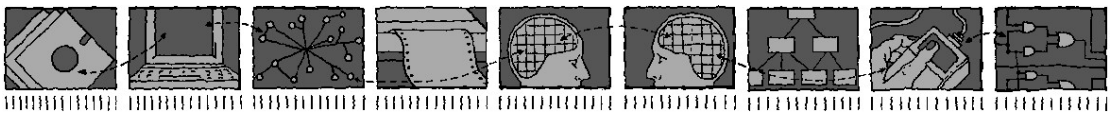
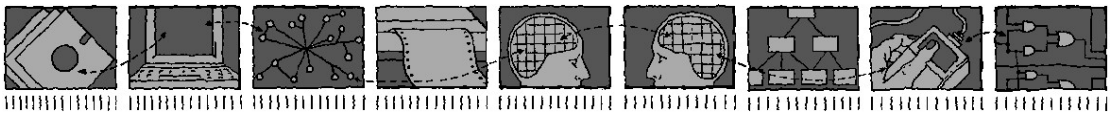


Computing Science and Mathematics
University of Stirling



Language Definition for REED

Xiang Fei and Evan H. Magill

Technical Report CSM-189

ISSN 1460-9673

May 11

Computing Science and Mathematics
University of Stirling

Language Definition for REED

Xiang Fei and Evan H. Magill

Department of Computing Science and Mathematics
University of Stirling
Stirling FK9 4LA, Scotland
Telephone +44-1786-467421, Facsimile +44-1786-464-551

Email x.fei@coventry.ac.uk and ehm@cs.stir.ac.uk

Technical Report CSM-189

ISSN 1460-9673

May 11

Abstract

Wireless Sensor Networks (WSNs) have emerged as an enabling technology for a variety of distributed applications. WSN middleware eases the development of these applications by providing a uniform programming environment. REED (Rule Execution and Event Distribution) is WSN middleware that supports both the distribution of rules and the events that trigger them. This report provides the full description of the REED language that is defined using a variant of BNF (Backus–Naur Form).

Keywords: Wireless Sensor Network, Middleware

1 Introduction

1.1 Context

Advances in sensor technology, wireless communications technology, and micro-processor technology make Wireless Sensor Networks (WSNs), consisting of distributed sensor nodes interconnected via wireless links, a potential and promising solution to a variety of distributed applications. PROSEN [1] (Proactive Condition Monitoring of Sensor Networks) is an example application to build a proactive condition monitoring system for wind farms.

To ease wireless sensor data collection, delivery and querying, WSN middleware is introduced that provides an Application Programming Interface (API) that shields the user from the complexities arising in WSNs. Based on the high-quality filtered data from sensor nodes and an AI-based data analysis, one of the main tasks of the PROSEN application/users is to implement proactive, goal-driven configuration management. This requires that the WSN system be programmable, especially when the system is running. To enable run-time programmability, REED (Rule Execution and Event Distribution) is proposed as middleware. This supports both the distribution of rules and the events that trigger them. REED employs a rule-based paradigm that allows sensor networks to be programmed at run time. This provides a flexible environment where applications and users can program the sensor nodes to allow their behaviour to adapt to the application's goals and the changing environment.

1.2 Scope and Objectives

In order to provide a clear description of the REED middleware, a formal notation is adopted. The REED language is defined using a variant of BNF described in [2].

1.3 Overview

In section 2, the full description of the REED language is provided.

2 REED Language Definition

<! Shorthand: oab is open_angle_bracket -->
<! Shorthand: cab is close_angle_bracket -->
<! Shorthand: ocb is open_curly_bracket -->
<! Shorthand: ccb is close_curly_bracket -->
<! Shorthand: osb is open_square_bracket -->
<! Shorthand: csb is close_square_bracket -->
<! Shorthand: orb is open_round_bracket -->
<! Shorthand: crb is close_round_bracket -->
<! Shorthand: ows is optional_white_space -->
<! Shorthand; sc is semi_colon -->
<! Shorthand; cond_set is condition_set -->
<! Shorthand; cond is condition -->
<! Shorthand; dest is destination -->
<! Shorthand; prpry is property -->
<! Shorthand; prpry_name is property_name -->
<! Shorthand; exist_op is exist_operator -->
<! Shorthand; comp_op is comparison_operator -->
<! Shorthand; logic_op is logical_operator -->
<! Shorthand; and_op is and_operator -->
<! Shorthand; or_op is or_operator -->
<! Shorthand; not_op is not_operator -->
<! Shorthand; gt is greater_than-->
<! Shorthand; lt is less_than -->
<! Shorthand; gte is greater_than_or_equal_to-->
<! Shorthand; lte is less_than_or_equal_to -->
<! Shorthand; ne is not_equal_to -->
<reedRuleSet> ::= <rule>+

<rule> ::= <rule_id> <equals><event_id> <event_handler>
<event_handler> ::= (<osb><cond_set><sc><actions><sc> <priority> (<csb>)+
<action> ::= < <set> <orb> <fact_id><dot> <prprt_name><comma><value><crb> | <send> <orb> <dest> ><comma> <event> <crb> >
<actions> ::= <action> (<comma><action>)*
<cond_set> ::= <cond> (<logic_op> <cond>)*
<cond> ::= <true>|<exist_op> <fact_id> <dot>< prprt _name> | <fact_id><dot> <prprt_name><comp_op><value> | <fact_id><dot>< prprt_name> <comp_op> <fact_id> <dot><prprt_name>
<logic_op> = <and_op>|<or_op>|<not_op>
<comp_op> ::= <gt>|<lt>|<gte>|<lte>|<ne>
<fact> ::= <state>|<event>
<fact_id> ::= <state_id>|<event_id>
<event> ::= <event_id><ocb><prprt>(<sc> <prpry>)*<ccb>
<state> ::= <state_id><property>+
<prprt> ::= <prprt_name><equals><value>
<oab> ::= <ows> “<” <ows>
<cab> ::= <ows> “>” <ows>
<ocb> ::= <ows> “{” <ows>
<ccb> ::= <ows> “}” <ows>
<osb> ::= <ows> “[” <ows>
<csb> ::= <ows> “]” <ows>
<orb> :: = <ows> “(” <ows>
<crb> :: = <ows> “)” <ows>
<equals> ::= <ows> “=”<ows>
<gt> ::= <ows> “>>”<ows>
<ls> ::= <ows> “<<”<ows>
<gte> ::= <ows> “>=”<ows>
<lse> ::= <ows> “<=”<ows>
<ne> ::= <ows> “!=”<ows>
<sc> ::= <ows> “;” <ows>
<comma> ::= <ows> “,” <ows>
<dot> ::= <ows> “.” <ows>
<exist_op> ::= <ows> “∃” <ows>
<and_op> ::= <ows> “&&” <ows>
<or_op> ::= <ows> “||” <ows>
<not_op> ::= <ows> “!” <ows>
<ows> ::= " " <ows> | ""
<set> ::= <ows> “set” <ows>
<send> ::= <ows> “send” <ows>
<dest> ::= <name>
<rule_id> ::= <name>
<event_id> ::= <name>
<state_id> ::= <name>
<fact_id> ::= <name>
<prprt_name> ::= <name>

3 Conclusion

This report has provided the full definition of the REED language for which prototype support has been implemented on PROSEN [3].

References

[1] University of Stirling. Proactive Condition Monitoring of Sensor Networks Home Page, http://www.cs.stir.ac.uk/~kjt/research/prosen/, October 2005.

[2] Wikipedia. Backus–Naur Form, http://en.wikipedia.org/wiki/Backus-Naur_Form, February 2011.

[3] Fei, X., and Magill, E. Rule Execution and Event Distribution Middleware for PROSEN-WSN. *Second International Conference on Sensor Technologies and Applications* (SENSORCOMM), pages 580-585, Cap Esterel, France, August 2008.