } f, s \in f \wedge f \text{ in condition of Constraint} \wedge \text{props} \subseteq s. \text{properties}$	In the model look for a certain type and constraint elements as the parent node , add the type of child nodes and assignment
Remove	$\forall \text{Type } n, n \text{ not in condition of Constraint}$	In the model look for a certain type and constraint elements and delete them

When the running model is changes, the system maintains the portfolio model with the system runtime data consistency of the model through the depth first algorithm.

Depth-first algorithm for pseudo code is as follows:

```
{ According to the description of the model fragment, extract the new model fragments in the runtime system model; /* step a*/
While (Starting from the root node of the old model fragments, for each node in turn to compare in the old and new model fragments) /* step b */
  { { If ( the node is the root node)
    { If (the node = the root node) go to step d;
      else error; }
    Else Go to step c; }
  Check whether there is a new model fragment in this node ; /* step c*/
    If (the node is in the new model fragment ) {According to the child node information to generate the Remove operation ; go to step b; }
    Else Go to step d;}
  Check the node in the new model fragments of whether the property value changes; /* step d*/
    If ( attribute values are changed) {According to the child node information to generate the Set operation ;
```

```

go to step e;}
  Else Go to step e;
In the old and new model fragment, comparing child nodes of the node and check a child node in the new model
  fragments whether there is add the node ; /* step e*/
  if (Add child nodes) According to the child node information to generate the Add operation;
}
}

```

Intelligent library wireless sensor network model transformation method

In different application scenarios of library, data from the sensors often used to represent different objective things, needs to establish the correlations between application of objective things in the scene object properties and the data collected from sensing device. As is shown in figure 1, the specific application scenario model describes the objective things of library management scenarios, while a portfolio model says the data collection collected by sensors, elements and their attribute information in application scenario model coming from the portfolio model, and the relationship between them needs establishing, which can pass application scenario model to obtain the state information of objective things. In this paper, by constructing the element mapping relation between a portfolio model and application scenario model to implement a portfolio model to the application scenario model transformation, and maintain the attribute values of application scenario model elements and the corresponding elements in the portfolio model are equal. Model transformation program needs writing in order to transform the portfolio model to the application scenario model. In this article a method of model transformation is used based on combination model, the application scenario model and element mapping rules between them, automatically generating a conversion program to the corresponding model. Table 2 shows the one-to-one, one-to-many, and many-to-one, three basic mapping relation of model elements.

Table 2 Three basic mapping relations between the model elements

mapping relation	instructions	Note (special)
One to one	a type of elements in portfolio model corresponds to a type of elements in application scenario model.	The corresponding attribute to the attribute of element in application scenario model can be found in the corresponding element of combination model. When a portfolio model to the application scenario model transformation occurs, the attribute of element in application scenario model will be assigned according to the corresponding attribute of its associated elements in the array model.
One to many	a type of element in portfolio model corresponds to two or more types of elements in application scenario model.	When the model transforms, the elements of the combination model need to choose a kind of specific type of elements in application scenario mode for mapping according to their attribute information.
Many to one	two or more types of elements in portfolio model corresponds to a type of element in application scenario model.	The element attribute of application scenario model corresponding to the attribute in the portfolio model is distributed in two or more elements. In a portfolio model to the application scenario model transformation occurs, the elements in the scenario model need to query in the portfolio model and at the same time two or more elements of the assignment for its properties.

The element mapping relationship between the combination model and the application scenario model can be expressed as the combination of the three basic mapping relations. In this paper a set of mapping relationship describing rules and the method of automatically generated code is put forward, enabling the administrator to get the corresponding model transformation program by defining element mapping rules between the models. Element mapping rules between the models are described by an XML file. Keyword MEE represents mapping between the elements, MPP represents mapping between attributes and MSS represents mapping between scenarios and model.

These definitions are as shown in table 3.

On the basis of the above keywords, mapping rules between the elements are described according to the model mapping.

Table 3 the keywords of Element mapping rules between the models

Keywords	use	attribute
MEE	Used to describe the mapping relationship between elements	The value represents elements in the combination model The key represents the corresponding element in the application scenario model The type represents the type of the element mapping relation
MPP	Used to describe the mapping relationship between attributes	The value represents attributes in the combination model The key represents the corresponding attributes in the application scenario model
MSM	Used to describe the mapping relationship between the different application scenario attributes and portfolio model attributes	The value represents the attributes of the application scenario model The key represents the corresponding attributes in the application scenario model The node represents the attributes in the combination model The condition represents the attributes in the combination model

One-to-one mapping relationship between the model elements case is shown in figure 4. “Power” type element of Combination model and “Light” type element in the application scene model are "one-to-one" mapping relationship, in which MEE labels express “Power” to “Light” elements mapping, and MPP label expresses the mapping of the attribute id in “Power” element, switch to the “Light” element attribute id, on/off.

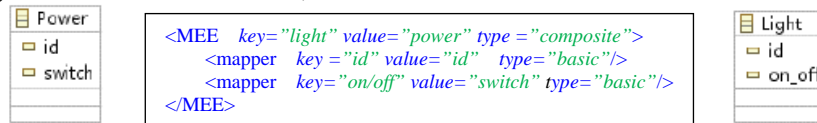


Figure 4 one-to-one mapping relationships between the model elements

The one-to-many mapping relationships between model elements case are shown in figure 5. It is “one-to-many” mapping relationship between the elements of the “Tag” type element in the portfolio model and the “Light” and “Air conditioner” type in the application scenario model. Tag elements might map for the Light or air conditioner elements, with a group of MPP tags for the type value "multi" to represent this mapping. When the Tag elements of the type attribute value is the "Light", the Tag element in the combination model maps for the Light elements in the application scenario model.

The many-to-one mapping relation between elements of the model can use many-to-one mapping relation of the Reader type element and Sensor element in the combination model and stack room element in the application scenario model. The elements of Reader in the portfolio model and stack room in the application scenarios model represent room. So MEE tag and MPP tag are used to describe the two types of elements and their mapping between attributes. However, stack room element and Reader elements contain more abundant information, which is present in the Sensor elements. So the query tags are used to describe the attributes of the Sensor element to the map Room elements corresponding attribute .The key and the value points in query tags use query tags to describe the attributes of the Sensor element to the map Room elements corresponding attributes. The key and the value in query tags separately represent the element properties in the application scenario model and composite model. The node attribute value for the element of Sensor says property for Sensor, and the condition attribute value "id = self. Id" is used to describe the element conditions to be met.

Mapping rules were in strong correlation with model conversion code. Every MEE label of mapping rule generates a method of code conversion, which implements mapping of elements. Each mapper tag of mapping rules generates a simple assignment statement of conversion code, which achieves the mapping of properties. Each query tag of mapping rules generates a query and

assignment statement of conversion code, which achieves lookup and mapping of attributes. Through the above definitions, model conversion code can be automatically generated on the basis of mapping rules.

Test results

In order to verify the feasibility and effectiveness of the Method, on the basis of school RFID perception equipment and stack room information system, the intelligent stack room based on runtime model of wireless sensor network system for stack room lighting, luminance and temperature information such as the management of the stack room has been achieved for part of the stack room. First we constructed the runtime model about RFID perception equipment and stack room information system of stack room lighting, brightness and temperature etc. The runtime model based on the RFID equipment can get basic information and the associated relations between reader-writer and labels in layer model for RFID device, the runtime model based on the stack room information system, the basic information and the associated relations between the stack room and all kinds of equipment are available in the model layer. And target information is combined and merged through the extraction. Then according to the requirements of stack room management application scenario model is structured, transiting a portfolio model to the application scenario model, eventually forming the management of the stack room. In addition, we have carried out the experimental comparison of the traditional solution and the method proposed in this paper. In the experiment the stack room 201 is selected, which has 90 spiral energy saving lamps, each 14W, Open time from 8:00 am to 17:30 PM. Experiments were carried out only for the illumination system of the test. With the traditional method, the stack room illumination system uses about 11970W every day, while with the method proposed in this paper the average use of the classroom every day is about 10705W. Average energy saved reaches 10.57%. The method can be more effective in the management of the stack room, more energy saving.

Conclusion

In order to improve the universality and extensibility of the library network system, a method of intelligent library wireless sensor network management based on the runtime model has been put forward. This method can more effectively manage the library facilities, more energy saving. In the future this method will be applied to other library, at the same time in practice further simplify programming and enhance the effect of management.

Acknowledgement

In this paper, the research was sponsored by the Department of Education Project of Sichuan Province.

References

- [1] Mottola L, Picco GP. Programming wireless sensor networks: Fundamental concepts and state of the art. *ACM Computing Surveys (CSUR)*, 2011,43(3). [doi: 10.1145/1922649.1922656]
- [2] Spiess P, Karnouskos S, Guinard D, Savio D, Baecker O, Souza LMSD, Trifa V. SOA-Based integration of the Internet of things in enterprise services. In: *Proc. of the IEEE Int'l Conf. on Web Services*. Los Angeles: IEEE Press, 2009. 968-975. [doi: 10.1109/ICWS.2009.98]
- [3] Janowicz K, Broring A, Stasch C, Schade S, Everding T, Llaves A. A restful proxy and data model for linked sensor data. *Int'l Journal of Digital Earth*, 2013,6(3):233-254. [doi: 10.1080/17538947.2011.614698]
- [4] Beckmann K, Thoss M. A model-driven software development approach using OMG DDS for wireless sensor networks. In: *Proc.of the Software Technologies for Embedded and Ubiquitous*

- Systems. Waidhofen-Ybbs: Springer-Verlag, 2010. 95-106. [doi: 10.1007/978-3-642-16256-511]
- [5] Thang NX, Geihs K. Model-Driven development with optimization of non-functional constraints in sensor network. In: Proc. of the 2010 ICSE Workshop on Software Engineering for Sensor Network Applications. New York: ACM Press, 2010. 61-65. [doi: 10.1145/1809111.1809128]
- [6] Shimizu R, Tei K, Fukazawa Y, Honiden S. Model-Driven development for rapid prototyping and optimization of wireless sensor network applications. In: Proc. of the 2nd Workshop on Software Engineering for Sensor Network Applications. New York: ACM Press, 2011. 31-36. [doi: 10.1145/1988051.1988058]
- [7] Rodrigues T, Dantas P, Delicato FC, Pires PF, Pirmez L, Batista T, Miceli C, Zomaya A. Model-Driven development of wireless sensor network applications. In: Proc. of IFIP 9th Int'l Conf. on Embedded and Ubiquitous Computing. Melbourne: IEEE Press, 2011. 11-18. [doi: 10.1109/EUC.2011.50]
- [8] Blair G, Bencomo N, France R. Models@ run.time. Computer, 2009, 42(10):22–27. [doi: 10.1109/MC.2009.326]
- [9] Wu YH, Huang G, Song H, Zhang Y. Model driven configuration of fault tolerance solutions for component-based software system. In: Proc. of the 15th Int'l Conf. on Model Driven Engineering Languages and Systems. Innsbruck: Springer-Verlag, 2012. 514–530. [doi: 10.1007/978-3-642-33666-933]
- [10] Chen X, Zhang W, Huang G, Li AP, Guo WZ, Chen GL. Management approach of wireless sensor networks based on runtime model. Ruan Jian Xue Bao/Journal of Software, 2014,25(8): 1696–1712 (in Chinese). <http://www.jos.org.cn/1000-9825/4665.htm>
- [11] Song H, Huang G, Xiong YF, Chauvel F, Sun YC, Mei H. Inferring meta-models for runtime system data from the clients of management APIs. In: Proc. of the 13rd Int'l Conf. on Model Driven Engineering Languages and Systems. Oslo: Springer-Verlag, 2010. 168-182. [doi: 10.1007/978 -3-642-16129-213]
- [12] Zhang W, Song H, Huang G. Object oriented accessing approach for wireless sensor network devices and data. Journal of Frontiers of Computer Science and Technology, 2011,5(12):1076–1084 (in Chinese with English abstract).
- [13] Huang G, Song H, Mei H. SM@RT: Applying architecture-based runtime management of internetware systems. Int'l Journal of Software and Informatics, 2009,3(4):439–464.
- [14] Song H, Huang G, Chauvel F, Xiong YF, Hu ZJ, Sun YC, Mei H. Supporting runtime software architecture: A bidirectional-transformation-based approach. Journal of Systems and Software, 2011, 84(5):711–723. [doi: 10.1016/j.jss.2010.12.009]
- [15] Peking University. SM@RT:Supporting models at run-time. 2009. http://code.google.com/_p/smattr/