

# Gartner's Roadmap to SOA

SOA is a software architecture style that uses server-side, consumer-independent business components, accessed in an interactive manner via documented remotely accessible programmatic interfaces. It's modular and distributable.

### How Much of Your Architecture Will SOA Cover?

Tactical Guideline: The development of the future-state architecture involves three main activities, each of which produces a type of artifact. The three activities are "develop requirements," "develop models" and "develop services".

An Enterprise Architecture Contains Multiple Viewpoints	Each Viewpoint Contains Multiple Levels of Abstraction	Each Level Contains Multiple Types of Artifacts
<b>Viewpoints</b> n Business n Information n Technology n ...	<b>Levels</b> n Conceptual (SOA) n Logical n Implementation n ...	<b>Types of Artifacts</b> n Requirements n Models n Service Specifications n Service Implementations

Too much SOA can lead to paralysis and indecision. Balance requires small bites, not big gulps.

Action: Using a top down approach, develop requirements, models and services at all levels of abstraction to describe the desired future state in increased levels of detail with increasingly detailed guidelines.

### What Business Benefits will SOA Deliver?

Strategic Planning Assumption: Through 2007, the upfront investment for large-scale service-oriented applications will be justifiable only for projects with a planned lifetime of three years or more (0.8 probability).

Benefits	Implications
<b>Architectural Partitioning</b> n Diverse life cycle "speeds" n Synergy of different technologies n Optimal tech skills allocation n Processes visibility n Greater maintainability n Easier outsourcing/offshoring <b>Incremental Deployment</b> n Gradual migration n Cost "spreading" across projects n Reduced maintenance cost <b>Reuse of Services:</b> n Faster time to deployment n Lower development cost n Greater adaptability	<b>Higher Upfront Costs</b> n Cultural change n Infrastructure (SOA backplane) n More formal methodology n Longer design time for services n Testing (unit/end-to-end) <b>More Distributed Infrastructure</b> n Extensive use of middleware n Transaction management n Debugging/troubleshooting n End-to-end management n More granular security n Metering/logging <b>Tighter Management/Governance</b> n Ownership/accountability n Cost allocation n Prioritization/conflict resolution

Action: Users should consider SOA only for the most strategic, systematically oriented projects.

### How Should Organizations Apply Technology to SOA Projects?

Strategic Planning Assumption: Through 2008, users will have a choice of many competing products, each of which is suitable for a specific type of SOA project.

Application (project type)	Data-Facing	
	User-Facing	Data-Facing
New	<b>Portal Product</b> BEA, BroadVision, CA, Day, Fujitsu, Hummingbird, IBM, Microsoft, Oracle, SAP, Sun Microsystems, Tibco, Vignette, webMethods	<b>Application Server</b> Applety, BEA, Borland, Caucho, Fujitsu, IBM, Iona, JforFlare, JBoss, Kaltura, Kinogee, Adobe, Microsoft, ObjectWeb, Oracle, Tmax, Paremus, Pramati, SAP, Sun, Sybase, Zend, Magic Software, Intersystems, Microsoft
	<b>Application Platform Suite</b> BEA, Cordys, Fujitsu, Hitachi, IBM, JBoss, Microsoft, ObjectWeb, NEC, Oracle, SAP, Sun, Sybase	<b>Application Platform Suite</b> BEA, Cordys, Fujitsu, Hitachi, IBM, JBoss, Microsoft, ObjectWeb, NEC, Oracle, SAP, Sun, Sybase
50-50	<b>Presentation Integration Server</b> Attachmate, Attunity, F&P, FS, GT, HostBridge, IBM, Jaccada, Microsoft, Neen, NetManage, OpenConnect, Red Oak, Seagull, SAG, Tivoli	<b>ESB, Integration Suite</b> Akway, BEA, Cape Clear, Cordys, Extol, Fiorano, Fujitsu, IBM, Intersystems, Iona, Way, Magic, Microsoft, Oracle, Polaris, SAG, SAP, Sonic, Sterling, Sun, Sybase, Tibco, Vitrina, webMethods
	<b>Application Platform Suite</b> BEA, Cordys, Fujitsu, Hitachi, IBM, JBoss, Microsoft, ObjectWeb, NEC, Oracle, SAP, Sun, Sybase	<b>Application Platform Suite</b> BEA, Cordys, Fujitsu, Hitachi, IBM, JBoss, Microsoft, ObjectWeb, NEC, Oracle, SAP, Sun, Sybase

Action: Choose a specialist platform product for focused SOA projects. However, for broad, demanding SOA projects (covers most areas on this chart), a specialist product can lead to cost overruns or underpowered applications.

### What are the SODA Hype Cycle Components?

Strategic Planning Assumption: Through 2009, organizations that incorporate SODA principles in development of SOBAs for internal and external consumption will achieve process efficiency increases of at least 15 percent (0.8 probability).

Action: Replace monolithic applications with component-based processes that can be combined, purchased and accessed according to real-time needs.

### How Should Organizations Plan the Transition to SOA?

Strategic Planning Assumption: Through 2011, enterprises will experience a continuing increase in complexity and cost of their SOA and SOA 2.0 projects (0.8 probability).

2006	Infrastructure-focused, basic small service delivery, introduction to composition
2008	Large number of services, compositions, transactions and micro-flows. In-house developed and vendor provided services, repositories and directories
2010	Events and services, long-running business processes, role establishment business strategy approach, architecture maturity

Action: Schedule projects assuming that client/server SOA will be increasingly well-supported in the next five years by vendor tools. Complex event-driven architecture will remain requiring advanced in-house architecture efforts. The full benefit of SOA will not be achieved until it is combined with event-driven principles.

### How Will Organizations Adopt Software as a Service?

Strategic Planning Assumption: Through 2009, fewer than 5 percent of business applications will be deployed in a hosted or on-demand model (0.8 probability).

Strategy	1st Basic Competence	2nd Functional Effectiveness	3rd Intraenterprise Integration	4th Business Ecosystem
Technology	n Departmental or siloed n Baseline capabilities n Limited to no configurational customization n Limited to no integration	n Departmental or siloed n Competitive differentiation n Configuration/customization required n Integration with apps/data	n Cross-departmental n 360° view of customer, supplier, etc. n Business process modeling n Master data integration (internal)	n Collaborative processes among multiple enterprises n Broad availability of application services n Data integration (external)
Adoption:	90%	10%	0%	0%
2005	70%	20%	8%	2%
2007	40%	30%	20%	10%
2010				

Action: Achieve basic competence with on-demand services before moving to more-complex generational capabilities.

## SOA Maturity Model and Key Features

	Level 1 - Introduction	Level 2 - Spreading	Level 3 - Exploitation	Level 4 - Plateau
<b>Business Goal</b>	Specific Pain Points (e.g., customer portal)	Process Integration (e.g., B2B)	Process Flexibility (e.g., time to market)	Continuous Adaptation and Evolution
<b>IT Goal</b>	Proof of Concept	Establish Technology Platform	Leverage Services Reuse	Scale Up
<b>Scope</b>	Single Application	Multiple Applications (single BU)	Multiple Applications (cross BUs)	Virtual Enterprise
<b># of Published Services</b>	Less than 25	Less than 100	Less than 500	More than 500
<b># of Service Consumers</b>	Less than 5	Less than 25	Less than 50	More than 50
<b>Skills</b>				
n Basic middleware	✓	✓	✓	✓
n Web services	✓	✓	✓	✓
n Integration middleware	✓	✓	✓	✓
n Business process management		✓	✓	✓
n Service oriented development of applications (SODA)			✓	✓
<b>Capabilities</b>				
n Integration competency center		✓	✓	✓
n Service registry			✓	✓
n Service design and management methodologies			✓	✓
n Planning, control and quality management			✓	✓
n Ownership and accountability policy			✓	✓
n Service reuse policy			✓	✓
n SOA domains			✓	✓
n Chargeback / cost allocation policy			✓	✓
n Formal SOA governance			✓	✓
n Enterprise nervous system strategy			✓	✓

## Gartner's SOA Position & Definitions

**Gartner's Position:** Service-oriented architecture will shift developer focus from software functions to business functions, thereby transforming installed software from an inhibitor to a facilitator of rapid business change. Realizing these benefits will, however, require increased investment in software, infrastructure, skills and business process change. SOA will become the dominant framework for creating and delivering software, shifting value from packaged software to subscription services and from monolithic suites to composite applications.

**Service:** A service component is an executable piece of software that offers a well-described interface that can be accessed by other independently developed and operating pieces of software. This access can be achieved through a process of runtime discovery — that is, the user of the service need have no prior knowledge of the service and its interface definition before it seeks to access the service. A service is more specifically defined as a Web service when the interface is defined using WSDL and is accessed via SOAP over HTTP. The principles of SOA are not new. The development of Common Object Request Broker Architecture (CORBA) was an important step in the evolution of SOA. Web services are the latest realization of standards-based SOA.

**Software as a Service (SaaS):** Software that is owned, delivered and managed remotely by one or more providers. The provider delivers an application based on a single set of common code and data definitions, which are consumed in a one-to-many model by all contracted customers, at any time, on a pay-for-use basis, or as a subscription based on usage metrics.

**UDDI:** To support discovery, the Universal Description, Discovery and Integration (UDDI) standard defines a directory to look up services given a list of characteristics. A range of other standards allows common understanding of attributes, such as security and transaction controls, when supplying or using a service.

**Service Oriented Business Applications:** SOBAs are business applications structured (in whole or part) using an SOA. SOBAs use Web service standards for Web-based messaging, application access and interfacing, and business process transactions. The majority of initial SOBAs are limited transformations of established applications that provide access to some functionality via service interfaces. New SOBAs will be constructed purely within an SOA. As services with standardized interfaces become more widely available, application functionality will be delivered by combining services to meet the needs of a particular user organization. These services may come from multiple suppliers.

**Service Oriented Development of Applications:** Application development within an SOA is known as SODA. Using services as the primary unit of modularity requires a new approach that involves composing applications from sets of loosely integrated processes. This is an "assemble first" approach, rather than the "code first" approach used in traditional software development. An integrated service environment (ISE) is a suite of integrated development tools and technologies used for building service-oriented and composite applications.

## SOA Adoption Strategies

SOA and services will have an impact on every business and IT department. However, there are a number of very different strategies for the adoption of SOA.

<b>Minimalist</b>	Adopt by default as application and tool vendors shift products to SOA, but take no proactive steps to exploit these capabilities until they are pervasive. This is a viable approach, particularly for smaller companies, but beware the danger of getting too far behind the skills curve. This could force dependencies on external service providers or result in low-quality internally generated SOA and Web services applications in the future.
<b>Integration-Focused</b>	Use services to ease integration of internal systems. Currently, this is the most common approach for active adoption of services. It serves a useful purpose in building technical skills, but the business benefit is likely to be limited, and may be more easily achieved using established application integration products. A narrow focus on integration may distract from opportunities to use SOA in a more differentiating way and delay addressing the governance and enterprise architecture challenges that will emerge as SOA becomes strategic.
<b>Externally-Focused</b>	Identify external system interfaces used by business partners and customers and offer those as web services. The value of this approach depends on adoption by other parties. In some cases, the service may be sufficiently compelling that it will drive adoption; in others, a consortium approach will be needed to drive mutual benefits.
<b>Radical</b>	Aggressively seek out opportunities to exploit SOA and services, perhaps including charging for access. The most-obvious candidates for this approach are high-tech companies, particularly those seeking to promote business models based on software by subscription, and processes in industries such as financial services with large volumes of relatively simple and standardized data to publish. However, as with the original shift to Web-based systems, there will be opportunities in many industries for companies to expose business processes in new ways, not simply improving their ability to interface with partners, but changing the way in which they do business, and thereby gaining competitive advantage.

Action Item: Choose between these strategies by assessing and comparing the potential for business benefit vs. investment in skills and processes

## Service Oriented Development Styles

Strategic Planning Assumption: Through 2010, more than 60 percent of services will be composites of new and "old" data and logic (0.8 probability).

**Action Item:** Use the best-available software engineering skills for designing service implementations.

## SOA Futures

### How will SOA enable new business process flexibility?

Strategic Planning Assumption: By 2008, service-oriented development of applications (SODA), SOBAs, Web services, orchestration tools and business process fusion will coalesce, enabling businesses to define their own business process platforms (0.8 probability).

Action Item: Users should seek an integrated platform, comprised of a business services repository and composition technologies, to enable the business process platform.

### How will SOA architectures evolve?

Background/Tutorial: Advanced SOA and modern integration technologies change the structure of application architecture, revising the meaning of the term "application system" and morphing the role of the enterprise network from low-level pipe to intelligent integration fabric.

Action Item: Business executives and IT leaders must understand the implications of an integration fabric because those who do not realize that they are implementing one will acquire a bad one through "accidental architecture."

### How will SOA morph to Advanced SOA?

Definition: Event Driven Architecture is fully asynchronous and uncoupled, partially asynchronous request/reply service-oriented architecture (SOA) is loosely coupled, and the combination of the two results in the flexible and adaptable application architecture of Advanced SOA.

Action Item: To derive the maximum value from SOA, use a mix of classical request/reply service-oriented SOA relationships and uncoupled, event-driven SOA relationships (EDA).