> GUIDE TO PROCESS RULES

TABLE OF CONTENTS

EXECUTIVE SUMMARY

The generic term "business rule" has been used extensively in a plethora of contexts, models, and domains. Everyone agrees on its importance. At the very core, the purpose of business rules is to capture what should or should not be allowed in a business enterprise.

Businesses can be viewed as an aggregation of rules and all activities in the business and processes as steps undertaken to carry out the rules. Policies and procedures are time-honored methods for collecting and publishing the rules of the business. Business rules are essential for all types of business processes: business-to-business (B2B), which includes application-to-application (A2A), business-to-consumer (B2C), and business-to-employee (B2E). We call these process rules. In the emerging and increasingly dynamic arena of e-business interactions, one of the most important challenges is to capture and put into effect the business rules of the processes that span the extended enterprise.

The purpose of this white paper is to compare conventional business rule techniques with the process-centric business rule paradigm. This white paper describes how Progress® Savvion BusinessManager™ captures and implements process rules.

CATEGORIZING BUSINESS RULES

The following is an illustrative though not comprehensive list of the different types of business rules one might encounter in an extended enterprise:

> **Fundamental quantitative or qualitative rules** governing a business starting at the highest level. The rules governing an organization as defined in its bylaws are definitely in this category, so are the various tax or financial auditing rules (for example, Sarbanes Oxley) to which the business must conform. These governing rules can apply to all aspects of the organization.

> **Operational business rules for the day-to-day management of an organization.** The rule stating that a purchase in excess of a

certain amount requires two approval signatures is an example of a rule in this category. Operational rules cover B2E activities.

> **Mission-critical manufacturing and production rules** for the core services and products produced by the organization. Rules governing the quality of products are examples of this type. This type also covers B2E activities. In the case of extended enterprises, it also spans B2B, which may be implemented through a Web service and even B2C interactions.

> **Commerce or transactional rules** concerning the interaction between trading partners or customers and consumers of the organization's products and services. The time allowed for acknowledging a purchase order or the policy applied to merchandise returns are examples in this category. These rules span all the B2B and B2C interactions that an organization can have with its customers or trading partners.

BUSINESS RULES IN CONVENTIONAL SYSTEMS

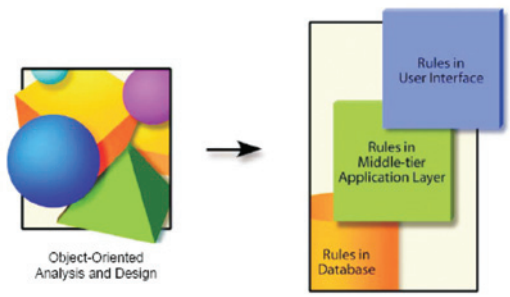
In conventional application development, business rules are expressed in requirements, object-oriented analysis and object-oriented design phases and then mapped onto multilayer application implementations, as illustrated in figure 1:

> **Business Rules in the Analysis and Design Phases.** When capturing a business requirement or problem and methodically moving from requirement and business modeling to analysis, design, and implementation within an object-oriented framework, business rules are sometimes formalized or semi-formalized in object-oriented analysis and design diagrams. The various relationships of classes, the constraints on object instances, and the pre and post conditions of the class or instance methods can all be considered business rules.

> **Business Rules in the Human Interface.** We should also point out that many business rules

Figure 1:

Business Rules in Conventional Application Development and Deployment



could be captured and represented in the human interface layer of the business applications. The human interface could be a conventional graphical user interface (GUI). Increasingly, in most emerging applications, these human interfaces or extended desktops are becoming browser-based portals and more recently provide interfaces through wireless devices such as PDAs or cellular phones. In all these access options, business rules could be presented, encoded and enforced at the presentation or interaction layer of the application.

- **Business Rules in the Application Layer.** Assuming there is a middle tier of application servers or an application layer, some of the business rules are represented in the application layer. Typically, the middle application layer is written in an object-oriented language such as Java or C++. Rules can be expressed as constraint rules or pre and post conditions of methods or encoded within the logic of the methods pertaining to the application layer. More advanced application server technologies can also be used to encapsulate the application logic. The key point is that some of the most important business model rules are often encoded in the application layer.
- **Business Rules in the DBMS.** Some of the business rules are represented within the schemata of underlying database management systems. In fact, DBMS's often provide direct mechanisms for capturing business rules. Many SQL schema constructs such as integrity constraints, triggers, stored procedures, and check constraints on database data value can be used to express and apply business rules.

PROCESS RULES

So far, our descriptions are relevant to the more conventional implementations of business rules. Processes and their rules in these conventional models are either secondary or dispersed in the various levels of application layers, interface conventions or database integrity rules.

When moving into process-centric models and implementations, business rules are expressed as rules pertaining to the internal as well

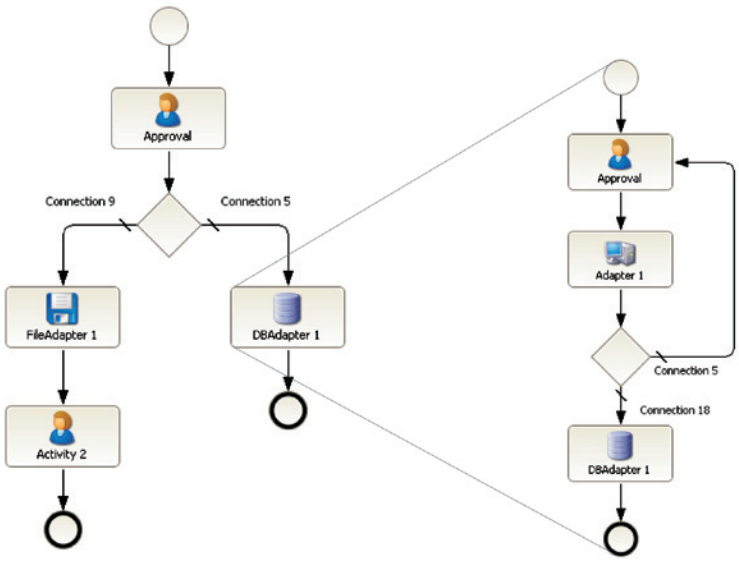
as extended enterprise. In this process perspective, business rules are associated explicitly with business processes. They are, therefore, business rules directly associated with process steps, process performers, and process message exchanges.

The focus on process provides a new, albeit subtle, paradigm for business rules. A process implements and automates business procedures in B2E, B2C, or B2B interactions. In extended enterprises, processes can span customer request fulfillment, internal operations, as well as external business partner interactions. In complex manufacturing, a production lifecycle process can span the entire supply and value chain involving multiple organizations. The model is typically that of a super process that launches and activates various subprocesses across internal or external organizations. This is illustrated in the following figure.

When thinking of business rules in the context of extended enterprise processes, one can see that they belong to three interrelated categories: process flow rules, management rules, and event-action rules.

PROCESS FLOW RULES

There are many types of rules associated with process flow:



- **Logic Rules.** These are rules that capture the ordered execution logic of the process; this is the "flow" diagram. There are many aspects of this flow. The condition for branching is one significant area where rules are captured and expressed directly in the process flow diagrams. The capturing and representation of process tasks that could take place in parallel and those that need to be done sequentially is another area.
- **Partner, Participant, or Performer Rules.** These are rules related to the various roles and completion criteria of tasks assigned in the flow of the process. They also specify which individual, group or organization should perform these tasks.
- **Preconditions, Prefunctions, and Postfunctions**. These are rules that specify the pre and post conditions or functions that need to be satisfied for the execution and completion of identified tasks in a process.
- **Firewall and Security Rules.** Typically, in extended enterprises there are processes that span multiple organizations. The message exchanges, or artifacts, as well as the protocol exchanges may need to be filtered or protected behind a firewall. For instance, an internal purchase order might contain additional information that needs to be kept confidential while an abridged version is communicated to an external trading partner. The business rules here capture and present the filtering, transformations, or reductions in process information that need to happen in process exchanges between trading partners.
- **Execution Time Constraint Rules.** Generally speaking, there are many internal, external, and extended enterprise processes in which time constraints are essential. For instance, one trading partner might expect a response within a specific period of time, or a particular delivery in an organization may need to be completed within a prescribed duration. All such requirements are time constraint business rules on processes.
- **Activity Notification or Acknowledgement Receipt Rules.** Often there are semiformal conventions as well as formal requirements concerning notifications, receipts, and acknowledgement of

messages. They frequently apply to internal procedures and processes indicating, for instance, the circumstances under which an employee or a manager should be sent a notification. These types of rules can also be used in external B2B interaction. Examples include, in contract negotiations, receipt for delivered goods or services, or more basic message exchange acknowledgement.

PROCESS MANAGEMENT RULES

In addition to these there are rules that pertain to the management aspects of the business.

This is very important and often ignored. The management process rules includes the following:

> **Process Monitoring and Reporting Rules.** These specify the single and multiple application reports that are to be periodically generated and monitored. Such reports are very important. Analysis of the management process will dictate what data is important to monitor and analyze during execution of the business processes in order to create the reports. In turn, analysis of the reports provides the basis on which management action is taken. Advanced business intelligence reports are useful to analysts and managers who want a high-level, aggregated view of data, for example, sales ranked by product/region.

> **Scorecards and Console Reporting Rules.** These specify values or key performance indicators that need to be monitored and evaluated for aggregate management reporting encompassing large operational areas or conceptual domains in an extended enterprise. Balanced scorecards and management consoles are two such types of management reporting. Management consoles are able to bring the status and performance of multiple processes into perspective by assembling them in a single view. Balanced scorecards go beyond reporting on process status and measure weighted averages of several distinct areas that could span the entire enterprise. The various graphical depiction of measures in a console or the weight of areas or key performance indication in a balanced scorecard are examples of process management rules.

PROCESS EVENT-ACTION RULES

There are also many event-based rules that span various processes or across the instances of particular process templates or groups of process templates. In extended enterprises, the rules are often activated when there is a correlation between occurrences of events generated from many sources. For example, an enterprise could have an internal rule stating that

expenditures for any purpose require two levels of approval. This means travel approval processes, purchasing processes and processes requesting time off will all require two levels of approval. Therefore, there are many situations where events and rules span multiple processes.

For these rules, the event types include:

- **Process Events.** These may issue from internal, B2B extended enterprise or distributed processes. Examples include creation of a process, activation of a process step, suspension of a process, completion of a step or response, or the completion of the process itself.
- **Application Events.** These are events that can be generated from back-end applications such as ERP systems, EMS systems, databases, or legacy applications.
- **External Events.** Examples include a customer using a shopping cart, a partner request for information on a product, or an employee requesting time off.
- **Monitored Events.** Frequently events generated externally, internally, or through processes need to be monitored. The monitored values are accumulated in data structures that can then be analyzed.
- **Alarms and Time Operations.** There are many situations when events could raise an alarm. Event action rules can be used both to generate and respond to alarms.

The process rules for events are represented through statements that specify the correlation of the events and the actions to be taken when the events happen.

IF EVENTS CONDITION -> THEN ACTIONS

The events could be combined through binary (AND / OR) predicate operators. For instance, if a purchase order process is started and PO received acknowledgement has been received, then a PO acceptance request could be sent to the vendor. Thus, events from multiple sources, internal, external, alarms, etcetera can be correlated.

When the conditions of an event are satisfied, the rule is created and started. The start of the rule could generate other events, change the values of monitoring structures, or invoke operations or methods. Event-action rules are typically named and collected in rule groups.

DEFINING PROCESS RULES

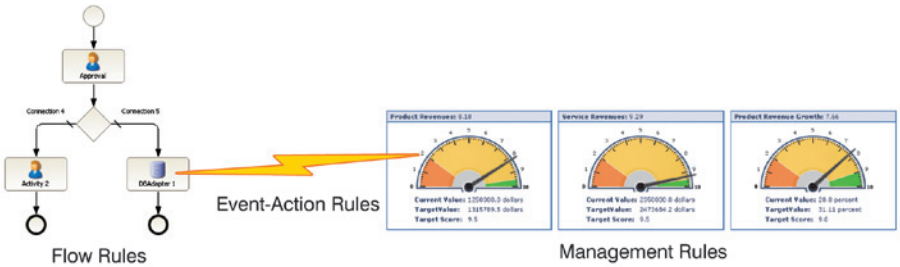
In summary, the process rules described above can be defined through the formula below.

```
Process Rules  =   Process Flow Rules +
                   Process Management Rules +
                   Process Event-action Inference Rules
```

Process rules are, therefore, explicit representations and expressions of rules that apply to internal and extended enterprise business processes. In process modeling, development and deployment of these process rules are expressed in the process requirement analysis and business modeling phases. They are then mapped onto flow rules, management rules and event-action inference rules.

RELATIONSHIP OF COVENTIONTIONAL BUSINESS RULES TO PROCESS RULES

In an extended enterprise application development methodology, the business requirement analysis is performed and business model specifications are defined prior to detailed analysis and especially detailed design of the application. The process rules are associated with the business layer of the



extended enterprise applications. The more conventional business rules emanating from detailed designs are associated with the technology layer of the application. Both are important and relevant, and, in fact, the lines of demarcation between these layers are not rigid or fixed. For instance, process rules can be associated with middle-tier application and databases, especially through event-action and management rules. Significantly, interface rules, as well as middle-tier application and database rules, have a direct effect on business processes.

PROCESS RULES IN PROGRESS® SAVVION™ BUSINESSMANAGER

Progress Savvion BusinessManager is a business process management platform used to move businesses to intranets, extranets or the Internet. It is an ideal e-business platform that lets employees, customers, and partners of an organization collaborate in robust extended enterprises through processes that span the entire supply and value chain. Savvion BusinessManager is the only platform providing direct support of process rules. BusinessManager provides an operational portal and a management portal that allow people to interact with the system. The two engines at the core of Savvion BusinessManager enabling the enactment of process rules are the BizLogic Server and BizPulse Server.

- **BizLogic Server.** The BizLogic Server is a computationally complete state transition process/workflow engine. Each process is an instance of a process template. Process definitions are composed of interconnected worksteps and a set of data slots (objects) that may be modified from one step to the next. Process template definitions include both data and control flow. They provide all the constructs to directly capture and present the process flow business rules of the application. Worksteps can either be manual (that is, performed by a person), or automatic (performed by an external application, system, or machine). A workstep in a process can itself be a process; that is, a process can have nested processes or subprocesses. Hence, complex processes can be designed in a top-down design fashion.

A subprocess can run on a Savvion Process Server in tandem with a parent process. This provides a means for distributing the work within one business process to multiple Savvion Process Servers running on different application servers.

> **BizPulse Server.** As a process is executed, the BizLogic Server generates events creating a complete audit trail of everything that happens during the execution of a process. The BizPulse Server can then process these events in real time. The BizPulse Server provides resource monitoring and management through rule-based event correlation. The BizPulse Server has a mechanism to listen to events from various sources through event adapters. It checks the events it receives against the current rules and "fires" the rules when their conditions are met. The BizPulse Server provides a rule language that allows for defining business rules applicable to events from the BizLogic Server and other applications and systems that generate events. BizPulse Server rules have advanced time constructs such as dates, interval, duration, and scheduling. A rule can include events from various sources; therefore, one can take intelligent actions that involve complex business conditions spanning various seemingly disparate enterprise systems. This, in a sense, provides a way for rulebased inferencing as well as integration. The BizPulse Server supports all the process rule event types.

A rule is made up of conditions and actions. It identifies a certain pattern of events and specifies the actions that should take place if that event pattern occurs. The action can be a new event, an alarm, a report update, or a corrective action such as starting a process in the BizLogic Server or starting an action in an external system through adapters. The BizPulse Server can collect and aggregate process rule results in **InfoPads**. InfoPads are data structures that store information related to an application. InfoPads are created and populated through rules. The information aggregated in an InfoPad can be viewed graphically in BPM Portal.

The dynamics between the BizLogic and BizPulse Server as well as external systems for generating and processing rule events is illustrated here.

The BizPulse Server (referred to as BizPulse in the graphic) receives events coming from all kinds of sources including the BizLogic Server (referred to as BizLogic in the graphic), internal events or alarms, or external adapters. Based on these events, rules trigger actions. Examples of actions include sending e-mails, scheduling events, monitoring functions or communicating

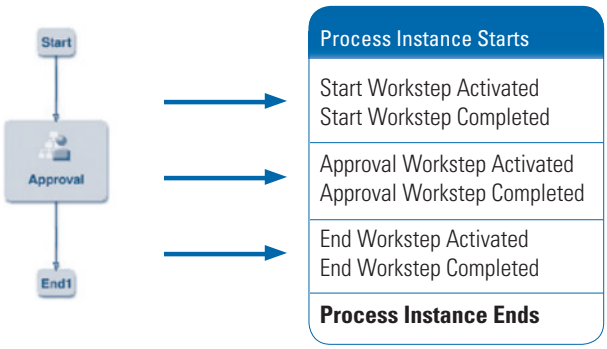
back to the sources via their application programming (API) invoking other applications. A single rule can take multiple actions. The BizLogic Server generates several types of events. The following diagram illustrates a very simple process consisting of a Start workstep, an Approve workstep and an End workstep.

More specifically, the events generated by BizLogic Server and acted upon by the BizPulse Server can be described as follows:

- **Process Instance Created.** Such an event is generated when the process instance is created in the process engine, but not yet started.
- **Process Instance Activated.** Such an event is generated when the process is started. At this time, some predefined initial values are known for some attributes (data slots) of the process instance.
- **Workstep Started.** Each workstep of the process will generate an activation event, including the Start workstep of the process.
- **Workstep Completed.** At the end of each workstep, this event is generated.
- **Process Instance Completed.** Such an event is generated when the process instance is completed.

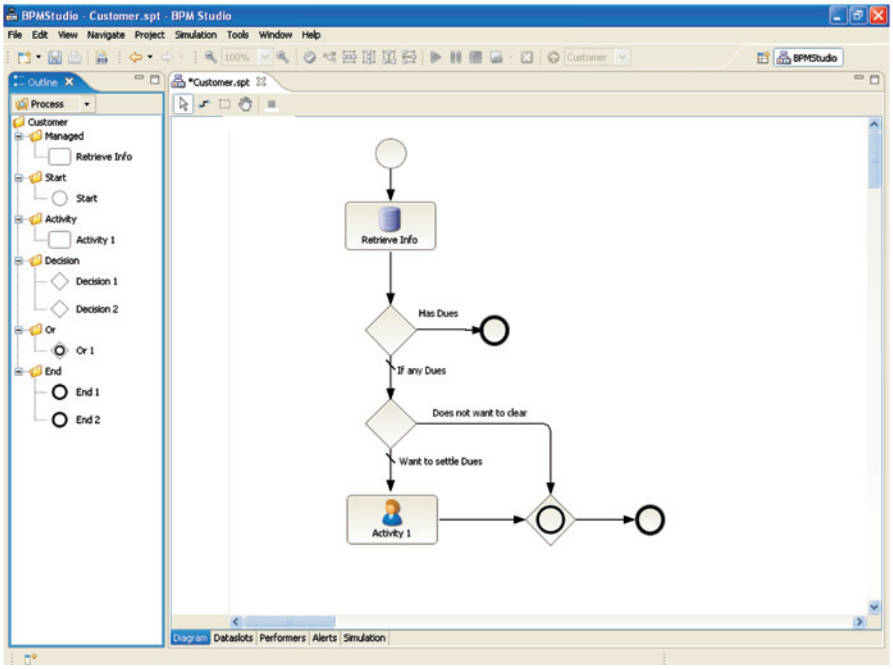
EXAMPLE OF PROCESS RULES

Here is an example of how process rules are applied to a credit and sales order processing application. The application needs process rules to record information about the customer and the invoices. Process rules are needed to retain information about the customer like an ID, payment status, reliability status, and other such information. Invoices have information



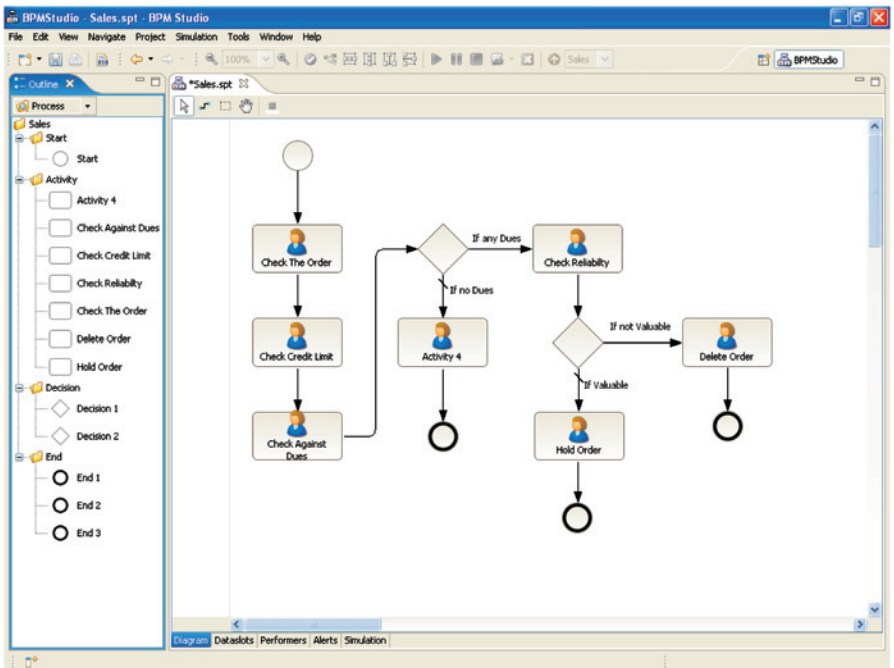
about the goods ordered, the total amount the customer has to pay, and the customer's ID. The customer is given a credit limit of $50,000 and a credit pay period of 30 days in which to clear the amount due. Process rules are also needed here to keep track of the payments made and amount due if any. Finally, process rules are also necessary to block or delete orders, depending on the customer, if the amount due is not cleared.

The sales order flow logic below checks the customer's order and the customer's credit limit against the amount due. If there is no amount due and the credit limit is not exceeded, the order is processed.



If there is no amount due and the credit limit is not exceeded, the order is processed. If there is an amount due to be paid and the customer is an important one, the order is put on hold. Otherwise, the order is rejected.

The second logic flow deals with the customer's payment of the amount due. After retrieving the information about the customer, if the



customer wants to settle a pending amount due, the status of the customer and associated account information are updated.

Next we list some of the process event-action rules of the application containing these two processes, providing the purpose of each event-action rule:

- **Purpose:** Infopads necessary to store information about the customer and the invoice. The following rule creates the Infopads.

  - **Rule:**

```
If      (EventA = initRules)
Then    (Ceate infopads CustomerInfo and
        InvoiceInfo and save them in
        the database)
```

- **Purpose:** If the TotalAmount to be paid by the customer is less than or equal to the credit limit and the PaymentStatus in the CustomerInfo InfoPad is “clear,” then the customer should be allowed to go ahead with the processing but should also be reminded about the credit pay period which is 30 days. The Payment Status in the CustomerInfo should also be updated.

  - **Rule:**

```
If      (EventA = WorkStep Completed) and
        (EventA.TotalAmount <= $50,000 and
        (CustomerInfo[] .PaymentStatus =
        “clear”)

Then    (Schedule an event to be triggered
        after 29 days which sends an email
        (if in fact an email is appropriate)
        to the Customer asking him to clear
        any of the credit taken if it is
        still pending

        (Update the CustomerInfo InfoPad
        with the value “credit” for the slot
        PaymentStatus against this customer,
        identified by his ID)
```

- **Purpose:** Here the second process, CustomerPayment, comes into the picture. This process keeps track of the payment status of the

customer. If the payments are all cleared, it will update the status of that customer in the infopad CustomerInfo. After the slot is updated, this will also resume the process instance that was suspended because of the uncleared amount due.

> Rule:

```
If      (EventA = WorkStep Completed) and
        (EventA.ProcessTemplateName =
        "CustomerPayment")

Then    (Then update the corresponding status
        of the slot PaymentStatus in the
        InfoPad CustomerInfo) (Resume the
        process instance after the
        PaymentStatus has been changed from
        "due" to "clear"
```

> **Purpose:** If the TotalAmount to be paid by the customer is less than or equal to the credit limit and the PaymentStatus in the CustomerInfo InfoPad is "due" and the customer priority is low, then send mail to the customer stating the reasons for not processing the order and also remove the process.

> Rule:

```
If      EventA = WorkStep Completed) and
        (EventA.TotalAmount <= $50,000) and
        (CustomerInfo[] .PaymentStatus = "due")
        and (CustomerInfo[] .status = "low")

Then    (Send mail to the Customer about the
        status)(Remove this process instance
        since order is not going to be
        processed further)
```

So far we have illustrated the flow rules and event-action rules. Management rules are also important. The management portal of Savvion BusinessManager allows the status of processes as well as the data aggregated in InfoPads by event-action rules to be viewed and analyzed.

The InfoPads in the rule set above accumulated information about customers and their invoices. The managers could also accumulate data about the sales people and their customer orders as illustrated.

The manager can then apply management rules to the performance of sales people who exceed or fall below the expected averages or potentially

identify and resolve unexpectedly favorable or poor results in certain months or seasons.

CONCLUSION

This paper has introduced and defined a very important category of business rules called "process rules."

```
Process Rules = Process Flow Rules +
Process Management Rules +
Process Event Inference Rules
```

All three elements of process rules are extremely important for capturing process rules in advanced business applications. In extended enterprises these applications can span B2E, B2C, or B2B processes.

The paper has also shown the key process rule components and capabilities of Savvion BusinessManager.

In fact, Savvion BusinessManager is the only platform that supports directly all components of process rules, process flow rules, management rules, and event-action inference rules. Because of the direct support of Savvion BusinessManager for process rules, it has become very easy for business application owners to develop and deploy their process rules over extended enterprises

PROGRESS SOFTWARE

Progress Software Corporation (NASDAQ: PRGS) is a global software company that enables enterprises to be operationally responsive to changing conditions and customer interactions as they occur. Our goal is to enable our customers to capitalize on new opportunities, drive greater efficiencies, and reduce risk. Progress offers a comprehensive portfolio of best-in-class infrastructure software spanning event-driven visibility and real-time response, open integration, data access and integration, and application development and management—all supporting on-premises and SaaS/cloud deployments. Progress maximizes the benefits of operational responsiveness while minimizing IT complexity and total cost of ownership.

WORLDWIDE HEADQUARTERS

Progress Software Corporation, 14 Oak Park, Bedford, MA 01730 USA
Tel: +1 781 280-4000 Fax: +1 781 280-4095 On the Web at: www.progress.com

For regional international office locations and contact information, please refer to the Web page below:
www.progress.com/worldwide

Rev. 04/10 | 6525-129666