A

absence pattern 222
abstraction 11
active diagnostics 10, 108
active rules 43, 261
general structure 261
WebSphere Business Events 262
actuator 104
adaptors 80, 84, 93
agents, definition of 41–42
Aggregate agent 52, 126, 196
calendar partition 199
Daily statistics creator EPA 209
derivation functions 199
Fast Flower Delivery 197
state accumulation 197
stream processing 196
aggregating events 45
aggregation 20
frequency 199
alert 344
all pattern 220
always pattern 227
Ambient Devices 106
analytics 18
any pattern 222
Apama 80
approximate event processing 11
architectures 304–305
artificial intelligence 3
Assignment manager EPA 207
assignment system, EPA table 130
asynchronous interactions 41
Atom 93, 107
auditing 280
availability 16
definition of 267
average distance pattern 233

B

BAM. See Business Activity Monitoring
Bayesian networks 299
Bid enrichment EPA 208
Bid Request
EPA table 129
Bid Request creator EPA 206
Bid routing EPA 208
bottom-up event processing
application design 141
BPM. See business process management
BPML. See business process modeling notation
BRMS. See business rule management system
broker 37
buffer
Compose agent 201
temporal order 293
build vs. buy 16
building blocks 47, 121
consumer 50
context 51
definition of 47
event channel 56
event processing agent 51

C

capitalization convention 63
cardinality policies 238
CCL. See Continuous Computation Language
checkpointing 268
chronon 284, 292
See also temporal granularity
clock synchronization 292
coffee shop scenario 5
asynchronous approach 6
synchronous approach 6
common attributes 76, 337
definition of 73
for correlation 79
Compose agent 52, 117, 126, 199
Assignment manager EPA 207
buffer specification 201
example 328
filter 200
input streams 200
left buffer specification 201
match condition 200, 202
matching 200
parameters 201
Performance evaluation EPA 209
right buffer specification 201
unmatched left policy 201
unmatched right policy 202
composing events 45
composite contexts 164
call center example 166
definition of 165
examples 164, 170, 337–338
priority ordering 165–166
segmentation-oriented 168
types of 165
composite events 196
computerized event processing
active diagnostics 10
dynamic operational behavior 10
emergency alert system 8
examples 7–9
fraud detection system 8
information dissemination 10
luggage monitoring system 7
observation 9
online trading system 8
patient monitoring system 7
personalized banking system 8
plant management system 8
predictive processing 10
proactive 309
reactive 309
road tolling system 9
safety regulation compliance system 8
social networking site 9
consumption policies 240
bounded reuse 240
consume 240, 349
reuse 240
content-based filter 103
context 158
context aware computing 144
context building block 51
context dimension 145
composite 147
context initiator policies 162–164
definition of 163
effects 163
possibilities 164
refresh 164
context parameters 149
interval end 150
interval start 149
recurrence 150
temporal ordering 150
context partitions 145, 165, 185
Aggregate agent 196
context specifications 146
fixed number 185
scalability 266
semantic 145
spatial context 185
temporal contexts 185
temporal context 165, 198
context specification 145
temporal context 165, 198
detection time 66, 81, 149, 229, 284
accuracy of 292
assigned by platform 292
assigned by producers 292

D
DAILY statistics creator EPA 209
dashboard 106
metrics display 106
performance-monitoring 105
data feed 93
data types 70
area 71
line 71
point 71
simple 81
traffic monitoring example 71
database management 3
date, interpretation of 149
decoupling 11, 41
principle of 34
SOA and 39
decreasing pattern 230
dedicated event processing software 15
definition elements 47
consumer class 343
definition elements 47
definition of 145
dimension 146
explicit partition 147
Fast Flower Delivery 168–173, 336–341
filter and 177
fixed interval 149
identifier 146
initiator policy 147
instance 286
interval 286
partition 198
segmentation-oriented context 144, 161, 165, 198
spatial context 144, 154, 165
state context 165
stateful agents 144
state-oriented context 144, 160
stream windows 144
temporal context 165, 198–199
three uses in EPAs 144
type 146–147
Continuous Computation Language 80
derivation 122, 188
Enrich agent 191, 193
expression 198
formula 122
iterative splitting 196
Split agent 195
derivation expression 189
definition of 188
derivation functions 197–198
table 199
derived events 287, 297
assigning timestamps 287
detection time 290
occurrence time 288
detection time 66, 81, 149, 229, 284
accuracy of 292
assigned by platform 292
assigned by producers 292

D
detection time (continued)
definition of 68
geneneration of 69
of derived events 203, 290
ordering 69
detectors 92, 94
deterministic event processing 11
development environments 263
Apama Studio 263
graphically based 264
StreamBase 264
text-based 263
dimensional patterns 228–236
distributed processing 273
Driver enrichment EPA 207
Driver event consumer 97, 342
Drivers’ Guild event consumer 113
dynamic consumer registration 37, 54
dynamic operational behavior 10
edges 117
Enrich agent 52, 126, 128
Bid enrichment EPA 208
Bid Request creator EPA 206
Driver enrichment EPA 207
Filter agent 184
query expression 193
query parameter 193
Request enrichment EPA 192
transformation 191
enrichment 45
example 327–328
enterprise application integration 45
enterprise service bus 16, 80, 308
entity distance location 158–159
entity reference attribute 72
ESB. See enterprise service bus
Esper 25
Etalis 80
evaluation policies 237
event annotation, definition of 68
event attributes
definition of 63
information gathered by 63
semantic roles 72
simple data types 70
event batching 36, 38
event bus 134
event causality
definition of 300
induced 301
potential 301
predetermined 300
tracing 301
event certainty, definition of 68
event channel 37–38, 56, 134–138
building block 49, 51, 56
definition element 134–135
definition of 134
distribution 36
event consumer and 111
explicitly modeled 44, 118
Fast Flower Delivery 345
identifier 135
implementation 37, 56
implicit 44, 89, 134
input terminal 135
modeled 56, 89
notion 134
output terminal 135
quality of service assertions 135
routing schemes 135–136
event cloud 144
event composition attribute 65
event consumer 17, 34, 41, 101
abstract type 101–103
annotation 102
API approach 111
authorized 268
building block 50
category 102
category attribute 103
class 51, 54, 101
complications 110
definition element 101–103
definition of 42
details 102
Driver 112, 343
Drivers’ Guild 113, 344
Fast Flower Delivery 112–113, 343–344
generalization of 103
graphical representation 53, 101
hardware 104
human interaction 105
identifier 102
identity of 36
instance 101
interaction patterns 110
interfacing mechanisms 111
interfacing with 109–112
intermittently connected 38
members of the class 110
registration 110–111, 343
scalability 265
software 108
specialization of 103
Store 112, 344
time categories 104
varieties of 103–109
event correlation 20
event distance location 159–160
event distribution 41, 95
characteristics of 35
congruent 55
serial 55
event identity 81
definition of 68
event instance See event object
event interval context
time definition 150
time initiator policy 151
time, examples 338
time examples 152, 170
expiration event count 151
expiration time offset 151
initiator 150
initiator event list 151
temporal ordering 151
terminator 151
event log 36, 38, 41, 108
event object 5, 62, 101
header 63
information carried by 63
payload 63
event occurrence 62
interval 286
event patterns 215–219
definition of 216
event processing
and related concepts 16–21
application structure 41
applications 304
approximate 11
architecture 40
BAM and 308
BI and 308
BPM and 308
business value, and 14–16
causality 299
centralized architectures 304
definition 10
deterministic 11
distributed architecture 304
diversified architecture 304
embedded 307
ESBs and 308
intermediary 311
intermediary 41, 44
### Event Processing (continued)

- **Introduction**: 10–14
- **Logic Abstraction**: 44
- **Main Concepts**: 40–47
- **Misconceptions**: 3
- **MOM and**: 308
- **Monolithic Architecture**: 304
- **Narrow to Wide**: 304
- **Packaged Applications and**: 308
- **Programmer-Centric**: 306
- **Reasons for Using**: 13
- **Retraction**: 299
- **SOA and**: 39
- **Software Engineering**: 311
- **Stand-Alone**: 307
- **Stateful**: 45
- **Table of Languages**: 26
- **Trends**: 303
- **Virtual Platforms**: 310

### Event Processing Agents

- **Aggregate Agent**: 52
- **Building Block**: 51
- **Compose Agent**: 52
- **Context Handling**: 145
- **Definition Element**: 127
- **Enrich Agent**: 52
- **Failure Terminal**: 188
- **Fast Flower Delivery**: 331
- **Filter Agent**: 51, 123
- **Functions**: 121
- **Identifier**: 128
- **Nested**: 128
- **Pattern Detect Agent**: 52, 127
- **Project Agent**: 53
- **Scalability**: 265–266
- **Seven Building Blocks**: 121–133
- **Split Agent**: 52
- **Transformation Agent**: 52, 125
- **Translate Agent**: 52
- **Type**: 128
- **Types of**: 123
- **Variants of**: 51
- **Vs. Event Producer**: 87

### Event Processing Applications

- **Defining Event Types**: 81
- **Differences Between Platforms**: 79, 81
- **Query Capability**: 95

### Event Processing in Action

- **Building Block Editor**: 47
- **Language-Based Part**: 24
- **Website Details**: 24

### Event Processing Networks

- **As Abstractions**: 54
- **As Event Consumer**: 87
- **As Event Producer**: 87
- **Conceptual Examples**: 140
- **Data Transfer**: 118
- **Dynamic**: 54
- **Examples**: 54
- **Explicit Model, Benefits**: 121
- **Fast Flower Delivery**: 326
- **Feedback in**: 117
- **Graphical Notation**: 53, 116
- **Hierarchical Design**: 118
- **Implementation**: 119
- **Modeling**: 47–57
- **Nesting**: 118, 127
- **Relation to Applications**: 44
- **Event Processing Platform**: 14
- **Event Processing Software**: 15–16
- **Event Producer**: 17, 34–35, 40, 42, 87
- **Abstract Type**: 87–90, 342
- **Annotation**: 89
- **API Attachment**: 96
- **Authorized**: 268
- **Building Block**: 50
- **Category**: 88
- **Class**: 87
- **Definition Element**: 87–90
- **Definition Element Type**: 89
- **Details**: 88
- **Event Distribution Delegation**: 37
- **Fast Flower Delivery**: 96
- **Generalization of**: 90
- **Graphical Representation**: 53
- **Hardware**: 91
- **Human Interaction**: 93
- **Identifier**: 88
- **Identity of**: 42
- **Instance**: 87
- **Interfacing Mechanisms**: 95
- **Interfacing with**: 94–96
- **Presence Detection**: 93
- **Protocol Attachment**: 96
- **Queriable**: 95
- **Query Capability**: 89
- **Relationship with Events**: 41
- **Scalability**: 266
- **Software**: 92
- **Specialization**: 342
- **Specialization Example**: 97
- **Varieties of**: 90–94
- **Vs. Event Processing Agent**: 87

### Event Streams

- **As Event Consumer**: 87
- **As Event Producer**: 87
- **Conceptual Examples**: 140
- **Data Transfer**: 118
- **Dynamic**: 54
- **Examples**: 54
- **Explicit Model, Benefits**: 121
- **Fast Flower Delivery**: 326
- **Feedback in**: 117
- **Graphical Notation**: 53, 116
- **Hierarchical Design**: 118
- **Implementation**: 119
- **Modeling**: 47–57
- **Nesting**: 118, 127
- **Relation to Applications**: 44
- **Event Processing Platform**: 14
- **Event Processing Software**: 15–16
- **Event Producer**: 17, 34–35, 40, 42, 87
- **Abstract Type**: 87–90, 342
- **Annotation**: 89
- **API Attachment**: 96
- **Authorized**: 268
- **Building Block**: 50
- **Category**: 88
- **Class**: 87
- **Definition Element**: 87–90
- **Definition Element Type**: 89
- **Details**: 88
- **Event Distribution Delegation**: 37
- **Fast Flower Delivery**: 96
- **Generalization of**: 90
- **Graphical Representation**: 53
- **Hardware**: 91
- **Human Interaction**: 93
- **Identifier**: 88
- **Identity of**: 42
- **Instance**: 87
- **Interfacing Mechanisms**: 95
- **Interfacing with**: 94–96
- **Presence Detection**: 93
- **Protocol Attachment**: 96
- **Queriable**: 95
- **Query Capability**: 89
- **Relationship with Events**: 41
- **Scalability**: 266
- **Software**: 92
- **Specialization**: 342
- **Specialization Example**: 97
- **Varieties of**: 90–94
- **Vs. Event Processing Agent**: 87

### Event-Based Programming

- **And Events**: 31–40
- **Definition of**: 39

### Event-Condition-Action Rules

- **261

### Event-Driven Architecture

- **4, 33

### Event-Driven Behavior

- **4–7

### Event-Driven Business Process Management

- **308

### Events

- **Aggregating**: 45
- **Composing Streams**: 45
- **Definition of**: 4, 33
- **Derived, Definition of**: 42
- **History**: 95
- **Identity of Derived Event**: 202
- **Interleaved**: 55
- **Interleaving**: 117
- **Logical Structure of**: 62
- **Messages and**: 36
- **Raw, Definition of**: 42
- **Real-World**: 4
- **Representing Changes of State**: 33
- **Representing Occurrences**: 33
- **Retraction**: 295, 299
- **Scalability**: 265
events (continued)
splitting 45
storage 38
type of derived event 203
types 5
visualization 107
vs. requests 33
eventual consistency 294
exceptions 7
excess type 238
explicit model 121
explicit partitions 156
segmentation-oriented context 162
Extensible Markup Language 80
Extensible Stylesheet Language Transformations 80

F
Failure terminal 193
false negative 297
false positive 297
Fast Flower Delivery
activity monitoring 24
assignment phase 23
bid phase 22
channels 136
delivery phase 23
event consumer definition elements 112
event processing agents 129
patterns 244–247
producers 96, 341–343
ranking evaluation 23
specification 21–24
Filter agent 51, 123, 184
Bid routing EPA 209
definition of 123
element 328
expression 123
 schematic view 183
stateful, stateless expressions 185
filter expressions 183
definition of 178
event content 178
event header 178
event type 178
features table 179
Transformation agents 188
filtering 16, 19–20, 44, 55, 80,
121, 127
event processing contexts 185
examples from EPLs 210–211
in EPAs 183
in EPNs 177–183
in event processing agent 185
Input terminal and 177
optimization of 111
order of 185
Split agent 195
stateful, in EPA 186
where specified 177
with XML and XSLT 179
filters combining multiple 186
Compose agent 200
failed 177
Failure 193
filtered-in 177
filtered-out 177
if-then-else 177
Non-filterable terminal 185
rate-limiting 177
up-front filtering 178
where specified 177
first m filter 186
sampling 186
fixed interval 203
context, example 169, 337
contexts 149–150
fixed location 155
context examples 156
entity 156
location attribute 155
partition identifier 156
spatial relation 155
flat event 80
fraud detection system 4
free-format data 62
generalization, definition of 74
geocoding 155
gEOFs 155, 205
geo spatial terms 71
spatial context 154
global state element 118, 128,
138–140
building block 51
definition element 139
Enrich agent 191
event log 138
event processing state 138
example of update 327
external entity state 138
Fast Flower Delivery 345
Fast Flower Delivery table 139
general information 139
identifier 139
Location service 205
metadata 139
query parameter 193
reference data 138
type 139
updating 139
GPS Location, payload attributes 76
GPS sensor event producer 97, 342
g graphical user interface systems 7

H
header 63
application defaults 67
attribute 65–70
attribute indicators 67, 69, 75
attributes 63, 202
attributes table 204
descriptive information 64
event type description attribute 65
generic information 64
platform-independent attributes 63
validation 202
human interaction 93
Driver event consumer 343
event consumer 105

I
IDE 81
identifiers 62
imperative language style 25
imperative programming language 43
implicit partitions 156
increasing pattern 230
inexact event processing 295–299
handling 298
inexact matching 295, 297
malicious source 296
probability indicators 298
probability-based methods 298
temporal anomalies 296
uncertainty 295
unreliable source 296
infinite recursion, risk of 117
INDEX

356

information dissemination 10
input terminal 53, 101, 116
absence of 87
Driver event consumer 112
Drivers’ Guild event consumer 113
event type 103
filter 103
filtering and 177–183
identifier 102
source 102–103, 118
Store event consumer 113
instrumentation 92
intelligent event processing causality networks 312
pattern detection EPAs 311
predicted events 312
interaction patterns
approach choices 95
detector-style 94
sensor-style 94
interfacing mechanisms 95
API approach 96
maps 107
protocol-based 96
IP Multicast 37

J

Java 7
Java Message Service 16, 37, 81, 96
JavaScript, geocode translation 206
JMS. See Java Message Service
join operator 52
Compose agent 200
outer 202
two-way 202

K

key performance indicators 18, 106
KPI. See key performance indicator

L

languages for event processing 26
API 79
data types 81
header attributes 81
lack of standards 256
stream-oriented 80
styles in use, list 43
transformations 190
XML 80
XSLT 80
last m filter 186
life line 32
line of business applications 108
links 55, 117, 134
specification of 118
load balancing 273
location attribute 154
location data type 71
Location service EPA 156
EPA table 130
location tracking 107
logic programming-based rules 262
logic rules 43

M

Manual Assignment, payload attributes 77
matching 122
Compose agent 200
sets 122, 127
matching sets 240
by pattern type 243–244
max distance pattern 233
members, definition of 74
membership 74
message 20
message-oriented middleware 16, 20, 37, 96, 308
messages, one-way 35
metadata 139
metrics 106
dashboard display 106
min distance pattern 234
mixed pattern 231
modal patterns 227–228
always 227
sometimes 228
model-driven approach to modeling 47
MOM. See message-oriented middleware
monitor 92
MonitorScript 25
monolithic architecture 304
multiple results policy
combine 207
definition of 194

N

nesting 118, 127–128
Network and Systems Management 20
news feed 93
node-sets, XML 179
non-decreasing pattern 231
non-event pattern 222
Non-filterable terminal 185
non-functional characteristics 54
non-functional properties 264–269
availability 267
optimization types 271–276
performance objectives 269–271
scalability 265
security 268
validation and auditing 276–280
non-increasing pattern 231
notification 36
Not-selected terminal 216, 221
illustration 221

O

OASIS 17, 84
object-oriented language 32
observation 9
occurrence time 66, 149, 158, 229, 284
accuracy of 291
ambiguity examples 284
approximate 67
assignment 289
definition of 67
instantaneous event 284
interval 285
nonsupporting systems 290
occurrenceTime 183
of derived events 203, 288
ordering 69
policy options 290
synchronization 291
time interval 67
time point 67
time point vs. interval 284
use of 77
zero duration event 284
OMG 17
one-event in/one-event out 45
one-way
event interaction 35
messages 35
<table>
<thead>
<tr>
<th>Index Entries</th>
<th>Page Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>online trading system</td>
<td>33</td>
</tr>
<tr>
<td>open-content data</td>
<td>62, 80</td>
</tr>
<tr>
<td>indicator</td>
<td>73</td>
</tr>
<tr>
<td>operational intelligence</td>
<td>18</td>
</tr>
<tr>
<td>optimizations</td>
<td></td>
</tr>
<tr>
<td>black box</td>
<td>311</td>
</tr>
<tr>
<td>code generation</td>
<td>274</td>
</tr>
<tr>
<td>distributed processing</td>
<td>273</td>
</tr>
<tr>
<td>EPA assignment</td>
<td>271</td>
</tr>
<tr>
<td>EPA code</td>
<td>274</td>
</tr>
<tr>
<td>event processing</td>
<td>310</td>
</tr>
<tr>
<td>execution</td>
<td>276</td>
</tr>
<tr>
<td>load balancing</td>
<td>273</td>
</tr>
<tr>
<td>parallel processing</td>
<td>273</td>
</tr>
<tr>
<td>partitioning of EPA instances</td>
<td>272</td>
</tr>
<tr>
<td>routing</td>
<td>276</td>
</tr>
<tr>
<td>scheduling</td>
<td>276</td>
</tr>
<tr>
<td>state management</td>
<td>275</td>
</tr>
<tr>
<td>stratification-based</td>
<td>273</td>
</tr>
<tr>
<td>white box</td>
<td>311</td>
</tr>
<tr>
<td>optional attributes</td>
<td>81</td>
</tr>
<tr>
<td>order policies</td>
<td>242</td>
</tr>
<tr>
<td>by detection time</td>
<td>242</td>
</tr>
<tr>
<td>by occurrence time</td>
<td>242</td>
</tr>
<tr>
<td>by stream position</td>
<td>242</td>
</tr>
<tr>
<td>by user-defined attribute</td>
<td>242</td>
</tr>
<tr>
<td>outer join</td>
<td>202</td>
</tr>
<tr>
<td>output terminal</td>
<td>53, 87, 116</td>
</tr>
<tr>
<td>event types</td>
<td>89</td>
</tr>
<tr>
<td>example</td>
<td>97, 342</td>
</tr>
<tr>
<td>Filtered-in</td>
<td>124</td>
</tr>
<tr>
<td>Filtered-out</td>
<td>124</td>
</tr>
<tr>
<td>identifier</td>
<td>89</td>
</tr>
<tr>
<td>Non-filterable</td>
<td>124, 185</td>
</tr>
<tr>
<td>on Filter agent</td>
<td>124</td>
</tr>
<tr>
<td>specialization and</td>
<td>90</td>
</tr>
<tr>
<td>target</td>
<td>89, 118</td>
</tr>
<tr>
<td>overlapping windows</td>
<td>153</td>
</tr>
<tr>
<td><strong>P</strong></td>
<td></td>
</tr>
<tr>
<td>parallel processing</td>
<td>273</td>
</tr>
<tr>
<td>participant events</td>
<td>218, 226</td>
</tr>
<tr>
<td>partitioning</td>
<td></td>
</tr>
<tr>
<td>optimization</td>
<td>272</td>
</tr>
<tr>
<td>SPADE</td>
<td>272</td>
</tr>
<tr>
<td>partitions, specifications</td>
<td>159</td>
</tr>
<tr>
<td>path expressions</td>
<td>180</td>
</tr>
<tr>
<td>pattern</td>
<td>122</td>
</tr>
<tr>
<td>pattern assertion</td>
<td>221</td>
</tr>
<tr>
<td>pattern context</td>
<td>218</td>
</tr>
<tr>
<td>Pattern detect agent</td>
<td>52, 117, 127, 215</td>
</tr>
<tr>
<td>Assignment not done</td>
<td>245</td>
</tr>
<tr>
<td>Automatic assignment</td>
<td>244</td>
</tr>
<tr>
<td>Consistent strong driver</td>
<td>247</td>
</tr>
<tr>
<td>Consistent weak driver</td>
<td>247</td>
</tr>
<tr>
<td>Delivery alert</td>
<td>245</td>
</tr>
<tr>
<td>example</td>
<td>328–329</td>
</tr>
<tr>
<td>Fast Flower Delivery</td>
<td>347</td>
</tr>
<tr>
<td>Idle driver</td>
<td>246</td>
</tr>
<tr>
<td>Improving driver</td>
<td>247</td>
</tr>
<tr>
<td>Improving note</td>
<td>246</td>
</tr>
<tr>
<td>logical structure</td>
<td>216</td>
</tr>
<tr>
<td>Manual assignment preparation</td>
<td>245</td>
</tr>
<tr>
<td>matching</td>
<td>215</td>
</tr>
<tr>
<td>Permanent weak driver</td>
<td>246</td>
</tr>
<tr>
<td>Pickup alert</td>
<td>245</td>
</tr>
<tr>
<td>Ranking decrease</td>
<td>246</td>
</tr>
<tr>
<td>Ranking increase</td>
<td>246</td>
</tr>
<tr>
<td>pattern</td>
<td>80</td>
</tr>
<tr>
<td>Esper</td>
<td>248</td>
</tr>
<tr>
<td>IBM WebSphere Business Events</td>
<td>249</td>
</tr>
<tr>
<td>Prova</td>
<td>249</td>
</tr>
<tr>
<td>SQL</td>
<td>248</td>
</tr>
<tr>
<td>StreamBase</td>
<td>248</td>
</tr>
<tr>
<td>pattern matching sets</td>
<td>215</td>
</tr>
<tr>
<td>pattern parameters</td>
<td>243–244</td>
</tr>
<tr>
<td>pattern policies</td>
<td>237–242</td>
</tr>
<tr>
<td>cardinality</td>
<td>238</td>
</tr>
<tr>
<td>definition of</td>
<td>217</td>
</tr>
<tr>
<td>evaluation</td>
<td>237</td>
</tr>
<tr>
<td>excess type condition</td>
<td>238</td>
</tr>
<tr>
<td>repeated type</td>
<td>239</td>
</tr>
<tr>
<td>pattern signature</td>
<td>217</td>
</tr>
<tr>
<td>example</td>
<td>218</td>
</tr>
<tr>
<td>pattern type</td>
<td>217</td>
</tr>
<tr>
<td>patterns</td>
<td></td>
</tr>
<tr>
<td>absence</td>
<td>216, 222, 248, 290</td>
</tr>
<tr>
<td>acquisition</td>
<td>311</td>
</tr>
<tr>
<td>all</td>
<td>220, 239</td>
</tr>
<tr>
<td>any</td>
<td>217, 222</td>
</tr>
<tr>
<td>basic</td>
<td>219–228</td>
</tr>
<tr>
<td>detection</td>
<td>13, 19–20</td>
</tr>
<tr>
<td>dimensional</td>
<td>219, 228</td>
</tr>
<tr>
<td>event pattern types</td>
<td>219</td>
</tr>
<tr>
<td>logical operator pattern</td>
<td>220–222</td>
</tr>
<tr>
<td>modal patterns</td>
<td>227</td>
</tr>
<tr>
<td>participant events</td>
<td>215, 218</td>
</tr>
<tr>
<td>pattern matching set</td>
<td>218</td>
</tr>
<tr>
<td>pattern parameters</td>
<td>217</td>
</tr>
<tr>
<td>pattern policy</td>
<td>217</td>
</tr>
<tr>
<td>pattern signature</td>
<td>216</td>
</tr>
<tr>
<td>pattern type</td>
<td>217</td>
</tr>
<tr>
<td>reference table</td>
<td>243–244</td>
</tr>
<tr>
<td>relevant event types list</td>
<td>217</td>
</tr>
<tr>
<td>sequence</td>
<td>248, 289</td>
</tr>
<tr>
<td>subset selection pattern</td>
<td>226</td>
</tr>
<tr>
<td>threshold</td>
<td>217</td>
</tr>
<tr>
<td>threshold patterns</td>
<td>223–226</td>
</tr>
<tr>
<td>trend</td>
<td>291</td>
</tr>
<tr>
<td>payload</td>
<td>63, 70, 84</td>
</tr>
<tr>
<td>collection of attributes</td>
<td>64</td>
</tr>
<tr>
<td>complexity</td>
<td>80</td>
</tr>
<tr>
<td>payload attribute</td>
<td>70–73</td>
</tr>
<tr>
<td>payment devices</td>
<td>93</td>
</tr>
<tr>
<td>Performance evaluation EPA</td>
<td>209</td>
</tr>
<tr>
<td>performance optimizations</td>
<td>269</td>
</tr>
<tr>
<td>composing metrics</td>
<td>271</td>
</tr>
<tr>
<td>latency leveling</td>
<td>270</td>
</tr>
<tr>
<td>max input throughput</td>
<td>270</td>
</tr>
<tr>
<td>max output throughput</td>
<td>270</td>
</tr>
<tr>
<td>min average latency</td>
<td>270</td>
</tr>
<tr>
<td>min maximal latency</td>
<td>270</td>
</tr>
<tr>
<td>real-time constraints</td>
<td>271</td>
</tr>
<tr>
<td>Pickup Confirmation payload attributes</td>
<td>77</td>
</tr>
<tr>
<td>platform-specific</td>
<td></td>
</tr>
<tr>
<td>definition element</td>
<td>49</td>
</tr>
<tr>
<td>runtime artifacts</td>
<td>48</td>
</tr>
<tr>
<td>point in time</td>
<td>228</td>
</tr>
<tr>
<td>polling</td>
<td>95</td>
</tr>
<tr>
<td>polyline</td>
<td>71</td>
</tr>
<tr>
<td>predictive processing</td>
<td>10</td>
</tr>
<tr>
<td>presence</td>
<td>92</td>
</tr>
<tr>
<td>presence detection</td>
<td>93</td>
</tr>
<tr>
<td>probe</td>
<td>92</td>
</tr>
<tr>
<td>procedural programming language</td>
<td>32</td>
</tr>
<tr>
<td>processing logic</td>
<td>14</td>
</tr>
<tr>
<td>processing operations</td>
<td>11</td>
</tr>
<tr>
<td>production rules</td>
<td>43, 260</td>
</tr>
<tr>
<td>OMG</td>
<td>260</td>
</tr>
<tr>
<td>UML</td>
<td>260</td>
</tr>
<tr>
<td>programming entity</td>
<td>5</td>
</tr>
<tr>
<td>programming styles, stream-oriented</td>
<td>256</td>
</tr>
<tr>
<td>Progress Software</td>
<td>25</td>
</tr>
<tr>
<td>Project agent</td>
<td>53, 126</td>
</tr>
<tr>
<td>transformation</td>
<td>189</td>
</tr>
<tr>
<td>projection</td>
<td>45</td>
</tr>
<tr>
<td>Split agent</td>
<td>195</td>
</tr>
<tr>
<td>proxies</td>
<td>42</td>
</tr>
<tr>
<td>publish/subscribe</td>
<td>37, 44, 54, 56</td>
</tr>
<tr>
<td>pull-style</td>
<td></td>
</tr>
<tr>
<td>distribution</td>
<td>55</td>
</tr>
<tr>
<td>distribution scenarios</td>
<td>38</td>
</tr>
<tr>
<td>event distribution</td>
<td>38, 111</td>
</tr>
<tr>
<td>push-style</td>
<td></td>
</tr>
<tr>
<td>distribution</td>
<td>35–36, 55</td>
</tr>
<tr>
<td>event distribution</td>
<td>35, 37, 111</td>
</tr>
<tr>
<td>event interactions</td>
<td>35</td>
</tr>
</tbody>
</table>
INDEX

Q

queries 31
   continuous 274
   optimization 274
query capability 95

R

radio-frequency identification 7, 88, 92
random m filter 186
sampling 187
ranking and reporting system EPA table 132
rate limiting 187
real-world events 4
real-world focus 11–13
recoverability 267
reference data 191
   Enrich agent 193
relational algebra 52, 80, 126
relevant event type 218
reliability 16
remote procedure call 32
repeated type policies every 239
   first 239
   last 239
   override 239, 349
   with maximal value 239
   with minimal value 239
representational state transfer 32
Request enrichment EPA 192
request-response 31–32
   decoupling 34
   for event distribution 37
resources 33
   in management applications 96
   polling 34
response 31, 34
REST-style web services 32
retraction 75, 299
   example of 76
   hardcoding 300
retrospective compensation 294
retrospective processing 295
reuse
   event processing network models 116
   software 39
reverse geocoding 205

RFID. See radio-frequency identification
rolling average 46
room occupancy 92
routing 16, 44, 56, 80
routing schemes 136
RPC. See remote procedure call
RSS. See Really Simple Syndication
ruleCore 80, 82
delivery request 82
filtering 211
rule-oriented languages 260
production rules 260
runtime artifacts 119
centralized implementation 120
functional assignment 120
geographical location 121
parallel execution 120
platform and language issues 119
segmentation context 120
sliding event interval 187, 248
context definition 153
context examples 154, 172, 339
context parameters 153
event list 153
event period 153
interval size 153
temporal ordering 153
sliding fixed interval 187, 203
context 152
overlapping and non-overlapping 153
Smalltalk 7
SOA. See service-oriented architecture
SOAP-style web services 32
social communications 94, 108
software 92
   event consumers 108
   event producer as application 92
feed reader 108
sometimes pattern 228
spatial context 51, 71, 154–160
   contained in 156
   contains 157
disjoint 157
   entity distance location 158
   equals 157
   event distance location 159
   fixed location 155
   location types 158
   overlaps 157
touches 157
spatial patterns 232–235
   average distance 233
   max distance 233
   min distance 232
   relative average distance 234
   relative max distance 234
   relative min distance 234
spatiotemporal patterns 235–236
moving in a constant direction 235
moving in a mixed direction 236
moving toward 236
stationary 236
specialization 74
Split agent 52, 126, 194
filtering 195
iterative splitting 195
output terminals 194
specifying 195

S

sampling, stream of events 186
scalability 265–266
security 16, 268
segmentation-oriented context 51, 161–162
definition of 161
   example 168, 336
   grouping key 174
   partition expressions 162
   partition identifier 162
   semantic role 77
   sensors 91, 94
   embedded 91
   emitting an event 94
   sentinel value 81
   sequence diagrams 32
   sequence number 149
   sequence pattern 228
   service providers 31
service-oriented architecture 17, 31
   composite service 40
   event consumers and event-based programming and
   40
   request-response and 39–40
   simulated sensors 92
   situation, definition 7

INDEX

INDEX
INDEX

Split agent (continued)
splitting composite events 196
static splitting 195
vs. Translate 195
splitting events 45
composite events 196
static 195
SQL 25
standardization
domain-specific metadata 305
EPA assembly model 306
EPA component model 306
EPL 306
event distribution standards 306
event processing 306
event structure 305
state machine 109
state management optimization 275
stateful
agent 45, 138
Aggregate agent 196
Compose agent 196, 199
EPLs 267
event processing 119
event processing and context 145
filtering 186
processing languages 46
stateless
definition 44
Transformation agents 188
state-oriented context 51, 160–161
static analysis 203
stationary pattern 236
Store event consumer 112
Store event producer 96, 341
stream computing 21
stream processing, Aggregate agent 196
StreamBase 26, 140
filtering 210
stream-oriented languages 43
operator example 259
SPADE 259
stream-oriented programming 256
Aleri 257, 259
CCL 259
CQL 258–259
SPADE 257
windows 257
streams 44, 62
Compose agent 199
Filter agent 184
left stream 199
output terminals and 117
right stream 199
stateful event processing and 45
Transformation agents and 188
windows 46
StreamSQL 140
StreamSQL EventFlow 26
subscriber, definition of 110
subscribing 110
subscriptions 37
subset selection patterns 226
relative n highest values 226
relative n lowest values 226
surveillance 94
symptoms 20
synchronous interactions 31
temporal anomalies 288
mitigation 292
out of order 293
temporal context 51, 147–154, 222
call center example 167
context partitions 148
definition of 147
derived event 284
detection time 284
dominant context dimension 284
EPAs and 148
event ordering 284
issues with derived events 286
occurrence time 284
order of events 284
segmentation with Amit 174
semantics 284–295
time interval 284, 286
time point 284
temporal granularity 67, 204
coarse 285
definition of 66
temporal order
buffer technique 293
distributed environments 292
event order 291
order of derived events 287
out-of-order event 294
parameter 149
retrospective compensation 294
sequence assignment 294
temporal patterns 228
first n 229
last n 230
sequence 228
threshold assertion 225
threshold patterns
count 223
functor 225
threshold assertion 223
value average 225
value max 224
value min 225
time granularity 284
time intervals
boundaries defined 149
definition 285
explicit representation 285
fixed interval context 150
non-repeating 150
periodically repeating 150
relationships 286
time point 285
time server 292
timeout
buffering technique 293
detection 348
timestamps 62
definition of 66
derived events and 287
temporal dimension 284
XPath 183
top-down
approach to modeling 47
event processing application
design 141
training simulators 4
transformation 16, 19, 187–205
example from an EPL 211
languages used 190
Transformation agent 52, 125
classification of 187
internal logic 188
Transformation agent header
attributes table 204
Translate agent 52, 125
Location service EPA 205
transformation 189
vs. Split 195
translating 45
transport 20
trend patterns 230–232
decreasing 230
increasing 230
mixed 231
non-decreasing 231
non-increasing 231
stable 230
tuples 80
Twitter 108
two-way join 202

U
certain events 68
unsubscribing 110
untyped events 62
updates 31

V
validation
actual reachability issues 280
actual termination issues 279
dynamic analysis 279
EPA is not reachable 278
event not used 278
output terminal is unusable 280
possible nondeterministic processing 278
runtime validation 203
static analysis 277
termination problem observation 277
value average pattern 225
value max pattern 224
value min pattern 225
Vehicle event producer 98, 342
verification devices 93
virtual platforms 310
cloud computing 310
embedded 310
hardware appliance 310
messaging 310
stream 310
virtual worlds 4

W
web browsers 31
Web Services Business Process Execution Language 17
Web Services Distributed Management 84
WebSphere Business Events 81
active rule 262
windows 46
event interval context 150
sliding 46
stream 257
workflow 98
WS-BPEL. See Web Services Business Process Execution Language
WSDM Event Format 84
WS-EventDescriptions 84
WS-Topics 84

X
XML 190
filtering 179
node-set 179
See also Extensible Markup Language
XML Schema 81
delivery request 83
XPath 190
attribute comparison with 182
built-in functions 182
expressions 181
filter expressions 181
testing and 180
testing XML node 181
time functions 183
XSLT 190
filtering 179
splitting and 196
transformation 191
See also Extensible Stylesheet Language Transformations

Z
zero duration 284