Int J Adv Manuf Technol (2006) 27: 999–1010
DOI 10.1007/s00170-004-2275-4

ORIGINAL ARTICLE

Jun-Jang Jeng · Henry Chang · Jen-Yao Chung · Markus Ettl · Steve Buckley · Haifei Li

COSAR: commitment-oriented “sense and respond” system for microelectronic manufacturing

Received: 10 December 2003 / Accepted: 2 June 2004 / Published online: 20 April 2005

Abstract Microelectronic manufacturing deals with the efficient coordination of manufacturing processes and owning enterprises along a value chain to provide microelectronic products to customers. Microelectronic manufacturing is an important subject of the supply chain management because of the vital role of the microelectronics industry in the global economy. The systems for microelectronic manufacturing processes are usually linear, rigid, and produce solutions that are far from optimal. This paper presents a commitment-oriented “sense and respond” system, called COSAR, for microelectronic manufacturing. COSAR enables the development of adaptive and configurable manufacturing systems. The concept of commitment is explained in this paper. Furthermore, the concept of “chain of commitments” has been extended to “network of commitments.” Functionality of COSAR can have vertical, horizontal, or mesh decomposition. The foundation, architecture and implementation status of COSAR are discussed.

Keywords Commitment · Microelectronic manufacturing · Sense and respond

1 Introduction

Microelectronic manufacturing processes are comprised of sophisticated interactions among multiple systems and decision makers, highly complex computation in order to optimize manufacturing plans, and non-trivial integration issues embodied in business data and processes themselves. Traditional microelectronic manufacturing processes are usually linear and rigid; and once done, will be very hard to change. Modern microelectronic manufacturing systems, however, attempt to replace inflexible supply chains with more adaptive supply networks that will result in configurable and more effective management of cross-company manufacturing and logistics to achieve a competitive advantage. The ultimate goal of many companies is to build adaptive and “sense and respond” manufacturing systems flexible enough to sense and respond to changes in their manufacturing environment [1].

This paper presents an architectural framework and implementation, called COSAR, for commitment-oriented “sense and respond” systems applied to the domain of microelectronic manufacturing. We discovered that the issues of managing complicated manufacturing processes can be addressed by the ideas and the architecture embodied in COSAR. Since the totality of all activities of microelectronic manufacturing can be viewed as distributed problem solving process, COSAR was intentionally built as an agent-based framework. We explore the conceptual foundation for this framework and develop the COSAR architecture based upon the foundation.

According to Merriam-Webster’s collegiate dictionary, commitment is “an agreement or pledge to do something in the future”. Commitments can be agreed between trading partners (called external commitments), or between internal parties within a business (called internal commitments). The definition of business commitments captures not only the current stable states (“agreement”) but also the future actions (“to do something”) and constraints (not to do something); therefore it is an appropriate concept to describe certain types of business relationships and interactions that may require both agreements and actions from participating parties. A set of business commitments can also be used to establish the agreement of a management platform to its customers regarding how their artifacts are to be managed in this platform. In our opinion, the concept of commitments nicely fits into the microelectronic manufacturing environment where the management system likely manages multiple trading partners (customers and suppliers) and multiple internal parties (internal distribution centers, multiple factory shops). As will be elaborated in the following sections, the concept of commitment is the corner stone of the COSAR systems and architectures. In the microelectronic manufacturing scenario, commitments can be used for describing management scenarios, constrain-

J.-J. Jeng (✉) · H. Chang · J.-Y. Chung · M. Ettl · S. Buckley · H. Li
IBM T.J. Watson Research Center,
Yorktown Heights, New York, USA
{jjjeng,hychang,jychung,mettl,sbuckley,haifeili}@us.ibm.com

ing agent behavior, measuring manufacturing metrics, enforcing desired actions, rendering commitments to other agents and so on.

The rest of this paper is organized as follows. Sect. 2 provides the background information for the domain and the COSAR approach. The conceptual foundation of COSAR is described in Sect. 3. The notion of commitment is described in Sect. 4. The COSAR architecture and implementations will be presented in Sect. 5. Related work and conclusions will be given in Sects. 6 and 7, respectively.

2 Background

This section provides the background information of the domain where COSAR is aimed. We will describe the problem domain in general and a scenario in particular. The COSAR approach for this domain will be briefly presented as well.

2.1 Problem domain

The problem domain we have been working on is supply chain management (SCM) for microelectronic manufacturing [2]. The core flow for microelectronic manufacturing contains sub-flows such as:

- From wafers to devices: The *wafer* is a round thin piece of silicon that looks similar to a CD. The wafers go through an elaborate procedure that has cycle times (time to complete the task) of one to three months, where millions of circuits are carefully etched onto it. When the wafer is completed, it is transmitted to a final test and then cut into small individual rectangular shaped parts called *devices*. Typically cycle times are a few days to two weeks.
- From devices to modules: The devices are then placed on a substrate and packaged to create a *module* which takes between a few days to two weeks.
- From modules to cards: Modules are then combined together on a *card* that is the topmost-level component.

Depending on the customer, a microelectronic manufacturer may dispatch wafers, devices, modules, or cards. Within each major manufacturing activity there is a series of operations and extensive testing. Operations are grouped into higher level modules such as sectors, work centers, and stages. The actual matching models deployed by the manufacturing planning team work with dynamically established stages as the core manufacturing activity or decision point. The number of levels depends on the specific model. The key decision points and corresponding decision support systems in the SCM process can be categorized by the supported *supply chain component* and its *time frame* or *decision tier*.

The decisions in the semiconductor industry typically fall into one of four decision tiers: *strategic*, *tactical*, *operational*, and *response* (dispatch). The categories are based on the planning horizon, the apparent width of the opportunity window, and the level of precision required in the supporting information. The decision tier of *strategic scheduling* is driven by the time frame required for the business plan, resource acquisition, and new product introduction. The timeframes concerned here are from three months to seven years into the future. The tier of *tactical scheduling* deals with problems the enterprise encounters in the next week to six months. Issues considered are made of yields, cycle times, and binning percentages, delivery dates estimated for firm orders, available “outs” by time buckets estimated for bulk products, and daily going rates for schedule driven product are set. The tier of *operational scheduling* deals with the execution and achievement of a weekly plan such as the shipments made, serviceability levels to be measured, and recovery actions to be taken.

The tier of the *real-time response* system addresses the problems of the next hour to a few weeks by responding to conditions as they emerge in real-time and accommodate variances from availability assumed by systems in the plan creation and commitment phases. COSAR systems are built to realize the capabilities of “sense and respond” in order to satisfy the requirements addressed in the above tiers by providing awareness among various elements such as humans and business artifacts, etc. A COSAR system dispatches scheduling decisions regarding either monitoring or controlling policies of actual manufacturing flows, and instructs the operators about the next steps of achieving manufacturing commitments. COSAR dispatches scheduling decisions that are concerned with monitoring and controlling of the actual manufacturing flow or logistics, and instructs the operator the next steps to achieve current manufacturing commitments. COSAR can also make decisions concerning trade-offs between running test lots for a change in an existing product or a new product and running regular manufacturing lots, lot expiration, prioritizing late lots, positioning preventive maintenance downtime, production of similar product to reduce setup time, down stream needs, and simultaneous requests on the same piece of equipment.

2.2 A use case for continuous demand-driven build plan and inventory optimization

We provide an example of a typical use case in the domain of microelectronic manufacturing.

2.2.1 Overview

End-of-quarter *revenue targets* (per module family) are released/updated after the meetings among business line managers and executives. A business line manager (BLM) has a predetermined set of module families for which he/she has financial responsibility and, therefore, whose actual revenue (accumulated so far) and revenue outlook (for remaining weeks in the current quarter) he/she is interested in tracking against the revenue target of the current quarter. Whether the progression of the accrued revenue is on target or above target or below target is determined by the system using a wineglass model [3]. The BLM has concerns for the “below target” situations and may choose to set up corresponding personal alerts for them. The above-mentioned monitoring capability is provided through a business dashboard for microelectronic manufacturing.

2.2.2 Main course

1. On the *i*th day of the current week, a BLM selects a set of saleable part numbers (they are built from the same common wafer and they are under the BLM's financial responsibility) to view the future weekly actual sales and planned demand quantities through COSAR monitoring portal.
2. Detection of situation and alert:
   (i) COSAR issues an alert showing the current sales quantities of some selected saleable part numbers in the *n*th week are out of their bands. A band is defined as an *n*-week-ahead statistically acceptable interval for the current sales quantity.
   (ii) COSAR recommends adjusting the planned demand quantities and safety stock requirements for the *n*th week.
3. Recommendation:
   (i) COSAR invokes a demand planning module and inventory planning module to provide recommended demand quantities and safety stock requirements for the *n*th week.
   (ii) COSAR disaggregates the weekly demand quantities into daily demand quantities.
   (iii)COSAR recommends altering the daily build plan in order to optimally match new daily demand statements, and thus achieve high serviceability, and minimize manufacturing and inventory costs.
   (iv)COSAR invokes MRP explosion and implosion module to generate optimal daily build plan, including common wafer start quantity and manufacturing release quantity in each part number level within the BOM chart associated with the selected saleable part numbers. The optimal build plan minimizes manufacturing requirements as well as resulted inventory levels, hence minimizes the manufacturing cost and inventory cost.
4. Prediction and risk assessment:
   (i) COSAR predicts the to-be manufacturing cost, inventory cost, and service level associated with the selected saleable part numbers based on new demand statements and new build plan.
   (ii) COSAR predicts the as-is manufacturing cost, inventory cost, and service level associated with the selected saleable part numbers based on new demand statements and old build plan.
   (iii)COSAR concludes the financial and serviceability benefits of applying newly recommended demand statements, safe stock requirements, and build plan.
5. Decision making by BLM:
   (i) The BLM summarizes the financial and service benefits and reports to the strategic management team (BLEs and executives) for their approvals.
   (ii) Upon the approval, the BLM releases the new build plan to sites for manufacturing execution. The BLM releases the new demand statements to procurement team to alter buy plan.

2.3 The COSAR approach

The emergence of e-business has dramatically changed the context in which decision-making takes place. While fundamental human and organizational processes remain largely unaffected, e-business places new constraints and demands on the decision maker to provide better service to the customers:

- Due to the increased rate of possible change in e-business, decisions must be made more quickly. Process participants must have instant access to information that is relevant to the current business context.
- Inter-organizational processes require visibility for each involved party. Increased alliances and co-operative arrangements mean that information is increasingly external.
- Result-oriented or outcome-oriented process metrics allow the decision maker to focus on the information that is most relevant for their business tasks. Due to the growth of information available on the Internet and modern IT systems, decision makers are exposed to an increasing amount of data.
- Increased telecommuting reduces the opportunity for informal information exchange and decision-making. Often decisions must be made in real-time to allow a straight-though processing.

These factors imply that a limited focus on the provision of information and analysis tools is insufficient. To be effective, decision support must take a broader view of decision-making. Figure 1 shows the decision flow supported by COSAR. This decision process is often referred to as sense-and-respond loop and was popularized by Haeckel [20].

COSAR supports the sense-and-respond concept by providing modular software support for each of the four indicated processes. *Measurement/sensing* is supported by managed objects that interface with business process systems and by managed agents which provide data extraction, transformation, and loading capabilities. *Interpretation* is provided by situation agents that identify situations requiring attention such as unexpected events and out-of-tolerance metrics. *Analysis* is performed by analytics agents that support risk-

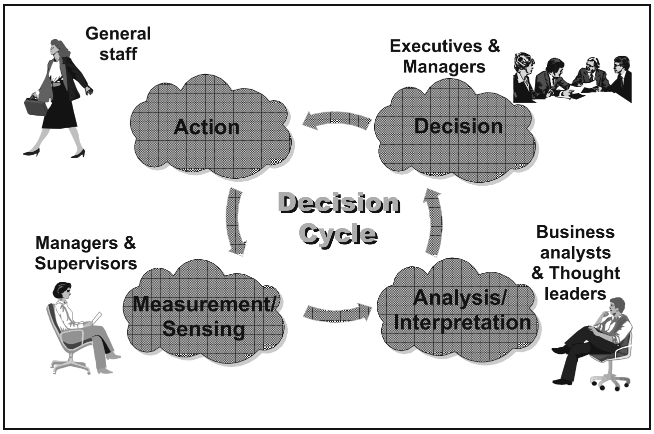


Fig. 1. Decision cycle (sense & respond loop)

based analysis of potential response actions. Such response actions often take the form of optimizations that attempt to achieve enterprise goals subject to current business constraints. *Decisions* are made by users who interact with dashboard portals that display relevant information. *Actions* are performed by management agents that interface with managed objects to trigger exception processes in business process systems.

3 Conceptual framework

In general, COSAR can be categorized as a system that is continually interacting with its managed substrate, and that is capable of autonomous actions in this substrate in order to meet its management commitments. A platform for COSAR takes inputs from the managed substrate, and produces actions that affect it. As such, COSAR interacts directly or indirectly with the situated entities in the managed substrates. Examples of situated entities include business processes, business organizations, managed resources, and business systems, where:

- *Business processes* are the first-class citizens that can be observed, measured, analyzed, and managed. In the example of PLM, managed business processes can be supply-chain management, customer relationship management, enterprise resource planning and so on.
- *Business organizations* refer to the participating parties of the business processes and BPM systems which are managed by COSAR. Business organizations can come from many roles: enterprises, business analysts, BPM system administrators, BPM integrators, and the business executives who are interested in knowing the status of business processes.
- *Management policies* refer to the management contracts established between COSAR and virtual organizations. An example is the maximum cycle time of some supply chain management process shall not be greater than 48 hours. If the agreement was violated, a certain penalty will be exerted.
- *Business systems* comprise manageable entities that are situated in the substrate. Resource manageability defines information that is useful for managing a resource and details the aspects of the resource including the instrumentation which allows COSAR to interact with it. There have been many standards of defining manageability at various levels, e.g., SNMP [4], CIM [5] and M12 [6]. Through instrumentation, a resource is turned into *managed* resource because its state can be perceived, aggregated, analyzed, and modified through the standard interfaces provided by the instrumentation layer that is located between COSAR and its environment.

3.1 Horizontal decomposition

The functionality of COSAR can be decomposed either horizontally or vertically. Figure 2 shows that COSAR is decomposed horizontally into three pillars: *perception*, *evaluation*, and *actuation*.

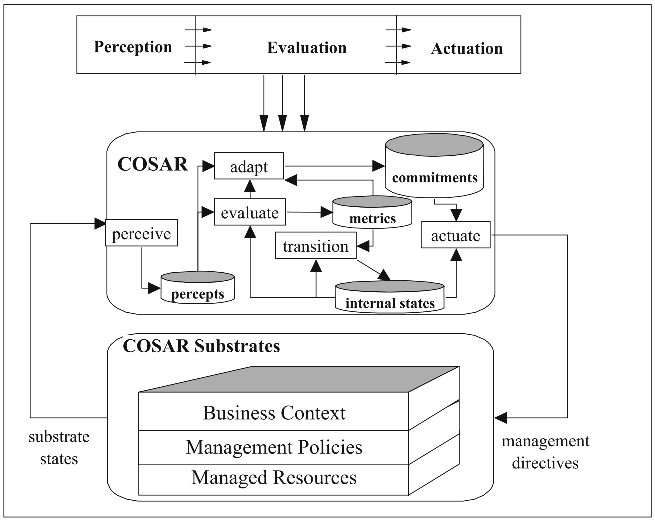


Fig. 2. Horizontal decomposition

- Perception pillar receives the data and events from the COSAR substrate.
- Evaluation pillar processes the perceived information.
- Actuation pillar renders management directives to the managed entities within the substrate.

Conceptually, the evaluation pillar can be further decomposed into three sub-functions: *measure*, *transition*, and *adapt*; and four local data stores: *percepts*, *metric stores*, *control states*, and *commitment store*. The percepts store contains the perceived values from the COSAR substrate. The function *measure* computes the metrics according to the values of percepts and stores them into the *metrics store*. The control states represent the current situation of COSAR as a whole. Control states are different from the states perceived from the substrate. In general, control states capture the status of COSAR as a whole, and the environment states capture the status of the managed entities within the COSAR substrate. The function *transition* changes the control state of COSAR according to the current control state, the local commitments, and the resultant metrics. The function *adapt* changes the local management commitments according to the existing control state and the business policies. The commitments store contains the local management commitments that the COSAR enforces on the managed entities of the substrates, business organizations, business systems.

3.2 Vertical decomposition

COSAR can also be decomposed vertically into three layers: *Reactive* management, *deliberate* management and *reflective* management. Figure 3 illustrates the vertical decomposition, where solid lines represent the flows of management directives, and the dotted lines represent the flows of management events.

- The reactive management layer responds to the management events quickly and directly through scripted business process

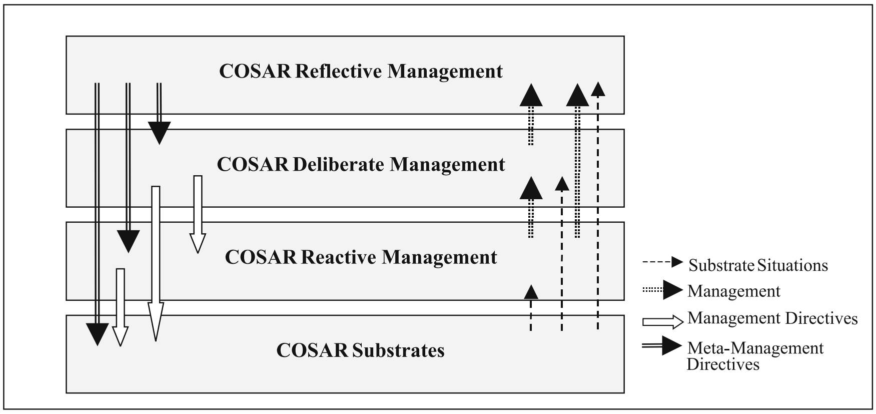

COSAR Reflective Management
COSAR Deliberate Management
COSAR Reactive Management
COSAR Substrates
Substrate Situations
Management
Management Directives
Meta-Management Directives

Fig. 3. Vertical decomposition of COSAR

models. A notable example of the reactive management is deterministic workflow management where workflow models are defined at the build time and executed at the run time. Another example is the alarm system that will notify the system administrator if some managed resource is suffering from severe performance problems, and demanding immediate attention.

- The deliberate management layer performs managerial tasks that require more reasoning and more complicated computation. It is not uncommon that COSAR needs to provide decision support capability so more intelligent management directives can be derived toward managed resources. An example of such managerial tasks are the business processes with the sense-and-respond capabilities [1, 7]. Another example of deliberate management is mapping QoS metrics from IT-level into business process level and vice versa. An event such as "disk failure" may mean little out of business context. However, it may imply a loss of gigantic capital for a business organization if it has a causal relationship with critical business process performance such as financial trading. The mapping rules in the layer of deliberate management should capture this relationship to prevent business loss.
- Reflective management layer enables COSAR to maintain information about itself and use this information to remain extensible and adaptable [8]. Reflexive management layer performs meta-management directives unto the lower management layers and managed entities. A meta-management directive is a higher sphere of control such as adapting the management commitments, modifying measurement and analysis algorithms in the deliberative management layer, or changing the alarm rules in the reactive management layers. As such, COSAR can have detailed knowledge of the managed resources, current status of managed business processes and business systems, the ultimate capacity in the inventory, performance expectation, and all connections to other systems to manage itself. Therefore, through reflective management mechanism, COSAR achieves the goals of both 2nd order management and autonomic computing [9].

3.3 Mesh decomposition

Horizontal and vertical decompositions can be combined to form *mesh decomposition* as shown in Fig. 4. The mesh model is used as a formal modeling tool for cognitive architecture [10]. We found that the mesh model is very suitable to illustrate different architectural aspects of management spaces for COSAR. Figure 4 defines nine regions, ($S_{i,j}$ $1 \leq i, j \leq 3$), called *management spaces* (or *m-spaces*) and their potential interactions. Only legitimate flows, both management directives and management events, are allowed to be transmitted between m-spaces. The entities in the COSAR substrates emit management events and receive management directives. Evaluation is completely an internal processing inside COSAR without interaction with its substrate. The meta-management directives are rendered only

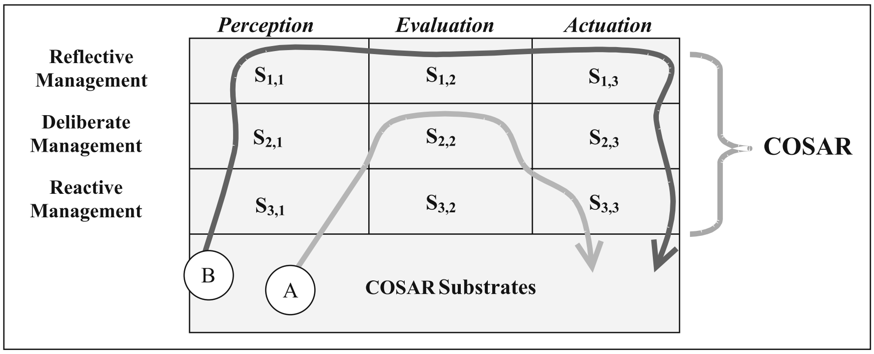


Fig. 4. Mesh decomposition of COSAR

through the actuation pillar. Within COSAR, management events can be generated or transformed between layers upwardly and management directives can be rendered downwardly.

Figure 4 shows two typical management scenarios based upon the decomposed m-spaces:

- Flow A is a common management scenario: management events are perceived by the reactive management layer, the evaluation is performed by the deliberate management layer, and the actuation is activated through the reactive management layer. This management scenario is called Ω model of management flow, which is the most common scenario among all.
- Flow B is a meta-management scenario where management events are delivered all the way to the reflective layer, evaluated, eventually some meta-management directives are delivered through the actuation pillar in the reflective management layer. An example of such scenario is data mining of the business data residing in the substrate. The reactive management layer extracts the data, the deliberate management layer transforms and loads the data, and the reflective management layer analyzes the data and makes/suggests business decisions accordingly.

4 Commitment-based manufacturing process

In the domain of microelectronic manufacturing, many business processes are required to ensure the functioning of the system. Hence, COSAR interacts with multiple participating parties such as business line managers who make decisions, building superintendent who conducts the day-to-day scheduling, production planning team who determines intermediate target outs, manufacturing team who makes the final determination of what is built and when. Moreover, COSAR needs to interact with multiple systems such as:

1. The central planning engine (CPE) that helps decision makers determine how to best meet prioritized demand without violating temporal, asset, or capacity constraints.
2. The optimal manufacturing resource planning (OMRP) tool that assists decision makers define detailed instructions about what manufacturing activities must be accomplished and when they must be completed.
3. The available to promise (ATP) tool that enables an organization to dynamically reallocate projected supply in response to incremental changes in the demand statement (new orders arriving, orders being filled, and order changes or cancellations) according to business policy guidelines, identify projected shortfalls with respect to committed orders, and provide real-time order commits and status.
4. The demand management (DM) tool that helps decision makers coordinate demand estimates from different sources such as orders, sales rep forecasts, customer forecasts, internal demand, and marketing forecasts in logical step by step

Therefore, the functioning of a manufacturing system such as COSAR depends on the coordination and management of many autonomous business processes and systems. If a decision maker makes an arbitrary decision without regard of any consideration of the other parts of the whole manufacturing process, the whole system will not work properly. Thus, there is a need for a set of management policies that work as “contract” among business processes and systems governing their behavior so as to guarantee the interests of all the stakeholders. These management policies can be viewed as exogenous commitments that a decision maker makes with respect to the stakeholders; or as endogenous commitments that are exploited to enable and enforce the behaviors of managed business processes and systems, named as COSAR substrates in the previous section. Hence, a commitment-based management is the natural choice of implementing COSAR [11].

The management commitments involved in COSAR can be of many types, for example, the pre-defined demand boundaries, the inventory level thresholds, system performance, and so on. Abstractly, management commitments define the constraints that would follow certain courses of actions, or to hold certain agreed and trusted situations manifested by the entities in the COSAR substrates, also called *expectations*. A commitment concerns either acting in a certain way, or it can be a commitment to hold a certain expectation. Commitments can be about the past or the future, where the former are called retrospective commitments and the latter are called prospective commitments. COSAR can be a standalone application or be composed of many management agents. In the latter case, each agent may hold its own commitment. A commitment consists of the following entities:

- *Actions* that it performs and resources (data) required.
- *Resources* that are governed by the commitment.
- *Expectations* that the commitment hold to and each expectation is composed of *situations*.
- *Reponses* that bind actions with expectations.
- *Triggers* that initializes the evaluation of expectations.

To execute an action is to perform the action; to execute an expectation is to assert it. A response will be formed in a commitment via bindings of expectations and actions. Notably, a response can be executed when the bound expectations are either met or violated depending which way has been defined in the expectation. Trigger elements in a commitment define when an expectation to be evaluated and consequently what actions to be executed. Examples of triggers include the arrival of events, timer events, substrate state changes, data conditions etc. Some meta-actions can be defined to perform actions on commitments themselves. One type of meta-actions is to *commit* a commitment to an agent. This is usually a future commitment, but it can be used for the establishment of a past expectation. Another type of meta-action is to put an expectation relationship between agents. For instance, an ATP agent *expects* an inventory agent to have an inventory level less than a certain quantified value. If the expectation is violated, some actions will be taken based upon the corresponding bound *response* that is defined in the ATP agent. A commitment can be applied to several agents, i.e., those agents *share* the same commitment. We can thus define the *locus of control* for the commitment C by defining a set containing the agents committing to C and the resources governed by C. In COSAR, it is assumed that an agent will always communicate its commitments truthfully with the other agents in response to queries and actions. Figure 5 presents the meta-model of management commitment. A commitment consists of triggers, resources, actions, expectations, and responses.

Each management agent may have one or more commitments. Management agents commit to one another through the response bindings. Resources model the entities in COSAR substrates. The diagram shows some examples of resources such as sales status, balanced scorecard, order status, demand status, and inventory. In fact, a resource can represent an even higher level concept such as manufacturing processes/activities or participating parties. Triggers initiate the process of evaluating whether expectation holds or not. A trigger consists of one or more situations which are nothing but logical expressions that can be evaluated by the inference engine of COSAR.

Since management agents can commit to one another, the "commit-to" relationships form a commitment network with agents as nodes and commitment as the edges connecting nodes. For example, the sales agent is committed to send an alert to the demand agent when the current sales quantities of some selected saleable part numbers in the nth week are out of their bands. The demand agent is committed to the portal agent to provide a list of recommendations on adjusting the planned demand quantities and safety stock requirements for the nth week. However, the demand agent cannot work by itself. The demand planning agent and the inventory planning agent commit themselves to the demand agent such that they can provide the demand agent recommended demand quantities and safety stock requirements for the nth week. The network of commitments (NoC) is a generalized notion of chains of commitment [12]. The NoC is used to define the triggering points, the control and data flows, the monitoring policies, the situation detection policies, and the actuation policies. Since an agent can *commit* and *un-commit* commitments dynamically, the applications built on COSAR become extremely configurable and can be adaptive to the enterprise needs, e.g., balanced scorecard, by simply modifying the definitions of NoC through the configuration tools. From the point of view of any stakeholder of the microelectronic manufacturing process, commitments themselves provide a means to define the quality of services from the system itself because the whole system is more visible, controllable, and configurable.

Figure 6 depicts a NoC in COSAR for the use case scenario described in Sect. 2.

The sales agent detects the "out-of-band" situation and notifies the committed agent, i.e., the demand agent that will consequently notify the recommendation agent and risk assessment agent in sequence to obtain recommended build plan(s) and necessary assessments such as inventory cost, manufacturing cost, and SLA measurement. Note that commitment relationships may imply either event/situation flows or data flows between commitment-related agents, and the actions to be taken really depend on the definition of the involved commitments, i.e., on the expectations, actions and responses that are delineated in the committed agents. The aforementioned commitments are all endogenous commitments. However, the portal agent has an exogenous commitment to the decision maker that represents the ultimate user of COSAR.

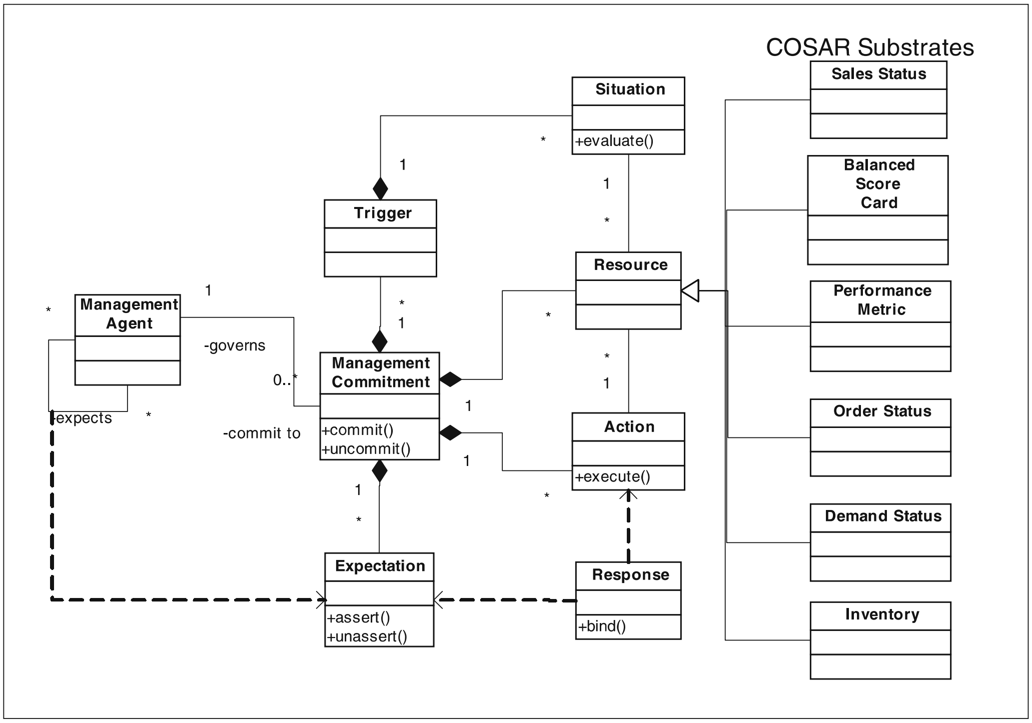


Fig. 5. The meta-model for management commitment

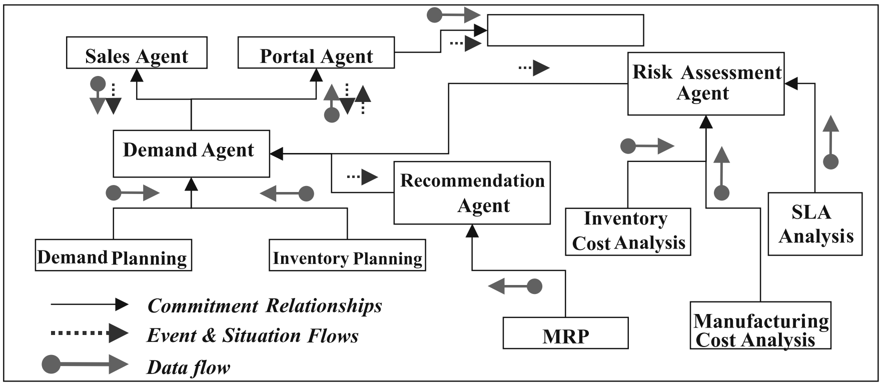


Fig. 6. An example of network of commitments (NoC)

5 The COSAR architecture and implementation status

COSAR is aimed for providing an adaptive platform for microelectronic manufacturing. It is dynamic in the sense that manufacturing applications can be built to recognize the addition and removal of management components and resources without explicit system administrator's intervention. COSAR infrastructure can be logically categorized into several tiers as shown in Fig. 7: COSAR substrates, management probes, management beans, management commitments, management agents, management adaptors, and finally management client.

5.1 Managed resources & management probes

In COSAR, managed resources within the substrates can be any artifacts to be monitored, measured, configured, and controlled. Managed resources are often managed through control points, a set of management APIs that acquire or change the behavior of the managed resources. The states of managed resources can be captured either by polling by management beans or events emitted by managed resources. In many cases of the microelectronic manufacturing domain, management probes are required to provide connectivity from COSAR to managed resources. A management probe is specific to an underlying managed resource. A connection can be either transactional or non-transactional. Examples include a database connection and a SAP R/3 connection. A management probe realizes the connection between a managing module and its targeted managed resources. The management probes and COSAR itself collaborate to provide the underlying mechanisms – transactions, security, and connection pooling. In most scenarios, management probes are used within the same address space as their situated substrates.

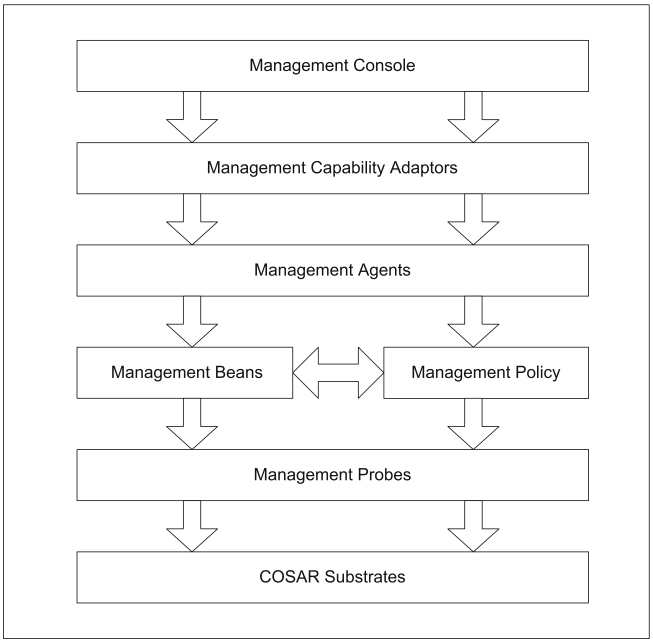


Fig. 7. Layered structure for COSAR

5.2 Management beans

COSAR uses the industrial standard of manageability to instrument managed resources and manufacturing processes. Management standards provide homogeneous interfaces of touch points to managed resources, esp. legacy business systems. Management beans expose the managed properties of underlying managed resources to privileged managing agents. At the run time, the managed resources will be connected to the management bean so that management data and functions can be delegated between management agents and managed resources. The composite of management beans and resource connector form a management façade that serves as a software driver to connect to the managed resources at the agent level. In most situations, the management bean is situated in the address space as the agent, and the managed resource connector is associated with managed resources. Note that the management façade is a logical entity and can be used to define the contracts of connectivity between management agents and those managed resources that are essential to fulfill the management commitments of management applications.

5.3 Management agents

The tier of management agents consists of agents and utilities that can be assembled, composed and committed to provide management functions. Each agent addresses specific needs and problems. Management agents obtain (by pulling or pushing) data from the tier of management beans and act on that data based upon management commitments. The composition of services and components into a purposeful set of functions contained in management agents are enabled by the configuration agent. A management agent is an active object that is situated in the environment of COSAR substrates, and is capable of autonomous actions in this environment to meet management commitments. Developers of management solutions can use COSAR libraries to build a certain amount of intelligence directly into their management agents in order to perform management tasks directly, without human intervention.

5.4 Management capability adaptors

Management capability adaptors expose the management services of management agents to external clients. In COSAR, management connectors adhere to industrial standards such as J2EE/JMX [13] and web services [14]. The design of management beans, management agents, and managed resources does not depend in any way on the protocol an agent uses for communicating with external applications. The provided management capability adaptors rely on the standard APIs and do not expose any communication details. Web services provide a means for different parties to connect their management agents with one another to conduct dynamic management services across a network, no matter what their application, design, or run-time environment. Web services are an incremental addition to a business's existing management platforms. Businesses can rapidly develop and deploy web services for integration of management functions with little or no change in their existing management infrastructure. The web services model is a simple interface for communicating with management agents. Moreover, web services enable management infrastructure to provide the necessary structure to develop an integrated management solution.

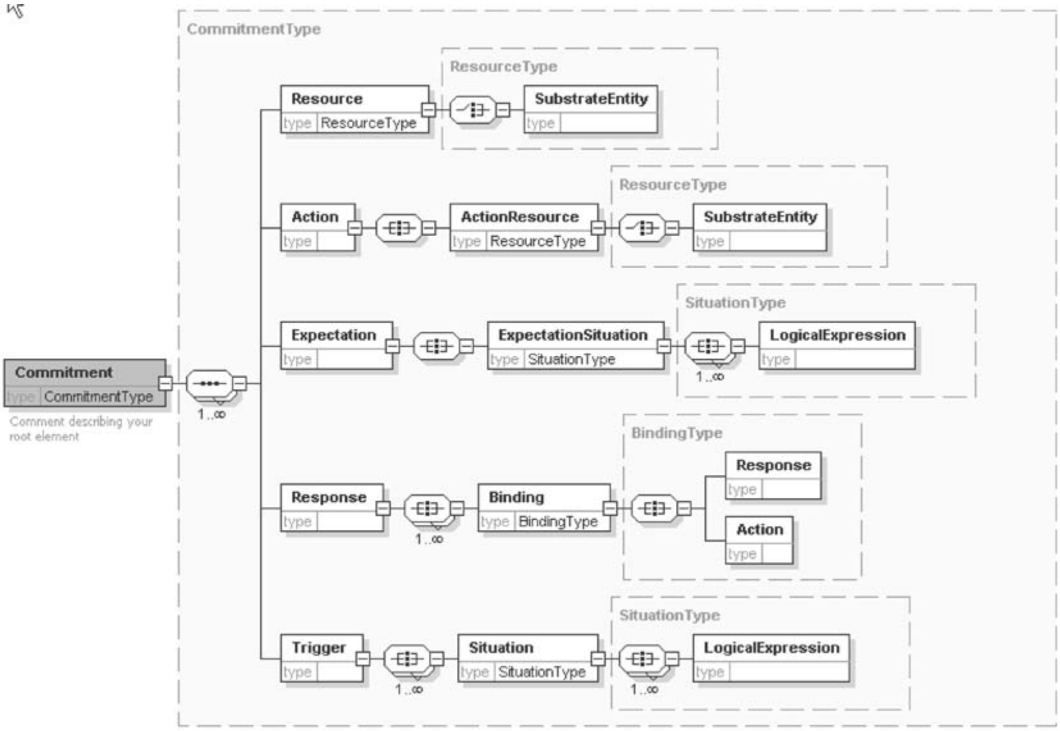
5.5 Management commitments

As mentioned in the previous section, COSAR embraces the style of commitment-based management in that COSAR exploits management commitments as the vehicle to drive management scenarios on managed resources and manufacturing processes. An XML-based language has been developed, called management commitment language (MCL), for describing management commitments. MCL formally describes the concept of business commitments [15]. The XML schema of MCL is shown in Fig. 8.

5.6 An implementation for manufacturing process

We have worked with a chip maker and applied COSAR to fulfill the requirements raised in the electronic manufacturing domain described in this paper. Specifically, we have developed a reference implementation. This system senses events generated in the manufacturing systems, detects manufacturing-related business situations, conducts analysis on the data embedded in the situations and enterprise database, and finally provides recommended actions to decision makers.

Fig. 8. XML schema definition for MCL



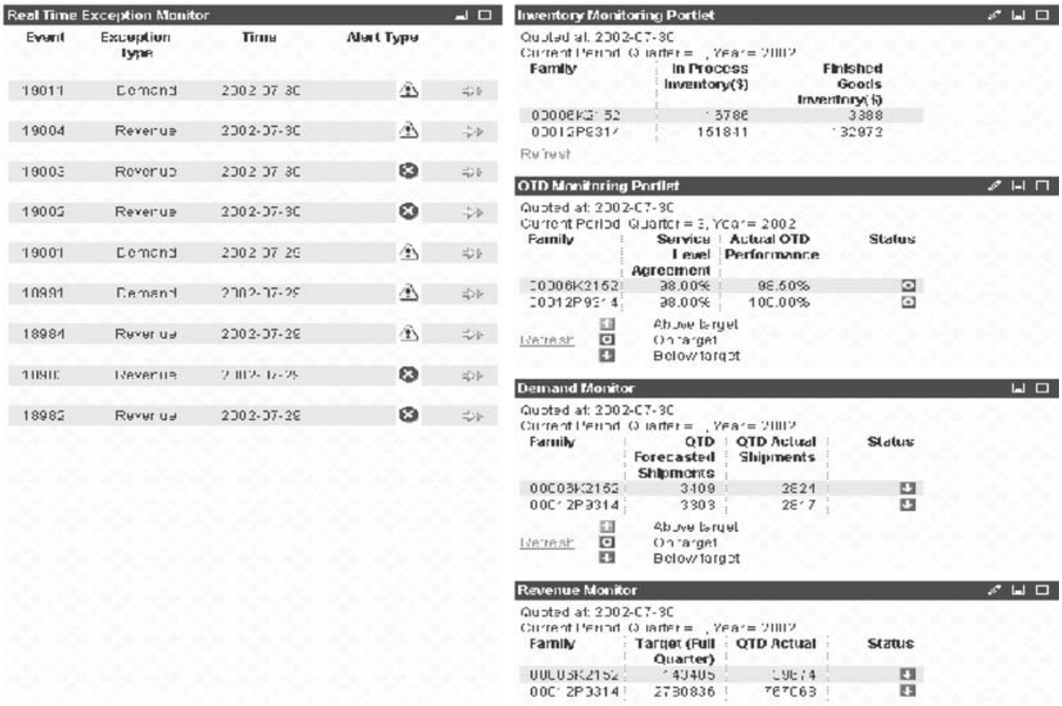
Between the dashboard and the back-end management agents, there is a dashboard façade with the following components: (1) COSAR widgets that are customized for presenting data for specific users. (2) Service agents that are data containers and functional components specialized for the manufacturing domain. Examples of service agents include charting controller, personal alert controller, event controllers, and tag libraries. (3) User agents that serve as the proxies for COSAR users. User agents are connected to the management agents and used to serve the requests from dashboard users (through user agents). This layer consists of adaptors that connect to COSAR artifacts and convert all requests to XML.

Management clients can be in many forms: management console, manufacturing portal, planning system client, OLAP client, and business activity dashboard. The COSAR management console in Fig. 9 presents the unified view for the COSAR users to monitor all the manufacturing processes and activities, manufacturing exceptions, links to perform OLAP analysis, presents recommended actions to manufacturing exceptions and so on. The right-hand side of this console present four monitoring portals and the real monitoring of manufacturing events are shown in the portal on the left-hand side. Figure 10 shows the revenue performance of a manufacturing process. The portal on the left-hand side shows the revenue statistics including both actual and predictive performance data. On the right-hand side, a graphical representation of the performance data is shown in a portal where the upper and lower bounds indicate the performance targets. A business situation will be raised whenever the revenue performance data is out of the boundaries. Thereafter, a decision making process will be triggered to resolve such situation. As mentioned, all of these interactions are handled by designated web services based upon predefined management commitments.

The first phase of the implementation has been finished. The implemented system has been demonstrated to IBM managers and executives in the IBM research division and deployed to the IBM microelectronics division. The implementation platform is based on IBM's software packages: IBM WebSphere Application Server, IBM DB2 Universal Database, IBM WebSphere Portal Server, and other open sources. The execution platforms are both Linux and Windows 2000. The first phase of the implementation took about six months and three research groups in four countries took part in the projects.

The feedback we got from customers and executives has been positive. They thought that COSAR took the right step to realize the sense-and-respond enterprise. Customers were particularly enthusiastic about visual portlets that are provided by the systems. Different widgets for on time delivery (OTD) and serviceability gave the executives exactly what they wanted to monitor about their business. Another important feature that customers appreciated most is the ability of the COSAR system to dynamically modify the relationship among business entities by simply changing the management commitments. Commitments have been regarded as a very useful tool of capturing the contractual relationships that are existing among enterprises or among departments within an enterprise.

At the time of writing, the second phase of the implementation is underway. We would like to focus on providing the OLAP capability through both in-house and third-party software packages. We are also implementing a control system based on the existing COSAR foundation to enable the rendering of management actions on the managed resources. Such a control system



Real Time Exception Monitor

Event	Exception type	Time	Alert Type
19011	Demand	2002-07-30	⚠
19004	Revenue	2002-07-30	⚠
19003	Revenue	2002-07-30	⊗
19002	Revenue	2002-07-30	⊗
19001	Demand	2002-07-29	⚠
18991	Demand	2002-07-29	⚠
18984	Revenue	2002-07-29	⚠
[illegible]	Revenue	2002-07-29	⊗
18982	Revenue	2002-07-29	⊗

Inventory Monitoring Portlet

Quoted at: 2002-07-30
Current Period Quarter = , Year = 2002

Family	In Process Inventory($)	Finished Goods Inventory($)
00006K2152	15786	3388
00012P9314	151811	132972

Refresh

OTD Monitoring Portlet

Quoted at: 2002-07-30
Current Period Quarter = 3, Year = 2002

Family	Service Level Agreement	Actual OTD Performance	Status
00006K2152	98.00%	96.50%	⊡
00012P9314	98.00%	100.00%	⊡

Refresh
⊞ Above target
⊡ On target
⬇ Below target

Demand Monitor

Quoted at: 2002-07-30
Current Period Quarter = , Year = 2002

Family	QTD Forecasted Shipments	QTD Actual Shipments	Status
00003K2152	3409	2621	⬇
00012P9314	3303	2617	⬇

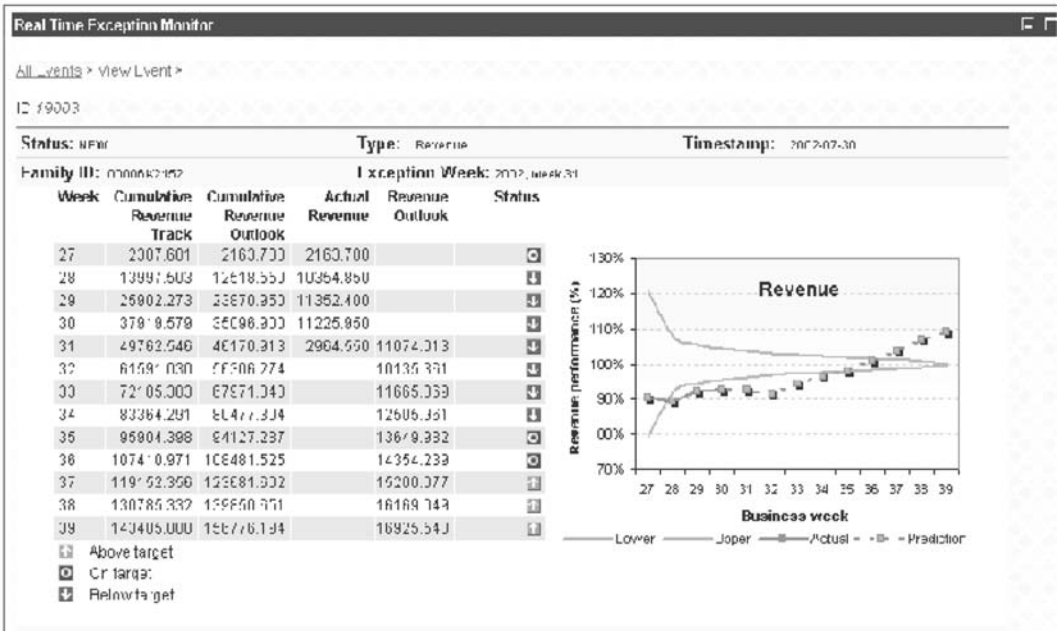
Refresh
⊞ Above target
⊡ On target
⬇ Below target

Revenue Monitor

Quoted at: 2002-07-30
Current Period Quarter = , Year = 2002

Family	Target (Full Quarter)	QTD Actual	Status
00003K2152	143405	[illegible]	⬇
00012P9314	2730836	[illegible]	⬇

Fig. 9. COSAR monitoring console



Real Time Exception Monitor

All Events > View Event >

ID 19003

Status: NEW　Type: Revenue　Timestamp: 2002-07-30

Family ID: 0200062152　Exception Week: 2002, week 31

Week	Cumulative Revenue Track	Cumulative Revenue Outlook	Actual Revenue	Revenue Outlook	Status
27	2307.601	2163.700	2163.700		⊙
28	13997.503	12618.550	10364.850		⬇
29	25902.273	23870.950	11352.400		⬇
30	37918.579	35096.300	11225.950		⬇
31	49762.546	46170.313	2964.550	11074.313	⬇
32	61591.030	56306.274		10135.391	⬇
33	72105.300	67571.340		11665.339	⬇
34	83364.281	80477.334		12606.361	⬇
35	95904.398	94127.237		13679.932	⊙
36	107410.971	108481.525		14354.239	⊙
37	119152.356	123681.332		15200.377	⬆
38	130785.332	139650.951		16169.743	⬆
39	143405.000	156776.194		16925.543	⬆

⬆ Above target
⊙ On target
⬇ Below target

Fig. 10. Monitoring revenue performance

is based on the principles of system dynamics [16]. The enablement of action rendering though system dynamics completes the feedback control loop that is used in the traditional control system. Consequently, the sense-and-respond cycle depicted in Fig. 1 will be fulfilled.

6 Related work and discussion

The supply chain operation reference (SCOR) model provided by the supply chain council defines inter-organizational business processes and their workflows [17]. The SCOR model contains the metrics for operational control and the best practices of the supply chain design. SCOR identifies key performance indicators (KPIs) in five process areas: plan, source, make, deliver, and return. However, it does not have standards for regulating actions when some KPI-based thresholds are exceeded. COSAR, however, addresses this issue by defining management commitments for both manufacturing metrics and management actions.

Chains of commitments architecture was proposed by Ervin et al. [12]. The domain of microelectronic manufacturing that COSAR is aimed for also belongs to the arena of supply chain management. Nevertheless, COSAR is focused on providing mechanism and concrete architecture enabling the representation, evaluation, and enforcement of management commitments. COSAR is an agent-based framework where agents can commit themselves to one another to form a network of commitments that can be used to drive management scenarios.

Similarly, Verdicchio and Colombetti also incorporated commitments into the supply chain domain [18]. They are concerned with the issue of information sharing among agents and proposed a data structure for commitments that can be used for the communication of agent-based framework for the management of a supply chain. However, COSAR is concerned with the holistic infrastructure that can be used for commitment-based management in the domain of microelectronic manufacturing.

Minsky and Ungureanu [19] described a mechanism called law-governed interaction (LGI), which is designed to satisfy three principles: (1) coordination policy needs to be coordinated, (2) the enforcement needs to be decentralized, and (3) coordination policies need to be formulated. COSAR satisfies all of the LGI principles. LGI uses decentralized controllers co-located with agents. LGI does not address the issues such as commitments-driven management processes. There are several future research directions. Since the formation of the "network of commitment" would be a costly and time-consuming task, it is worthwhile to conduct the NoC simulation. It is important to consider many factors during the simulation such as the "level of commitment", "penalty of failing to make the commitment", and commitment escalation. Sect. 3 describes two typical management scenarios, it should be an interesting research issue if all 27 (3 × 3 × 3) possible combinations are contrasted and compared. Third, it is not clear to us whether COSAR can be used in layers other than the first one: real-time response systems.

7 Conclusions

In this paper, we have presented the concepts of commitment and COSAR that enables commitment-based management for microelectronic manufacturing processes. Instead of using traditional integration approaches, COSAR takes commitment as the driving factor of managing manufacturing processes, systems, and data. The consequence is that we have obtained an adaptive framework and architecture for microelectronic manufacturing. With COSAR, the business line manager has better control over the microelectronic manufacturing system since it is highly configurable and easier to work on commitments instead of disintegrated policies. This work can definitely be applied to domains beyond microelectronic manufacturing.

Acknowledgement We would like to show our appreciation to Grace Lin, Harriet Cao, David Cohn, Josef Schiefer, Markus Ettl, Shubir Kapoor, Santhosh Kumaran, Pawan Chowdhary, and John Kearney for their contributions to this project.

References

1. Lin G, Buckley S, Cao H, Caswell N, Ettl M, Kapoor S, Koenig L, Katircioglu K, Nigam A, Ramachandran B, Wang K (2002) The sense-and-respond enterprise. IBM researchers develop integrated SAR system of global value chain optimization, OR/MS Today. http://www.lionhrtpub.com/orms/orms-4-02/frvaluechain.html. (cited at April 2002)
2. Fordyne K (2001) New supply chain management applications provide better customer service: serious gets exciting. J IBM Microelectron, pp 15–19
3. Wu LSY, Hosking, JRM, Doll, JM (2002) Business planning under uncertainty: will we attain our goal?. IBM Research Report RC 16120, 20 Feb 2002
4. The Internet Engineering Task Force (IETF), SNMP Version 3 (snmpv3). http://www.ietf.org/html.charters/snmpv3-charter.html (cited at November 2002)
5. Bumpus W, Sweitzer JW, Thompson P, Westerinen AR, Williams RC (1999) Common information model: implementing the object model for enterprise management. Wiley, New York
6. IBM (2002) Management and manageability: manageability (M12) model core specification, version 1.3. IBM Internal Report
7. Anupindi R, Chopra S, Deshmukh SD, Mieghem JAV, Zemel E (1999) Managing business process flows. Prentice-Hall, New York
8. Buchmann F, Meunier R, Rohnerr H, Sommerlad P, Stal M (1996) A system of patterns: pattern-oriented software architecture. Wiley, New York
9. IBM. Auton computing: IBM's perspective on the state of information technology. IBM Research External Web Site, http://www.research.ibm.com/autonomic/
10. Sloman A (2001) Varieties of affect and the CogAff architecture schema. Proc AISB'01 Symposium on Adaptive Agents and Multi-agent Systems, University of York, United Kingdom 21–24 March 2001
11. Jain AK, Aparico IVM, Singh MP (1999) Agents for process coherence in virtual enterprise. Commun ACM, 42(3):62–69
12. Ervin R (2002) Chains of commitment software architecture. ACM SIGecom Exchanges Special Issues on Chains of Commitment. http://www.acm.org/sigs/sigecom/exchanges/volume_3_(02)/3.1-Ervin.pdf
13. Sun Microsystems Inc. (2000) Java management extensions instrumentation and agent specification. Sun Microsystems Inc, California, USA
14. Graham S, Simeonov S, Boubez T, Daniels G, Davis D, Nakamura Y, Neyama Y (2002) Building web services with Java.SAMS, Indianapolis
15. Jeng JJ, Buckley S, Chang H, Chung J-Y, Kapoor S, Kearney J, Li H, Schiefer Jn (2002) BPSM: an adaptive platform for managing business process solutions. Fifth International Conference on Electronic Commerce Research (ICECR-5), Montreal, Canada, 23–27 October 2002
16. Sterman J (2003) Business dynamics: systems thinking and modeling for a complex world. McGraw-Hill, New York
17. Supply Chain Council (2000) Supply-chain operations reference model, version 3.1. Supply Chain Council. http://www.supply-chain.org
18. Verdicchio M, Colombetti M (2002) Commitments for agent-based supply chain. ACM SIGecom Exchanges Special Issues on Chains of Commitment. http://www.acm.org/sigs/sigecom/exchanges/volume_3_(02)/3.1-Verdicchio.pdf
19. Minsky NH, Ungureanu V (2000) Law-governed interaction: a coordination and control mechanism for heterogenous distributed systems. ACM Trans Softw Eng Methodol, Vol. 9(3):273–305
20. Haeckel SH, Slywotzky, AJ (1999) Adaptive enterprise: creating and leading sense-and-respond organizations. Harvard Business School Publisher, Boston