IBM Operational Decision Manager

Subhajit Maitra
zChampion
November 2013
<table>
<thead>
<tr>
<th>Agenda</th>
</tr>
</thead>
<tbody>
<tr>
<td>► What is Decision Management?</td>
</tr>
<tr>
<td>► Introducing IBM Operational Decision Manager</td>
</tr>
<tr>
<td>► Business Decisions Defined</td>
</tr>
<tr>
<td>► Components, Architecture and Technical Dive</td>
</tr>
<tr>
<td>► ODM on z/OS</td>
</tr>
<tr>
<td>Agenda</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>► What is Decision Management?</td>
</tr>
<tr>
<td>► Introducing IBM Operational Decision Manager</td>
</tr>
<tr>
<td>► Business Decisions Defined</td>
</tr>
<tr>
<td>► Components, Architecture and Technical Dive</td>
</tr>
<tr>
<td>► ODM on z/OS</td>
</tr>
</tbody>
</table>
Operational Decision Management Transforms Business Outcomes

Providing an easily manageable, single source of truth for operational business decisions

**Codifies business policies, practices and regulations**

**Enables changes to be easily made by business people**

**Automates decision making with the fidelity of an expert**
Business Rules in organizations

Can get quite large

Are scattered everywhere

Changes are costly, resource & time-intensive

- Rules are hidden in COBOL code
- Most changes have to be programmed – costly

Lack of consistency

- No central management
- No reuse of decision logic

Gap between business analysts & IT administrators

- Knowledge fades over time

Lack of audit ability

No easy way to test/simulate changes
Operational Decision Management approach

Operational Decision Manager Platform

- **Decision logic is defined, analyzed and maintained**
  - User Tools
- **Decision logic is stored and shared**
  - Repository
- **Decision logic is deployed, executed and monitored**
  - Execution Runtime

Manual Processes

- Online Transactions
- Batch / COBOL
- Spreadsheets
- Databases
- People
Externalizing Decisions from Applications into Business Rules

Manage decision logic independently from applications

**Without Decision Management**

- Rules written in software code cannot be read by business people
- Hard coded rules are difficult to change
- Rules intertwined within applications cannot be reused by other systems

**With Decision Management**

- Natural language rules can be easily read
- Externalized rules are easy to change
- Centralized rules enable reuse and consistency
Decisions Need to Change Faster Than Software Applications

Software Development Lifecycle

- Design
- Construct
- Test
- Deploy
- Validate
- Author
- Analyze

Rule Management Lifecycle

- Change Request
- Validate
- Author
- Analyze

Manage and Monitor

- Deploy
- Validate
- Author
- Analyze

IBM ODM 8.0.1 Technical Presentation
Harness the Strengths of Business Users & IT Experts

*Without Business Rules*

*Make changes in months*

Business Analyst
- Writes requirement

Information Analyst
- Creates specifications

Application Developer
- Codes implementation

Functional Tester
- Validates feature

Integration Tester
- Re-validates entire application

Business Analyst
- Runs user acceptance test

Performance Tester
- Ensures application scalability

IT Systems Administrator
- Puts full application into production
Harness the Strengths of Business Users & IT Experts

**Without Business Rules**

- **Make changes in months**

**With Business Rules**

- **Make changes in weeks or days**
- Eliminate rework from misinterpreted requirements
- Speed up rule changes

### Business users (SMEs) take control of the decision logic & minor rule updates

- Business Analyst
- Subject Matter Expert
  - Reads existing rules in natural language
  - Edits or adds new rules without any coding
  - Runs testing and simulation

### IT experts ensure application availability, reliability & new application development

- Performance Tester
  - Ensures application scalability
- SME or IT Systems Administrator
  - Puts updated rules into production

- Information Analyst
- Application Developer
- Functional Tester
- Integration Tester
- Business Analyst
- Performance Tester
- IT Systems Administrator
- Business Analyst
- Information Analyst
- Application Developer
- Functional Tester
- Integration Tester
- Business Analyst
- Performance Tester
- IT Systems Administrator

© IBM 2012
Simple Decision Artifacts Supporting Complex Decisions

If "the vehicle" has anti lock brakes and "the state of residence" is "NJ" then add a 2% discount to 'the coverage quote', reason: "Anti-lock Brakes Discount".

If the channel of the quote contains "CallCenter" and Customer uses Web Channel and all occurrences of quote offered is at least 3 then offer customer a promotion with message: "Multichannel Account Promotion".
### Why Operational Decision Manager?

<table>
<thead>
<tr>
<th>Agility</th>
<th>Efficiency and Productivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operationalize policy changes in days versus months</td>
<td>Improve straight-through-processing</td>
</tr>
<tr>
<td>Decrease cost of implementing business practices, policies &amp; regulations</td>
<td>Involve business users for change requests with built-in governance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Decision Quality and Precision</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Implement more fine-grained, targeted decisions</td>
<td>Perform what-if and impact analyses prior to deployment</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consistency</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Automate and consistently enforce decision policies</td>
<td>Ensure policies and associated semantics are consistent across channels</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transparency, Auditability, Compliance</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Track what decisions were made and why (runtime)</td>
<td>Track what policies were changed and by whom (rule management)</td>
</tr>
</tbody>
</table>
Business Policies Built for Change vs. Built to Last

The Consumerization of Change through Visibility, Collaboration and Governance

**VISIBILITY**  
Achieve clearer line-of-sight to business operations
- Easily readable business rules in natural language
- Testing and simulation to assess impact of rule changes

**GOVERNANCE**  
Align business operations with strategic intent in the face of change
- Rule versioning and snapshots to safely and reliably change rules
- Auditability of changes to ensure traceability

**COLLABORATION**  
Foster cross-functional and cross-divisional outcomes
- Social media style UI to tap into organizational business expertise
- Single repository for business and IT users to collaborate

Enabling regular business people to easily and safely make changes to their business operations
Agenda

- What is Decision Management?
- Introducing IBM Operational Decision Manager
- Business Decisions Defined
- Components, Architecture and Technical Dive
- ODM on z/OS
Next Generation Business Rules

- Manage business policies at scale
- Capture, automate and operationalize your business expertise
- Enable social collaboration to manage and govern business change

Apply Operational Decision Manager…

- To flexibly and reliably manage repeatable, automated decisions
- When decisions change frequently
- To increase straight-through processing
- When decision services can be shared across systems
- To manage and govern large numbers of rules
- When real-time events require immediate actions
IBM ODM v8.5

*Visibility*

New business interface to bring more visibility on how decisions are being changes across the team

*Collaboration*

New social based communication capabilities to improve interactions across the business team

*Governance*

- Enriched controls and governance capabilities simplifying version management at the rules and project level
Visibility Through Operational Decision Manager v8.5

“What’s New” quickly displays deployed assets

Search can return rules, tables and folders
Collaboration Through Operational Decision Manager v8.5

Decision Center enables collaboration on assets of interest and change notifications.

Details on the Activity Stream in the ‘New Features’ section.
Timeline enables a quick view of version history

Details on the **Timeline** in the ‘New Features’ section
Full Decision Lifecycle Management

Manage changes in a safe and predictable environment

Model
Validate
Deploy
Monitor

Business Analyst
Developer
Policy Manager
System Administrator
Rule Administrator

Complex Decisions

Analyze
Author
Test
Validate
Execute
Deploy
Monitor
Why Operational Decision Manager?

- **Agility**
  - Operationalize policy changes in days versus months
  - Decrease cost of implementing business practices, policies & regulations

- **Efficiency and Productivity**
  - Improve straight-through-processing
  - Involve business users for change requests with built-in governance

- **Decision Quality and Precision**
  - Implement more fine-grained, targeted decisions
  - Perform what-if and impact analyses prior to deployment

- **Consistency**
  - Automate and consistently enforce decision policies
  - Ensure policies and associated semantics are consistent across channels

- **Transparency, Auditability, Compliance**
  - Track what decisions were made and why (runtime)
  - Track what policies were changed and by whom (rule management)
Business Policies Built for Change vs. Built to Last

The Consumerization of Change through Visibility, Collaboration and Governance

**VISIBILITY**  
Achieve clearer line-of-sight to business operations
- Easily readable business rules in natural language
- Testing and simulation to assess impact of rule changes

**GOVERNANCE**  
Align business operations with strategic intent in the face of change
- Rule versioning and snapshots to safely and reliably change rules
- Auditability of changes to ensure traceability

**COLLABORATION**  
Foster cross-functional and cross-divisional outcomes
- Social media style UI to tap into organizational business expertise
- Single repository for business and IT users to collaborate

Enabling regular business people to easily and safely make changes to their business operations
Agenda

- What is Decision Management?
- Introducing IBM Operational Decision Manager
- Business Decisions Defined
- Components, Architecture and Technical Dive
- ODM on z/OS
What is a Business Decision?

*Combination of contextual and/or time-based rule artifacts*

**Contextual Decisions**

- **Rules**
  - Application, Process, Service invokes a specific Decision, passing information to be processed by the rules,
  - The contextual rules process the information and form a result
  - Once complete, the Decision passes a result back to the Application, Process or Service to take action on

**Validation Decision**
- Eligibility

**Calculation Decision**
- Pricing

**Classification Decision**
- Gold, Silver, Bronze

**Situational Decisions**

- **Events**
  - Specific situations are defined in event rules looking for specific known sequences which should be correlated from a real-time stream of events.
  - When the known sequence of rules matches a specific sequence of events, then either
    1. This simple decision takes some immediate Action
    2. A synchronous decision is invoked to do additional processing, prior to making a decision and taking some action.

**Occurrence of Decision**
- If customer asks for 3 quotes in 24 hours then provide 2% discount for immediate acceptance

**Missing Events Decision**
- If medical equipment event is not received every 1 hour then send alert to medical staff
Business Decision - Example

A process or application is required to contact a customer. It needs guidance on the most appropriate method to contact them.

Use Classification Decision passing in the customer information.
3 or more alternative results may be returned to process or application.

IF the Total purchases of Account > $2,500 THEN the enterprise value of Customer is GOLD

IF past occurrences the Customer returned a product in the last 2 months is more than 3 THEN classify Customer as a LOW LOYALTY Customer

IF the enterprise value of Customer is GOLD AND the Loyalty of Customer is LOW THEN the personalization action is “Call” ELSE the personalization action is “email”
Decision Tables and Scorecards

If all of the following conditions are true:
- the loan grade in 'the loan report' is 'C'
- the amount of 'the loan' is at least 600000,

then set insurance required in 'the loan report' to true;
set the insurance rate in 'the loan report' to 0.0145;
Rule Authoring: Decision Trees

Visualize decisions and all possible outcomes

- **Actions**: set the corporate score in 'the loan report' to 50, set insurance required in 'the loan report' to true, set the approved of 'the loan report' to false, set the corporate score in 'the loan report' to 30, set the corporate score in <a report> to 15

- **Values**: 50, 30, 0.5

- **Condition**: set the corporate score in 'the loan report' to 40, set the corporate score in 'the loan report' to 20

- **Built-in Gap/Overlap checking**: Partition have gap(s).

- **Automatic Rule generation**: all of the following conditions are true:
  
  - (the loan grade in 'the loan report' is "B")
  - (the Loan to Value of 'the loan' is more than 0.5)

  then
  
  set the corporate score in <a report> to 20 ;
Rule Authoring: Visual Ruleflows

Model and control rule execution sequence
What is Decision Management?

Introducing IBM Operational Decision Manager

Business Decisions Defined

Components, Architecture and Technical Dive

ODM on z/OS
Rule Designer

*Eclipse-based Development Environment*

- Developers
- Business Analysts
Rule Designer

*Test Suite and execution reports*
Rule Execution Server

High Performance and Scalability

- High performance and scalable rule execution
  - Support transactional and batch rule execution
  - Inference (forward-chaining) and sequential rule engine
  - Cluster enabled
- Integrate with Java, XML, WSDL, Enterprise COBOL
- Exposes rule services as
  - Rule Session (POJO, EJB or MDB)
  - Transparent Decision Services (Web Services)
- Rule services management & monitoring
  - Rule Persistence and Versioning
  - Rule Execution statistics & trace
  - JMX-based administration console
Leverage a wide range of platforms to meet the varying needs of enterprise architectures.

Decision Center
(Windows, AIX, Linux, z/Linux, z/OS)

Repository

Windows | AIX | Linux, z/Linux | z/OS

Multiple z/OS deployment options
Decision Center – Business Console

Social Media Style Collaboration

- Maintain awareness across the team
- Ensure automatic notifications of changes
- Ensure team collaboration
Decision Center – Enterprise Console

*Web-based Business Policy Management Environment*

- Rule and Event experts
- Analysts
- Policy Manager
Decision Center - Enterprise Console

A complete Set of Capabilities

- **Authoring**
  - Plain English, Decision Table, Decision Tree
  - Quick edit mode using MS Office
  - Templates

- **Managing**
  - Queries
  - Smart Views
  - Version Management
  - Baseline

- **Validating**
  - Syntactic Check
  - Semantic Check
  - Semantic Queries

- **Testing / Simulating**
  - Decision Validation Services

- **Auditing**
  - Version Management
  - Baseline
  - Query-based reporting

- **Deploying**
  - Query base Extraction
  - Rule service management

- **Administration and configuration**
  - Authorization policies
  - Rule Service
  - Display Options

Enterprise Console
Multiple Release Management

*Greater flexibility for deploying business decisions*

- Enable business users to make changes to a deployed rule application without interfering with work they are doing on an upcoming release
- Merge and diff between releases
Decision Center Console

Integrated decision validation services

- Out-of-the-box ruleset testing in Rule Team Server
- Business impact simulation in Rule Team Server
- Scenario configuration and customization in Rule Studio
- Audit - Decision Warehouse in Rule Execution Server
Agenda

► What is Decision Management?
► Introducing IBM Operational Decision Manager
► Business Decisions Defined
► Components, Architecture and Technical Dive
► ODM on z/OS
How does it work?
Rule Authoring Concepts
Rule Authoring – XOM options

- 4 options for the underlying execution model
  - Java object graph
  - XML schema
  - COBOL copybook
  - PL/I Structures

- Java
  - Standard approach
    - Engine always uses Java regardless of choice
  - Best performance

- XML Schema
  - Java dynamically created at runtime
Rule Authoring – COBOL Copybook XOM

- Support Enterprise COBOL 3.4, 4.1 & 4.2
- A Java is created from the copybook structure
  - Java XOM & Java code to marshal between COBOL <-> Java
  - 01 level structures mapped to class in BOM
- Redefines statements supported
  - Select which redefines structure to import
- COBOL Table support
  - Mapped to Java `List<type>` structures
- COPY statements supported
- Level 88 supported
  - Mapped to methods in BOM
Starting from a COBOL copybook or PL/I include

Scenario
- Existing COBOL containing business rules
- Data model defined in COBOL copybook
- Use ODM to modernize the business policy

Benefits
- Modernize business policies in ODM
- Rules can be invoked ‘naturally’ from existing application
- Business policy/rule lifecycle detached from application lifecycle
Rule Authoring COBOL & PL/I -> XOM

- Support Enterprise COBOL & PL/I
- Java is created from the copybook or include structure
  - Java XOM & Java code to marshal between COBOL or PL/I <-> Java
  - 01 level structures mapped to class in BOM
- Redefines statements supported
  - Select which redefines structure to import
Starting With an Existing Java Project

Scenario
- Existing Rule projects exist that are currently in use on distributed platforms
- Concurrent execution of rules required on System z

Benefits
- Consistent decision rules wherever executed
- Rules can be invoked ‘naturally’ from existing applications on all platforms
- Enables central rule management across System z and distributed execution
- Business policy/rule lifecycle detached from application lifecycle
Deployment options?
IBM Operational Decision Manager on z/OS
ODM Deployment Options on z/OS

• Deployment runtime options on z/OS:
  - Deployed on WebSphere Application Server for z/OS
  - Deployed standalone to z/OS for COBOL and IMS Batch Applications
  - Deployed in CICS TS 4.x and above in JVMServer environment

*Limited Use
Options for Rules Invocation, Integration Patterns and Benefits

**Deployment Option**

Rule Execution Server for WAS on z/OS

**Integration Patterns**

- Web Services, Plain Old Java Object (POJO), Message Driven Beans (MDB), Enterprise Java Beans (EJB).
- Decisions can also be invoked from COBOL applications using the WebSphere Optimized Local Adapter (WOLA).
- WOLA is a functional component that provides an efficient cross-memory mechanism for calls both inbound and outbound calls to WAS z/OS

**Benefits:**

- Unique Cross memory communication, z/OS WLM and a secure infrastructure.
- Ease sharing of Rules: Decision services can be invoked from distributed platforms using the patterns shown above
- Unique Automated Workload Balancing, High Availability and Scalability with WAS on z/OS topology
- Rule execution is offloaded to zAAP reducing mainframe software costs and freeing up CPU cycles.
- Rules shared across Logical Partitions in zOS for unparallel high availability requirements (SYSPLEX)
Options for Rules Invocation, Integration Patterns and Benefits

**Deployment Option**
- zRule Execution Server (Standalone)

**Integration Pattern**
- Simple COBOL API

**Benefits:**
- COBOL applications use copybooks as the interface to invoke decisions making development simple.
- z/OS permits the copying of message buffers from one virtual address space to another which makes ODM efficient, fast and secure as it is within LPAR.
- zRES can be added in an Automatic Restart Manager (ARM) sequence. This will allow zRES to be started automatically in case of a failure.
- Rule execution is offloaded to zAAP reducing mainframe software costs, freeing up CPU cycles.
- Rules shared across LPARS: With DB2 data sharing, rules can be stored in DB2 and accessed from multiple LPARs in a SYSPLEX environment.
- Batch/CICS and Rules Engine running on the same LPAR adds to better performance as there are no network hop.
- Decision services are deployed and instantly available across z/OS without making any application changes.
Options for Rules Invocation, Integration Patterns and Benefits

- **Deployment Option**: zRule Execution Server (CICS JVM)
- **Integration Pattern**: Simple COBOL API

**Benefits:**
- Same benefits as option 2 plus additional
- Running ODM inside CICS allows access to the CPSM* Qualities Of Service like workload management, reliability, availability and security within a CICS infrastructure.
- Local execution within CICS region.

---

* CICSPlex System Manager (CPSM) is a system management tool to manage multiple CICS Intances from a single point.
# Decision Server Options Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>z/OS Rule Execution Server deployed in WebSphere Application Server for z/OS</th>
<th>z/OS Rule Execution Server deployed as a Standalone JVM server environment</th>
<th>z/OS Rule Execution Server deployed in CICS TS v4.x environment</th>
<th>z/OS COBOL Generation Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full support for all rule authoring constructs</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Hot deployment support for new decision versions</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Integration with Decision Center business tooling</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Testing and simulation support</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Decision Warehousing rule auditing support</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Easy sharing of rules with distributed deployments</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Local execution support for CICS TS v4.x</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Full HA &amp; transactional support</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
zRule Execution Server Invocation options in v8.0.1
zRule Execution Server for z/OS – Stand alone

User COBOL Application
Decision Service Stub

Native Control & Comms

JVM
RES Mediation Layer
Rule Server

Address Space

Notification

Web Container
RES Console

Address Space

Deploy

DB2
File System zFS
Runtime Rule Repository
zRule Execution Server for z/OS – CICS TS 4.x and above

CICS TS 4.x and above

User COBOL Application

Decision Service Stub

TRUE

Invocation

JVMServer

RES Mediation Layer

Rule Server

Deployment

Notification

JVM

Web Container

RES Console

DB2

File System zFS

Runtime Rule Repository
Calling to the zRule Execution Environment for z/OS

- Easy integration with existing COBOL applications running in CICS & Batch
- Designed for unique needs of the System z customer base
  - Native zOS Execution in Java
  - COBOL stubs to glue the application to Rules Execution Native Server
  - Integrating core functionality from z based products
- Fully integrated with all key components of the existing BRMS offering
- For enhanced co-location, can implement within an existing CICS 4.x and above region

Diagram:
- User Application
- COBOL Call
- API Stub
- Cobol data structure
- Native
- Comm Layer
- Data Conversion
- JVM
- JRules J2SE RES
- RES Native
- New Address Space
Runtime enablement

- Write the Decision Service invocation in COBOL
- COBOL code remains independent of the Business Rules lifecycle on a stable decision service signature

Decision Service Hot Deployment

- New decision version ‘instantly’ available
- From Rule Designer & Decision Center
- Versioned service made ready for execution from COBOL
- Let running executions complete
New Programming API

* Connect to Execution Region

call ‘HBRCCONN’
   using HBRA-CONN-AREA

* Populate Header with parameter data

* Connect to Execution Server

call ‘HBRRULE’
   using HBRA-CONN-AREA
   IF HBRA-CONN-COMPLETION-CODE = HBR-CC-OK
   THEN
      ...

* Disconnect from Execution Region

call ‘HBRDISC’
   using HBRA-CONN-AREA
Rule execution support for IMS processing regions

- Full capabilities of Decision Server on z/OS now available to IMS batch and online applications
- Natural decision invocation from IMS applications
- Use standard zRES API to call from IMS application to
  – zRES stand alone server
  – RES deployed to WebSphere Application Server for z/OS via WOLA
Testing, simulation, and decision audit support for z/OS

- Run decision tests and simulations from Decision Center in
  - zRule Execution Server – Stand Alone
  - Rule Execution Server on WAS for z/OS
- Supplied sample for accessing test data from z/OS VSAM file
- Decision audit support available through Decision Warehouse capabilities
  - Support for Decision Warehouse in all z/OS Rule Execution Server deployment options
zRES failover support

- Simplified configuration for creating multiple stand alone servers
- Connect client to ‘list’ of servers
- Client fails over to next server on list in event of server connection failure
It’s Time For Demo!

A Decision Service to calculate late fee
The copybook to create the Business Object Model (BOM) is as follows:

01 CustomerLateFee.

05 customerSince PIC 9(03).
05 avgBalance PIC 9(08)V9(02).
05 defaultTerms PIC 9(02).
05 loyalty PIC X(01).
88 platinum value 'P'.
88 gold value 'G'.
88 silver value 'S'.
88 regular value 'R'.

05 lateFee PIC 9(05)V9(02).
The first rule is to determine the loyalty of the customer based on the number of years the customer has been with the bank and the average balance maintained by the customer. It will determine whether the customer is:

- Platinum
- Gold
- Silver
- Regular

### Table 4-1 Rule: Calculate Customer Loyalty Level

<table>
<thead>
<tr>
<th>Customer since (in years) min - max</th>
<th>Average Balance (in $) min - max</th>
<th>Customer Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 - 99</td>
<td>100,000 - 999,999</td>
<td>P</td>
</tr>
<tr>
<td>5 - 10</td>
<td>50,000 - 100,000</td>
<td>G</td>
</tr>
<tr>
<td>3 - 5</td>
<td>30,000 - 50,000</td>
<td>S</td>
</tr>
<tr>
<td>0 - 3</td>
<td>0 - 30,000</td>
<td>R</td>
</tr>
</tbody>
</table>
## Table 4-2  Rule: Calculate Late Fee

<table>
<thead>
<tr>
<th>Customer Level</th>
<th>Default Term</th>
<th>Late Fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>75</td>
</tr>
<tr>
<td>G</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>125</td>
</tr>
<tr>
<td>S</td>
<td>1</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>150</td>
</tr>
<tr>
<td>R</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>200</td>
</tr>
</tbody>
</table>