THE LANGUAGE DURA: A DECLARATIVE EVENT QUERY LANGUAGE FOR REACTIVE EVENT PROCESSING

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

vorgelegt von
Diplom-Informatiker
Steffen Hausmann

München, den 7. August 2014
CONTENTS

1 INTRODUCTION 1
   1.1 Motivation 1
   1.2 Contributions 3
   1.3 Organization 5

1 INTRODUCTION TO EMERGENCY MANAGEMENT 7

2 EMERGENCY MANAGEMENT IN CRITICAL INFRASTRUCTURES 9
   2.1 A Vision for Modern Emergency Management 9
   2.2 Incidents in Critical Infrastructures 12
   2.3 Three Challenging Use Cases 13

3 FOUNDATIONS OF DYNAMIC EMERGENCY MANAGEMENT 15
   3.1 Supervisory Control and Data Acquisition 15
      3.1.1 Basic Components of SCADA Systems 16
      3.1.2 Limitations wrt Emergency Management 17
   3.2 Complex Event Processing 18
      3.2.1 Composition Operator Based Languages 20
      3.2.2 Data Stream Query Languages 22
      3.2.3 Production Rules 24
      3.2.4 Timed Automata 26
      3.2.5 Logic Languages 28
      3.2.6 Summary 31
   3.3 Means for Reactivity in Event Processing 32
      3.3.1 Remote Procedure Calls 32
      3.3.2 Integration with Imperative Languages 34
      3.3.3 Event-Condition-Action Rules 35

II THE LANGUAGE DURA 37

4 FOUNDATIONS AND LANGUAGE DESIGN 39
   4.1 Declarative Rule-based Reasoning over Streams 39
      4.1.1 Reasoning with Rules 40
      4.1.2 Data Model 41
      4.1.3 Pattern-based Queries 42
   4.2 Full Acknowledgment of Orthogonal Concepts 43
   4.3 Deep Integration in a Uniform Language 44
   4.4 Time as a First Class Citizen 45
   4.5 Explicit Specification over Implicit Assumptions 45
   4.6 Clear Separation of Concerns 46
      4.6.1 Dimensions of Complex Events 46
      4.6.2 Dimensions of Stateful Objects 47
      4.6.3 Dimensions of Complex Actions 48

5 SYNTAX AND INFORMAL SEMANTICS 51
   5.1 Complex Events 51
      5.1.1 Representation of Events 52
5.1.2 Atomic Event Queries 54
5.1.3 Composite Event Queries 55
5.1.4 Temporal and other Conditions 58
5.1.5 Data Definition 62
5.1.6 Grouping and Aggregation 63
5.1.7 Existential Queries 65

5.2 Deductive and Reactive Rules 66
5.2.1 Range Restriction of Queries and Rules 67
5.2.2 Deductive Rules 69
5.2.3 Reactive Rules 73
5.2.4 Recursive Rules 74
5.2.5 Progressing Attributes 76

5.3 Stateful Objects 80
5.3.1 Representation of Stateful Objects 81
5.3.2 Atomic Stateful Object Queries 83
5.3.3 Integration with Event Queries 83
5.3.4 Modifying Stateful Objects 87
5.3.5 Creating and Terminating Values 88
5.3.6 Querying State Changes 89
5.3.7 State Based Processing 90
5.3.8 Resolving Simultaneous Updates 91
5.3.9 A Generalization of ECA Rules 93
5.3.10 Processing Static Data 94

5.4 Complex Actions 96
5.4.1 Properties of Physical Actions 96
5.4.2 Representation of Actions 98
5.4.3 Action Invocation 101
5.4.4 Action Composition 101
5.4.5 Temporal Dependencies 102
5.4.6 Execution Status 105
5.4.7 Temporal Assertions 106
5.4.8 Semantic Analysis for Actions 107
5.4.9 Complex Action Rules 108
5.4.10 Conditional Actions 110

5.5 Relations to XChangeEQ 113

6 EMERGENCY MANAGEMENT USE CASE 117
6.1 Preliminaries 117
6.1.1 Station Layout and Characteristics 117
6.1.2 Representation in Dura 118

6.2 Situation Assessment 120
6.2.1 Enrichment of Basic Events 121
6.2.2 Incident Categorization 122
6.2.3 Alarm Verification 122
6.2.4 Fire Size Estimation 125

6.3 Operation Mode Governance 127
6.3.1 Updating the Operation Mode 127
6.3.2 Detecting Operation Mode Crossovers 128
6.3.3 Identifying and Adapting the Operation Mode 128
6.3.4 Propagating Operation Modes 129
6.4 Immediate Reactions 130
6.4.1 Elevator Deactivation 131
6.4.2 Announcing Safe Evacuation Routes 132

III FORMAL SEMANTICS AND SEMANTIC ANALYSIS 135
7 SEMANTICS OF COMPLEX ACTIONS 137
7.1 Informal Introduction 137
7.1.1 Properties Specific to the Execution of Actions 137
7.1.2 Satisfying Temporal Dependencies 138
7.1.3 Basic Ideas and Approach 140
7.2 Formalization of Complex Actions 140
7.2.1 Formal Representation of Complex Actions 141
7.2.2 Formalization of Domain Knowledge 142
7.2.3 Formalization of Conditional Actions 143
7.3 Fixpoint Theory 144
7.3.1 Preliminaries 144
7.3.2 Fixpoint Iteration 146
7.3.3 Runtime Traces 150
7.3.4 Recapitulation of Notions 151

8 STATIC ANALYSIS OF COMPLEX ACTIONS 153
8.1 Requirements and Desirable Properties 153
8.1.1 Undesired Behavior of Complex Actions 153
8.1.2 Desirable Properties of Complex Actions 155
8.1.3 Requirements for the Semantic Analysis 156
8.2 Static Temporal Analysis 157
8.2.1 Preliminaries 158
8.2.2 Basic Ideas and Informal Introduction 159
8.2.3 Analogies to Skolemization 164
8.2.4 Desirable Properties Reconsidered 165
8.3 Analysis Algorithm 165
8.3.1 Pseudocode 166
8.3.2 Correctness and Completeness 167
8.3.3 Compliance Under Incomplete Knowledge 168
8.3.4 Variation for Non-definite Runtime Traces 168
8.3.5 Revision of Prior Work 169
8.4 Temporal Constraint Satisfaction Problems 169
8.4.1 Simple and Disjunctive Temporal Problems 169
8.4.2 Temporal Problems with Uncertainty 170
8.4.3 Temporal Problems with Predicates 170

9 FORMAL PROOFS 173
9.1 Preliminaries 173
9.2 Properties of Runtime Traces 175
9.3 Properties of Fair Actions 179