

By M.P. Papazoglou and D. Georgakopoulos,
Guest Editors

SERVICE-ORIENTED COMPUTING

SERVICE-ORIENTED COMPUTING (SOC) IS THE computing paradigm that utilizes services as fundamental elements for developing applications. SOC involves the service layers, functionality, and roles (see the sidebar “Services Overview”) as described by the extended service-oriented architecture (SOA) depicted in the figure here. Basic services, their descriptions, and basic operations (publication, discovery, selection, and binding) that produce or utilize such descriptions constitute the SOA foundation. The higher layers in the SOA pyramid provide additional support required for service composition and service management.

The service composition layer encompasses necessary roles and functionality for the consolidation of multiple services into a single composite service. The resulting composite services may be used by service aggregators as components (basic services) in further service compositions or may be utilized as applications/solutions by service clients. Service aggregators thus become service providers by publishing the service descriptions of the composite service they create. Service aggregators develop specifications and/or code that permit the compos-

ILLUSTRATION
BY
PETER HOEY



ite service to perform functions that include:

- **Coordination:** Control the execution of the component services, and manage dataflow among them and to the output of the component service (by specifying workflow processes and using a workflow engine for runtime control of service execution).

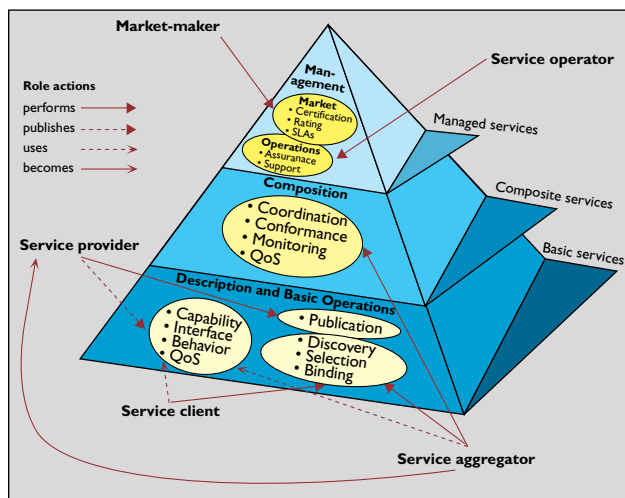
- **Monitoring:** Subscribe to events or information produced by the component services, and publish higher-level composite events (by filtering, summarizing, and correlating component events).

- **Conformance:** Ensure the integrity of the composite service by matching its parameter types with those of its components, impose constraints on the component services (to ensure enforcement of business rules), and performing data fusion activities.

- **Quality of Service (QoS) composition:** Leverage, aggregate, and bundle the component's QoS to derive the composite QoS, including the composite service's overall cost, performance, security, authentication, privacy, (transactional) integrity, reliability, scalability, and availability.

To manage critical applications/solutions and specific markets, SOA provides managed services in the service management layer depicted at the top of the SOA pyramid. In particular, SOA's operations management functionality is aimed at supporting critical applications that require enterprises to manage the service platform, the deployment of services and the applications. Operations management functionality may provide detailed application performance statistics that support assessment of the application effectiveness, permit complete visibility into individual business transactions, and deliver application status notifications when a particular activity is completed or when a decision condition is reached. The organization responsible for performing such operation management functions is known as the service operator, which may be a service client or aggregator, depending on the application requirements.

Extended service-oriented architecture: Service layers, functionality, and roles.



Another aim of SOA's service management layer is to provide support for open service marketplaces. Sev-

Overview of Services

Services are self-describing, open components that support rapid, low-cost composition of distributed applications. Services are offered by service providers—organizations that procure the service implementations, supply their service descriptions, and provide related technical and business support. Since services may be offered by different enterprises and communicate over the Internet, they provide a distributed computing infrastructure for both intra- and cross-enterprise application integration and collaboration.

Service descriptions are used to advertise the service capabilities, interface, behavior, and quality. Publication of such information about available services provides the necessary means for discovery, selection, binding, and composition of services. In particular, the service capability description states the conceptual purpose and expected results

of the service (by using terms or concepts defined in an application-specific taxonomy). The service interface description publishes the service signature (its input/output/error parameters and message types). The (expected) behavior of a service during its execution is described by its service behavior description (for example, as a workflow process). Finally, the Quality of Service (QoS) description publishes important functional and nonfunctional service quality attributes, such as service metering and cost, performance metrics (response time, for instance), security attributes, (transactional) integrity, reliability, scalability, and availability. Service clients (end-user organizations that use some service) and service aggregators (organizations that consolidate multiple services into a new, single service offering) utilize service descriptions to achieve their objectives. **C**

eral vertical industry marketplaces currently exist, including those for the semiconductor, automotive, travel, and financial services industries. Open service marketplaces operate much in the same way as vertical marketplaces, however, they are open. Their purpose is to create opportunities for buyers and sellers to meet and conduct business electronically, or aggregate service supply/demand by offering added-value services and grouping buying power (similar to a co-op). The scope of such a service marketplace would be limited only by the ability of enterprises to make their offerings visible to other enterprises and establish industry-specific protocols by which to conduct business.

Open service marketplaces typically support supply chain management by providing to their members a unified view of products and services, standard business terminology, and detailed business process descriptions. In addition, service marketplaces must offer a comprehensive range of services supporting industry trade, including services that provide business transaction negotiation and facilitation, financial settlement, service certification and quality assurance, rating services, service metrics such as number of current service requesters, average turnaround time, and manage the negotiation and enforcement of service level agreements (SLAs). SOA's service management layer includes market management functionality, as illustrated in the figure, which is intended to support these marketplace functions. The marketplace is created and maintained by a market-maker (a consortium of organizations) that brings the suppliers and vendors together. The market-maker assumes the responsibility of marketplace administration and performs maintenance tasks to ensure the administration is open for business.

The application of SOC on the Web (including several aspects of the SOA) is manifested by Web services. A Web service is a specific kind of service that is identified by a URI, whose service description and transport utilize open Internet standards. Interactions between Web services typically occur as SOAP calls carrying XML data content. Interface descriptions of the Web services are expressed using Web Services Definition Language (WSDL). The Universal Description, Discovery, and Integration (UDDI) standard defines a protocol for directory services that contain Web service descriptions. UDDI enables Web service clients to locate candidate services and discover their details. Service clients and service providers utilize these standards to perform SOA's basic operations shown in the figure. Service aggregators may use the Business Process Execution Language for Web Services (BPEL4WS) to create new Web services by defining corresponding compositions of the interfaces and

SINCE SERVICES MAY
BE OFFERED BY
SPECIFIC PROTOCOLS
AND COMMUNICATE
OVER THE INTERNET,
THEY PROVIDE A
DISTRIBUTED COMPUTING
INFRASTRUCTURE FOR
BOTH INTRA- AND
CROSS-ENTERPRISE
APPLICATION INTEGRATION
AND COLLABORATION.



SOA'S OPERATIONS
MANAGEMENT
FUNCTIONALITY IS AIMED
AT SUPPORTING CRITICAL
APPLICATIONS THAT
REQUIRE ENTERPRISES TO
MANAGE THE SERVICE
PLATFORM, THE
DEPLOYMENT OF SERVICES,
AND THE APPLICATIONS.



internal processes of existing services.

This section focuses on Web services, particularly issues that relate to Web service standards, their formal aspects, composition approaches, transactions involving Web services, and the application of Web services in business solutions. In the first article, Curbera et al. describe how Web services are evolving from the basic operations in the SOA foundation layer to support robust business interactions in the composition layer. The authors explain the capabilities currently enabled by the specifications of BPEL, WS-Coordination, and WS-Transaction, illustrating how they can be used together in the composition layer.

Yang introduces the concept of a service component that raises the level of abstraction in Web service compositions. Service components represent modularized service-based applications that associate service interfaces with business logic into a single cohesive conceptual module. Service components can be extended, specialized, and generally inherited, to facilitate the creation of applications.

Meredith and Borg also concentrate on the SOA composition layer and examine the complexity problem of distributed heterogeneous applications (assuming that service connectivity has been addressed). They propose a formal approach based on the development of type systems for the specification and automatic verification of crucial properties of service behavior.

Transactions involving Web services are examined in the article by Little. Such transactions differ from traditional transactions in that they execute over long periods, they require commitments to the transaction to be negotiated at runtime, and isolation levels must be relaxed. Little further describes an extended transaction model that builds on existing Web service standards and defines an interoperable transaction protocol and message flows that negotiate cross-enterprise transactional guarantees at the composition layer.

In the final article, Casati et al. shift attention to the management layer of the SOA and more specifically to operations management. The proposed business-oriented management of Web services is an attempt to assess the impact of service execution from a business perspective and, conversely, to adjust and optimize service executions based on stated business objectives. This is a crucial issue as corporations strive to align service functionality with business goals. **C**

M.P. PAPAIOGLOU (mikep@uvt.nl) is a professor at Tilburg University in The Netherlands.

D. GEORGAKOPOULOS (dimitris@research.telcordia.com) is a senior research scientist at Telcordia in Austin, TX.
