Presenter

- **Presenter:**
  Paul Vincent, CTO Business Rules and CEP, TIBCO Software
  - Member OMG PRR and W3C RIF rules standards bodies

- **TIBCO Software Inc.:**
  - Provides enterprise software that helps companies achieve service-oriented architecture (SOA) and business process management (BPM) success
  - Headquartered in Palo Alto, California
  - Over 3,000 customers and offices in 40 countries
  - CEP product is TIBCO BusinessEvents
    - Developed from a customer solution and launched 2005
    - Currently at Release 3.0
- Introducing CEP
Customer Logon
Fed Base Rate Increase
Production Item Arrives at Store
Customer Checks Close Account Web Page
Mobile Call from CT @11.13
New Liability Added
New Order
Contract Submitted
Contract Returned thru EDI
Rental Car Returned
Rental Car Crashed
Fraud Risk!
Risk of Customer Defection
Employee Over hours
Compliance Limit Approached
Rental Contract Complete
Customer CrossSell Opportunity
Change in Product Sales Trend
Cell phone fraud alert
Customer now rated Gold
Contract Validated
Contract Valid
Conventional Event Processing

(Lots of Events)

Event-at-a-time

Synchronous Events

Simple event processing

Aggregation as data

Business Processes: BPM & Workflow Processes

IT Services: SOA & traditional Data Processing

Persistence Services

New Order

New Order (Lots of Events)

Synchronous Synchronous Synchronous Synchronous Events Events Events Events

Event-at-a-time

Simple event processing

Aggregation as data
Simple EP = default IT Model, 1950-now

- Based on “human workflow”: one thing at a time
  - Processes handle cases 1 at a time ← office clerk
  - Use database and refer to it where necessary ← card index
  - Provide some service flexibility with middleware ← internal mail
  - Use BPM to document / manage / automate processes
  - Use SOA to distribute / manage / automate services

This model does NOT exploit
ALL the information / data / events
ALL the time

Behaviour (and business logic) is silo’d
There is a better way!
Complex Event Processing

- Asynchronous Events

(Lots of Events)

Continuous Event Processing

Complex Patterns of Events

- Event Store
  - High performance pattern-matching via rules, states, & queries
  - High performance persistence

Business Processes:
BPM & Workflow Processes

IT Services:
SOA & traditional Data Processing

Persistence Services

(Lots of Events)

New Order
New Order
Event Store
Complex Patterns of Events
Asynchronous Events
High performance pattern-matching via rules, states, & queries
High performance persistence
CEP Terminology

- CEP (technology) applies pattern detection & filtering to the event clouds & streams and their histories
- Multiple modelling / execution paradigms are available for pattern detection
What does CEP cover?

“CEP applies to a very broad spectrum of challenges in information systems. A short list includes:”

- Business process automation
- Computer systems to automate scheduling and control network-based processes and processing
- Identifying when complex contracts are fulfilled
- Detection intrusion, fraud and other network attacks
- C3I


David Luckham
What does CEP Solve?

**CEP raison d’être: quickly respond to detected variances**

**Figure 1:** The steps involved in taking action to respond to business events

*the “Latency Problem”*

usually: warnings precede threats

Value of action

Potential business value

Time

Action time

Business event

Data captured

Insight delivered

Decision made

Action taken

Data latency

Insight latency

Decision latency

Action latency
What CEP provides

“Situational Awareness”

“Sense and Respond”

“Track and Trace”
History
Command and Control

- Correlate all available information
- Determine tactics based on strategy and up-to-date information

-- from RAF Battle of Britain Fighter Control System 1940
http://www.raf.mod.uk
Data Fusion

EVENT SOURCES

EXTERNAL

DISTRIBUTED

LOCAL

- Sensors
- Documents
- People

Data Sources

Level 0 Processing
Sub-Object Assessment

Level 1 Processing
Object Assessment

Level 2 Processing
Situation Assessment

Level 3 Processing
Impact Assessment

Level 4 Processing
Process Refinement

Database Management System
Support Database
Fusion Database

Human/Computer Interface

-- Revised JDL data fusion model, 1998
Condition Based Maintenance

Events and CEP
Complex Business Problems

- **Fraud / Theft**
  - Thousands-to-millions of high-value small-size product items or transactions
  - How do you identify known patterns of “suspicious” behavior?

- **Logistics / Scheduling**
  - Raw material, production & delivery scheduling and resources are complex and prone to change
  - How do we reallocate resources to handle business and production changes?

- **Activity Monitoring**
  - Complex production and supply process with multiple actors
  - How to measure and action Key Performance Indicators?
Associated Events

- **Positive Events**
  - Product item X arrives at Production station S from Store T
  - Production worker Y arrives at Production station S
  - Production contract for item Z by time T is posted

- **Negative Events**
  - Product item X has been in transit to Store T for >15 minutes
  - Subcomponent Y hasn’t arrived at the Production station by the ETA
  - Delivery of contract Z has not taken place

- **Sets of Events**
  - 5+ items of Product item type Y failed to arrive at destination
  - Supplier Y was 5 mins late for 1 delivery, but made it early to the next
  - Return rate on component Z exceeds SLA %
Significant features of these Events

- **Time Sensitivity**
  - A thief may leave the building at the same time as stolen product
  - A product should take 40 minutes to travel a given production line segment

- **Distributed Event Sources**
  - A series of produced items fails at various QA stages, and their common attribute was a storage location
  - Multiple suppliers for a subcomponent are reporting delivery delays
What *is* an “event”?

- **Change of state in some entity**
  - Customer call
  - Bank debit
  - Aircraft movement

- **Observation of some entity**
  - CRM record of a customer call
  - ATM report of debit transaction success
  - Radar plot update of an aircraft

- **IT Message**
  - Queued point-to-point message
  - Publish / subscribe message
Sample Event Metamodel

- Event contains data + timestamp
- Various classifiers (int/ext, transport, lifecycle, …)

Channel

Destination

| Default Destination |

Simple Event

General Event

Inherits from

| Timestamp |
| TimeToLive |
| ExpiryAction |

Custom Attributes

XML
Event-driven vs Event Processing

Event Sources → Event Bus

Simple Event Processing
- Business Process Management
- Services
- Data Services

Invoke / single event

Complex Event Processing
- Event Processing Agents
- Event Store

Inform / multiple events
Event Driven Architecture

S O A

EDA

Assumptions...

Request - Reply

Transmit - Listen

Synchronous Events

Asynchronous Events
CEP in the Event Driven Architecture

SOA

EDA

Continuous Event Monitoring

Complex Event Processing

Synchronous Events

Asynchronous Events

Assumptions...
Complex Event Processing

Sense and Respond / Track and Trace / Situational Awareness

Processing type:
- Decision Processing
- Event Processing

TIBCO Reference Architecture

Impact Analysis
Situation Identification
Filtration
Events and Information

Flexible Service-Oriented Architecture

Underlying Applications and Infrastructure
Why CEP?

- Detecting event patterns across multiple event types + time is difficult for simple event processing solutions
- Computers can correlate across large volumes of events at high speed, identifying patterns that are not conventionally visible
- The architecture pattern of “continuous event processing” applies to many business domains such as BAM
- Examples in use:
  - **Track and Trace** of RFID data
  - **Situation Assessment** of airline operational delays (+ their causal events)
  - **Sense and Respond** to fraud indicators in internet transactions
Agenda

- CEP Technologies
CEP = an Event-Decision Architecture

Proposed for EPTS Reference Architecture

Adapted from JDL Data Fusion Model by Tim Bass
Requirements for CEP Technology

- **Access and Monitor the “Event Cloud”**
  - JMS, RV, MQ, TCP/IP, etc…
  - Timers to detect lack of events
  - Determine event state changes

- **Match Patterns, Apply Business Logic**
  - Detect events
  - Detect event patterns
  - Maintain State and Facts over time
  - Update Detection algorithms as events change
CEP-Related Standards

- **Event Bus or Source**
  - Low-latency reliable message delivery

- **Event and Data Model**
  - for information modeling

- **Temporal Model**
  - for determining time-dependent info

- **Pattern Detection Model**
  - for recognizing patterns, defining actions

- **History and Cache**

- **JMS, (DDS)**

- **OMG EMP**
  - (RFP in progress)

- **OMG UML2 Class Models**

- **OMG UML2 State Models**

- **OMG PRR, W3C RIF Production Rules**

- **ANSI SQL-based continuous queries**

* = in devt
Example CEP Technologies

Event Services

Types of CEP Processing

Rule-Driven CEP
- Inference
- Rule Engine
- Rulebase

Neural Net CEP
- Neural Net
- Network

Event Stream Query CEP
- Query Engine
- Stream Queries

Orchestrated Event Stream CEP
- Flow Engine
- Flow Model
Sample Event Processing Metamodel

General Behavior

Event-driven Behavior

Inherits from

Input

Output

Event

Data

External

Internal

Timer

State Chg

Event TimeOut

Rule Process

State Manager

Query Process

Procedural Process

Inherits from
CEP Behavior: State-oriented

Event Bus or Source
Low-latency reliable message delivery

Event and Data Model
for information modeling

Temporal Model
for determining time-dependent info

Pattern Detection Model
for recognizing patterns, defining actions

History and Cache

Condition / TimeEvent

State Model

State 1 → State 2

New Order

New Order

New Order
State Model / Process Flow CEP Agent features

1. Visual modeling metaphor
   - State diagram / flow diagram is simple to follow

2. State / flow transitions can be time-related
   - Can model missing events through time-outs etc
CEP Behavior: Rule-oriented

- Event Bus or Source
  - Low-latency reliable message delivery
- Event and Data Model
  - for information modeling
- Temporal Model
  - for determining time-dependent info
- Pattern Detection Model
  - for recognizing patterns, defining actions
- History and Cache

(Lots of Events)

Complex Patterns

Rule

Reaction
Inference Rule CEP Agent features

1. High performance pattern matching
   - Rete algorithm determines rules that are executable based on underlying data changes

2. Declarative + Inferencing
   - Rules defined in terms of classes: can be relevant for any # instances
   - Rules’ actions can cause other rules to fire automatically

3. In-memory
   - Limited only by JVM / process memory
CEP Behavior: Query-oriented

- Event Bus or Source
  - Low-latency reliable message delivery
- Event and Data Model
  - for information modeling
- Temporal Model
  - for determining time-dependent info
- Pattern Detection Model
  - for recognizing patterns, defining actions
- History and Cache

(Lots of Events) → Complex Query → Results Event

Query
Query CEP Agent features

1. Common query language
   - Usually SQL-based – widely used language
   - May be in-memory, in-file or both
   - Can include query optimizers

2. Continuous
   - Extensions usually support time windows for the query to operate over
Examples
## Typical Business Situations for CEP

<table>
<thead>
<tr>
<th>Detected Business Situation</th>
<th>Resulting Situation-Decision</th>
</tr>
</thead>
<tbody>
<tr>
<td>User X is behaving suspiciously (high likelihood of fraud)</td>
<td>Investigate for fraud manually</td>
</tr>
<tr>
<td>Subcomponent delivery Y is slightly late</td>
<td>Issued an automated reminder to supplier</td>
</tr>
<tr>
<td>Customer Y payment for policy P is very late</td>
<td>Alert Customer Agent</td>
</tr>
<tr>
<td>Orders for product Z are up &gt;20%</td>
<td>Alert manufacturing and marketing</td>
</tr>
</tbody>
</table>
CEP in Action: Investment Banking

Algorithmic Trading CEP Application

- Query Engine
- Querybase

Compliance CEP Application

- State Engine
- State Model
- Inference Rule Engine
- Rulebase

Front Office
Middle Office
Back Office

Data Feeds (events)
SOA Services
Archival etc Database
CEP Processing

1. Order event arrives over JMS / EMS or RV on a particular channel
2. Order event classified into appropriate class, stored in cache for future reference
3. Order state assessed and updated due to event details
4. Comparison rules run against order by rule engine, new information inferred
5. State transition rules executed to check for any state change due to new inferred data.
6. If order can be completed, run FSA compliance rules
Example CEP Product Architecture

CEP engine(s)

Event Preprocessor

Rule Agent(s)
- Rulesets
- State Models
- Functions
- Rete Working Memory
- Conceptual Model
- Events
- Queries
- Distributed cache of events etc

Query Agent(s)

Cache Node(s)
- Backup DB

Channels

Destinations

ESB / Message Bus / Event Bus
“Advanced” CEP
“Advanced” CEP defined in many ways

- **Intelligent CEP**
  - Adaptive
  - Learning
  - Logic
  - AI

- **Semantic CEP**
  - Ontologies + Logic
  - Text / language interpretation

- **Multiple CEP**
  - Including all types of data processing paradigm
    (transactional, CLP, inference, mathematical methods, …)
Advanced CEP Infrastructure

Event Sources

Event Consumers

Event Bus / Store

Event Processing CEP Agents
- State Engine
- Inference Rule Engine
- Streaming Query Engine

State Model
Rulebase
Queries

BI / Analytics tools
- Statistical Pattern Detection
- Graphical Pattern Detection

Optimization & Machine Learning
- CLP Engine
- Constraints
- Ontologies, Logic
- MetaRules

Dashboard
- Graphical Reports

BPM / SOA

Business User Interface
- CEP Editors

Graphical Pattern Detection

Graphical Pattern Detection
Advanced Patterns & Event Behaviors

- Many EP apps fit the standard CEP patterns:
  - Filter interesting rules
  - Detect predefined patterns / state changes
  - Update data / invoke processes and services based on business rules and high level events

- Advanced EP:
  - Apply interesting statistical functions to event data to detect new / complex trends
  - Apply different algorithms to event data
  - Modify parameters used in other rules ("metarules")
“Event-Decision” Architecture

Event-Decision Architecture

- **Event Tracking & Refinement**
- **Situation Detection & Refinement**
- **Impact Assessment & Predictive Analysis**
- **Process Refinement & Adaptation**

**Event Pre-Processing**

**Level One**
- Event Tracking & Refinement

**Level Two**
- Situation Detection & Refinement

**Level Three**
- Impact Assessment & Predictive Analysis

**Level Four**
- Process Refinement & Adaptation

**DB Management**
- Historical Data
- Profiles & Patterns

- **User Interface & Systems**

--- Adapted from JDL
What are the variables that can be adjusted in real-time to optimize system performance?
If RFID event for product X
Then
  monitor, else ignore

Becomes

If RFID event for product in list Y where cost > Z
Then monitor, else ignore

Updated by

If average loss increase for all products in Y > 2%
Then reduce Z by 5%
If drug class X and dose > 200ml
Then
move to monitored drug state

Becomes

If drug class X and dose > Y ml
Then
move to monitored drug state

If clinical negative events for drug class X increase
Then reduce Z by 10ml
If bag X is not on prescribed flight at (depart - 20)
Then
move X state to MissedFlight

Becomes
If bag X is not on prescribed flight at lastBagTime
Then
move X state to MissedFlight

Updated by

If flight NOT international
Then set lastBagTime to carrier’s min( DoorCloseTime)
If  
product.ShipDelay  
> 1 days  
Then  
contactLegal  

Becomes

If  
product.ShipDelay  
> contract.SLA  
MaxDelay - AllowedShipLag  
Then  
warnLegal

Updated by

If  
contract.customer.status = Hi  
Then set AllowedShipLag to 2 days
Issues

- **Needs constraints**
  - Eg Cannot reduce discount to <0 or increase above>25
  - Can handle as “change events” and rules to test…

- **Difficult to test**
  - May be based on statistical functions – implies complex test regimes (or test-specific rules)

- **Complex to prove ROI / value**
  - End-user may not be able to source or validate the advanced rules

- **Requires statistical function libraries / analytics**
Other sources for “advanced rules”

- Uncertainty
  - Scoring

- Generating rules
  - Machine learning
  - Predictive Analytics
  - Reasoning + Ontologies

- Other types of rules
  - Constraint Logic Programming
Scoring

- Simple technique to handle “variable” decisions
- Rules update a score
- Example: insurance scoring
- Typically handled in a special ruleset (or decision table)
- Good as a KPI in a scorecard
- Typically used with an aggregation rule

<table>
<thead>
<tr>
<th>Object Property</th>
<th>Condition</th>
<th>Score Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>&lt;18</td>
<td>-10</td>
</tr>
<tr>
<td>Age</td>
<td>19 to 26</td>
<td>-15</td>
</tr>
<tr>
<td>Age</td>
<td>27 to 49</td>
<td>+5</td>
</tr>
<tr>
<td>Age</td>
<td>50 to 69</td>
<td>0</td>
</tr>
<tr>
<td>Age</td>
<td>70+</td>
<td>-5</td>
</tr>
</tbody>
</table>
Machine Learning

- Given a set of data, deduce classification patterns and hence rules
- Requires sophisticated algorithms

<table>
<thead>
<tr>
<th>data</th>
<th>result</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
</tr>
<tr>
<td>other</td>
<td>other</td>
</tr>
</tbody>
</table>
Predictive Analytics

- Analyze data to deduce segmentation breaks for tasks like customer classifications
  - Eg: Which customers should be offered what interest rate to maximise profit?

- Typically using specialist data mining tools
  - Exports decision tree, rules etc in varieties of PMML

- Overlaps with BI (eg custom reports on historic data)

- Analytical functions may also be mapped to a ruleset in CEP for real-time analytics
Reasoning + Ontologies

- “Semantic Event Processing”
- Use Semantic Web technologies to augment CEP
  - Textual news etc analysis
  - Use of deeper ontology relationships
- Example components
  - OMG Ontology Definition Metamodel ODM joins W3C OWL to UML concept models
  - OWL, RDF, RDFS for terminology, relationships
  - Logic languages/rules to reason about truth over event types and metadata
Example: Semantic Technology to Refine CEP
Semantic Networking Event Monitoring Architecture

Notifications of Anomalies, Course Corrections, Instructions for Retraining & Recharacterization, Actions, Queries

Sense events → Assess conditions → take action

CEP Engine + Production Rules
CEP + Production Rule reasoning

Event Source / Bus

-- courtesy of Sandpiper Software

• Features include:
  • Production rule reasoning can
    • report up (situation assessment & current course of action)
    • report laterally (situation assessment, & sensed changes, etc.)
    • report down (initiating actions, querying)
  • Adaptive capabilities are possible at all levels

Pattern & Trend KB

SPARQL/Pellet/OWL + Description Logics Reasoning
+ Probabilistic & Special Purpose Reasoning

Longer-Term Reasoning
Trend Analysis, Contextual Learning

Event KB

SPARQL/Pellet/OWL + Description Logics Reasoning
+ Production Rule Reasoning

Event Classification & Aggregation

Operational Decisions
Constraint Logic Programming

- **Constraint rules for systems**
- **Constraint solver to find best values** (e.g., optimize price)
  - With response time as a system constraint!
  - Goal-driven
- **Uses:**
  - Maximizing value of inventory
  - Scheduling the best routes for trucks
  - Maximizing probability for SLA achievement
The End
Agenda

- Q & A
A. Appendices & Back-up Information
Appendix: Useful web resources

- Event Processing Technical Society EPTS
  www.ep-ts.com

- Luckham’s web site
  complexevents.com

- Various vendor blogs (reference from complexevents.com)