Authoring and deploying business policies dynamically for compliance monitoring

Yurdaer Doganata, Keith Grueneberg, John Karat, Nirmal Mukhi

IBM T. J. Watson Research Center
Hawthorne, NY 10532 USA

June 7, 2011
Automate policy authoring, deployment and compliance monitoring
Architecture

- Architecture model
- Implementation of the Business Process Model
- Event Data Capturer
- Provenance recorder
- Semantic mapping
- Policy Authoring Tool
- Policy deployment
- Provenance Store
- Compliance Browser & Root cause analysis

Relations:
- Architecture model to Implementation of the Business Process Model
- Event Data Capturer to Provenance recorder
- Semantic mapping to Provenance Store
- Policy Authoring Tool to Policy deployment
- IT Terminology
- Business Ontology
- Business Vocabulary
- Provenance data
- Business rules
- Graph
Architecture

- Architecture model
- Implementation of the Business Process Model
- Event Data Capturer
- Provenance recorder
- Semantic mapping
- Policy Authoring Tool
- Policy deployment
- Provenance Store
- Provenance data
- Compliance Browser & Root cause analysis
- Business Ontology
- IT Terminology
- Business Vocabulary
- Business rules
The dimensions of business provenance

- **Who**
  - resource as actor
- **Did What,**
  - task execution,
  - data manipulation
- **When**
  - process instance
- **and Why?**
  - goal drives process
Architecture

- Architecture model
- Implementation of the Business Process Model
- Event Data Capturer
- Provenance recorder
- Semantic mapping
- Policy Authoring Tool
- Provenance Store
- Compliance Browser & Root cause analysis
- Policy deployment
- Business rules
- Business Vocabulary
- IT Terminology
- Architecture model
- Business Ontology
- Provenance data
- Compliance Browser & Root cause analysis
Rule authoring is a challenge

- Writing and deploying rules with IT Level data requires data management and programming skills
- Business people need tools to author rules in the language that they understand

Example Rule:

```
<trigger>name()='processItem' && $item.name contains 'Challenge' </trigger>

<action><ae:createItem>
    <ae:condition>
        <ae:select-none name='$item.name' elementType='item' elementSubType='financialKeyControl'/>
    </ae:condition>
    <ae:template inputElement='item' copyAll='true'
    <prov:item name='$item.name' subType='financialKeyControl' />
</ae:template>
</ae:createItem></action>
```
Architecture

- Architecture model
- Implementation of the Business Process Model
- Event Data Capturer
- Provenance recorder
- Provenance Store
- Compliance Browser & Root cause analysis

- Semantic mapping
- Policy Authoring Tool
- Policy deployment

- Business Ontology
- IT Terminology
- Provenance data
- Business rules
Using templates to simplify rule creation

- **BUSINESS RULE in NATURAL LANGUAGE**
  - If the **car value** is **less than** the **initial damage estimation**, then the **car should be totaled** by the **claim handler**

- **TEMPLATE**
  - If {**DATA**} {**OPERATION**} {**DATA**}, then {**RESOURCE**} {**OBLIGATION**} {**TASK**}

- **INSTANTIATIONS**
  - If {**car.value**} {**less than**} {**initial damage estimation**}, then {**claim handler**} {**should**} {**total the car**}
  - If {**car.value**} {**more than**} {**initial damage estimation**}, then {**claim handler**} {**should**} {**not total the car**}
Semantic Mapping: Linking business vocabulary to IT terminology

Business Vocabulary

Repair cost must not be more than the “car value”

IT Terminology

- accidentReport: AccidentReportType
- additionalAuthorizationRequest: DamagesAreasType
- additionalWorkAuthorization: DamagesAreasType
- amendedInvoice: AmendedInvoiceType
- autoReportShop: NameType
- claimAdjuster: NameType
- claimDoc: ClaimDocType
- claimHandler: NameType
- car: NameType
- decision: Total
- deductible: PaymentType
- invoice: PaymentType
- paymentRepairCost: PaymentType
- paymentTotal: PaymentType
- policyOwner: NameType
- updatedDamageEstimation: DamageEstimationType
Semantic Mapping for “car value”

Schema for IT level data

XML file for semantic mapping

```xml
<v2:typeSemantic>
  <v2:typeRef/>cla:ClaimDocType</v2:typeRef>
  <v2:path>/car/value</v2:path>
  <v2:semantics>
    car value
  </v2:semantics>
</v2:typeSemantic>
```
Policy Authoring Tool
Welcome to the Policy Management Environment!

To get started authoring policies, select the Author link on the left Navigator.
Select an attribute to edit values or click "New" to add an attribute:  Now  Import  Delete All

Data
- Claim document (Data.ClaimDocument)
- Police report (Data.PoliceReport)
- Invoice (Data.Invoice)
- Amended Invoice (Data.AmendedInvoice)
- Initial authorization to repair (Data.InitialAuthorizationToRepair)
- Additional authorization to repair (Data.AdditionalAuthorizationToRepair)
- Initial damage estimation (Data.InitialDamageEstimation)
- Updated damage estimation (Data.UpdatedDamageEstimation)
- Decision To Total (Data.DecisionToTotal)

Resource
- Claim Resource (Resource.ClaimResource)

Task
- Claim Task (Task.ClaimTask)
Select an attribute to edit values or click "New" to add an attribute: New Import Delete All

Data

- Claim document (Data.Claimdocument)
  - Car: Make
  - Car: Model
  - Car: Value
  - Car: Vehicle Identification Number
  - Car: Year
  - Claim ID
  - Policy number
  - Policy owner

- Police report (Data.Policereport)
  - Accident Location: City
  - Accident Location: State
  - Accident Location: Street
  - Accident Location: Zip Code
  - Time of the accident
Select or create a new policy set:

Maritopia Policy Set

Policy Templates

| All | Claim Handler (0) |


Select All Add Remove Edit Update

| Name | Description |

June 5 18, 2011
Claim handler must determine if car should be totaled and should decide to total if Car: Value is less than initial repair cost estimate OR updated repair cost estimate.
“Each claim must have an initial damage estimation”
Deploying the rules in Provenance Graph
Dashboard before the rule is deployed
### Claim Processes

<table>
<thead>
<tr>
<th>Process</th>
<th>KCP1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClaimProcess 10001</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10002</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10000</td>
<td>Defect</td>
</tr>
<tr>
<td>ClaimProcess 10005</td>
<td>Defect</td>
</tr>
<tr>
<td>ClaimProcess 10006</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10003</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10004</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10009</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10007</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10008</td>
<td>Defect</td>
</tr>
</tbody>
</table>

**KCP1:** Each claim must have an initial damage estimate.
Maricopa Compliance Monitoring - Auto Insurance Scenario

### Claim Processes

<table>
<thead>
<tr>
<th>Process</th>
<th>KCP1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClaimProcess 10001</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10002</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10000</td>
<td>Defect</td>
</tr>
<tr>
<td>ClaimProcess 10005</td>
<td>Defect</td>
</tr>
<tr>
<td>ClaimProcess 10006</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10003</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10004</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10009</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10007</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10008</td>
<td>Defect</td>
</tr>
</tbody>
</table>

KCP1: Each claim must have an initial damage estimate.
Dashboard after the rule is deployed
## Maricopa Compliance Monitoring - Auto Insurance Scenario

### Claim Processes

<table>
<thead>
<tr>
<th>Process</th>
<th>KCP1</th>
<th>KCP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClaimProcess 10001</td>
<td>Compliant</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10002</td>
<td>Defect</td>
<td>Warning</td>
</tr>
<tr>
<td>ClaimProcess 10000</td>
<td>Defect</td>
<td>Defect</td>
</tr>
<tr>
<td>ClaimProcess 10005</td>
<td>Compliant</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10006</td>
<td>Defect</td>
<td>Warning</td>
</tr>
<tr>
<td>ClaimProcess 10003</td>
<td>Compliant</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10004</td>
<td>Compliant</td>
<td>Warning</td>
</tr>
<tr>
<td>ClaimProcess 10009</td>
<td>Compliant</td>
<td>Compliant</td>
</tr>
<tr>
<td>ClaimProcess 10007</td>
<td>Compliant</td>
<td>Warning</td>
</tr>
<tr>
<td>ClaimProcess 10008</td>
<td>Compliant</td>
<td>Warning</td>
</tr>
</tbody>
</table>

**KCP1**: Each claim must have an initial damage estimate.

**KCP2**: Claim handler must determine if car should be totaled and should decide to total if Car: Value is less than initial repair cost estimate OR updated repair cost estimate.
Claim handler must determine if car should be totaled and should decide to total if Car: Value is less than initial repair cost estimate OR updated repair cost estimate.
Conclusions

- Demonstrated the integration of policy authoring tool to business provenance management system

- Semantic mapping of the IT level data to business language is essential but not automated

- Using templates in policy authoring help to avoid natural language processing

- Mapping the rules from authoring tool to provenance graph require automation
Questions?
Data

- Claim document (Data.Claimdocument)
  - Car: Make
  - Car: Model
  - Car: Value
  - Car: Vehicle Identification Number
  - Car: Year
  - Claim ID
  - Policy number
  - Policy owner

Add, Delete Selected, Delete All