2009

Mapping the Domain of Service Science

Steven Alter
University of San Francisco, alter@usfca.edu

Mapping the Domain of Service Science

Steven Alter
University of San Francisco
alter@usfca.edu

ABSTRACT

The emerging discipline of service science currently lacks coherence because it calls on knowledge from many disciplines and covers topics ranging from services involving human interaction and discretion through invisible services that are hidden in computerized infrastructures. This paper explains the service domain framework, which is designed to help in understanding, analyzing, and researching service topics across the entire domain of service science. This framework is presented as four concentric layers, with the inner layer most closely related to specific service processes and activities, and each of the other layers successively broader in scope and further from action related to specific services. Figures in the paper illustrate the location of topics from different disciplines, synergies between quadrants, links within layers, the location of service-dominant logic, the location of various aspects of SaaS, and the path for bypassing the gap between human and machine services.

Keywords (Required)

Service science, SSME, service domain framework, work system, service system

SEEKING COHERENCE IN A BROAD DOMAIN

The recent groundswell of attention to services and the service economy is relevant to IS/IT for many reasons, including the fact that over half of the revenues of technology firms come from services. (Wood, 2007) Services are important to IS/IT executives because over 75% of the U.S. economy is in the service sector and because IS/IT groups in all economic sectors produce internally directed services for their own firms. Part of the relevant context is a concerted effort by IBM and other leading technology companies to encourage development of a new service science (Chesbrough and Spohrer, 2006; Spohrer et al, 2007) and university degree programs in SSME (service science, manufacturing, and engineering).

The groundswell of attention sometimes glosses over major disagreements about the domain of service science. As an example, here is a list of typical areas of interest related to services within the IS field:

- Impact of IT on service economies
- Human-intensive services for people and organizations
- Services provided by IT organizations
- Software as a service
- Service oriented architecture (SOA)
- Service computing
- Web services

It is unclear whether there is any commonality between all of these areas other than the use of the word *service*.

Merely defining services in the IS field is problematic due to the vastly different connotations of services provided by people versus by computers. Researchers and practitioners in marketing and operations assume that services involve human service providers and human service consumers, both of whom may use IT while performing or receiving the service. The quality of interactions between service providers and service consumers is usually considered quite important, and often viewed as the essence of service, e.g., Carlzon's (1989) term "moments of truth" and Teboul's (2006) book *Service is Front Stage*. Within this view of service, the provider's ability to recognize and respond to the consumer's stated or unstated needs, interests, and concerns is an important aspect of service quality.

In contrast, computer scientists tend to view service within the paradigm of client-server computing, whereby a client entity poses an unambiguous request to a server entity, which produces an unambiguous response. The client and server are machines that interact through definable IT-based interfaces. Neither the client nor the server has the capability of discerning unstated needs, interests, or concerns, methods used by the other, or anything else that is not included in explicitly coded messages governed by the requirements of the interface. Statements in *IBM Systems Journal* illustrate this paradigm:

> A service "is generally implemented as a course-grained, discoverable software entity that exists as a single instance and interacts with applications and other services through a loosely coupled (often asynchronous), message-based communication model." (Brown et al, 2005)
>
> "The component that consumes business services offered by another business component is oblivious to how the provider created the business service." (Cherbakov et al, 2005)

Many IT-based services combine both types of approaches. For example, despite the highly automated nature of some services for operating computer centers and computer operations, personal interaction is often an important determinant of user satisfaction, for example, as measured by the empathy dimension in SERVQUAL.

Goal and organization. This paper's main contribution is the service domain framework, (Figure 1), a framework for understanding, analyzing, and researching service topics across the entire domain of service science. That framework encompasses a broad range of topics that have been covered by many authors with a variety of goals. Figure 2 shows specific disciplines associated with specific areas of the map. Although many service topics have been explored in great depth in specific disciplines, e.g, service-oriented architectures within computer science, to the author's knowledge no one has successfully combined topics from the full range of service science into a single framework that is useful for locating and comparing topics and concepts across the entire domain of service science.

This paper's approach to mapping the domain of service science builds on an AMCIS 2008 paper (Alter, 2008b) that compared and searched for synergies between four lenses for understanding and analyzing service systems. The four lenses were: IT-reliant work systems, co-creation of value, outputs of IT-based tools, and services computing.

The service domain framework is presented as four concentric layers, with the inner layer most closely related to specific service processes and activities, and each of the other layers successively broader in scope and further from action related to specific services. The innermost layer of the new service domain framework, the action layer, is a slightly updated version of the four-lens framework from Alter (2008b). The other three layers are the architecture layer, economic exchange layer, and industry and society layer.

The service domain framework is oriented around two axes that create four quadrants in the inner layers. The vertical axis distinguishes sociotechnical versus primarily technical; the horizontal axis, production orientation versus co-production orientation. The two top quadrants encompass sociotechnical services delivered by people, often with extensive help from IT. The two bottom quadrants encompass primarily technical services delivered by computerized devices. The left two quadrants typically focus on production activities and resources of service providers, whereas the right two quadrants focus on co-production by service providing entities and service consuming entities.

The usefulness of the service domain framework depends primarily on two factors: (1) whether the framework covers all service processes and all research under the general umbrella of service science and (2) whether the framework helps in locating, comparing, understanding, and recognizing synergies and conflicts between service science topics.

Possible uses of the framework start with visualizing where specific concepts or theories reside within the larger domain. That could be helpful in extending the previous paper's attempt to find synergies between alternative lenses for understanding and analyzing services. A specific concept or theory located in a specific place is not closely related to topics located in distant parts of the framework. If the location of a specific topic proves unclear, then that topic probably requires more careful definition. The framework can also help in interpreting statements of the general form, "services are the wave for the future," or "service-oriented architectures will provide great competitive advantage." In the former example, the inability to localize the statement in the framework reflects the statement's vagueness. In the latter example, the long distance in the framework between service-oriented architectures and business organizations and competition in the visible economy implies that the statement requires a lot of explaining, especially in regard to what needs to happen in the action layer in order to realize operational benefits that produce the claimed competitive impacts.

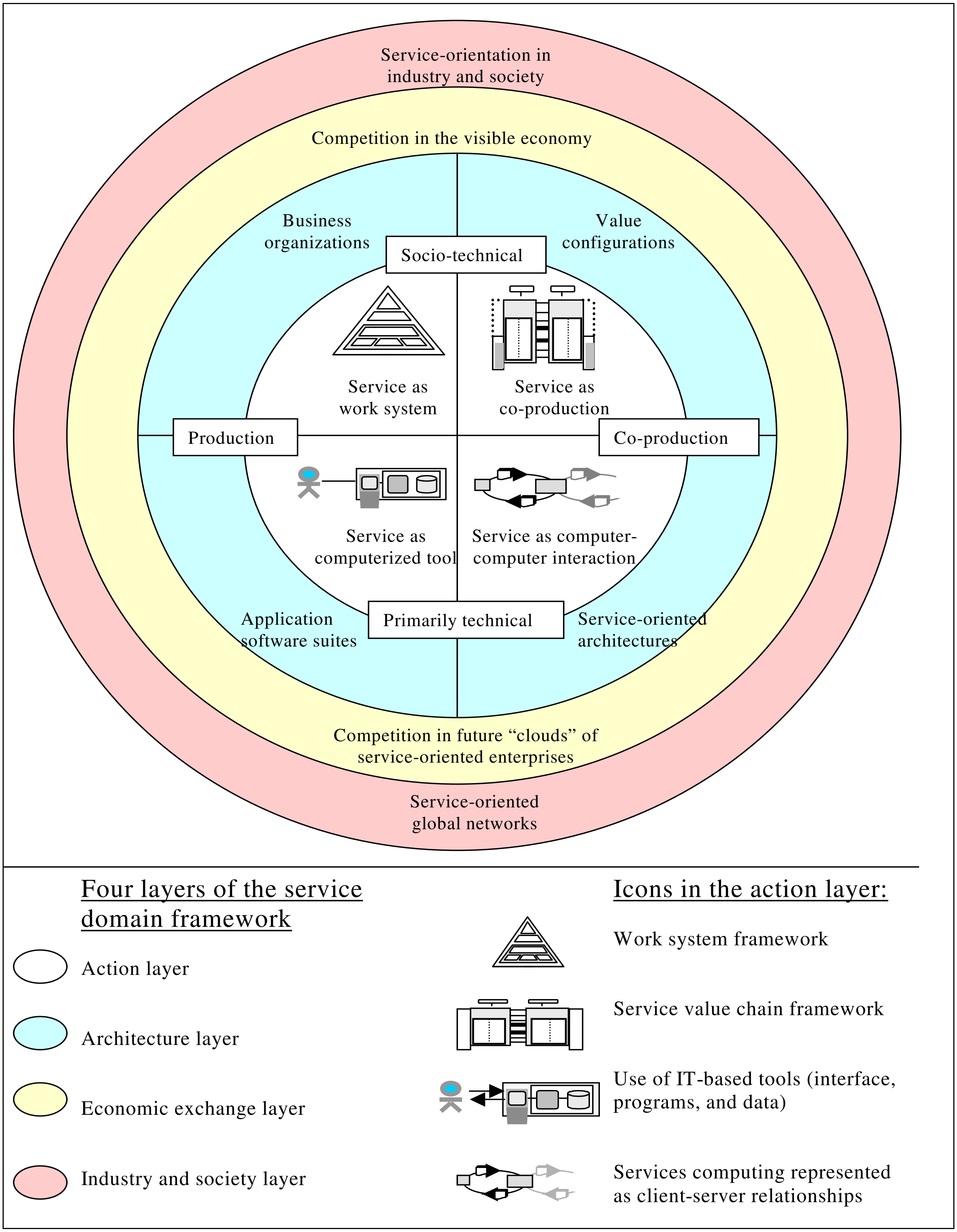


Figure 1: Service domain framework for understanding, analyzing, and researching services

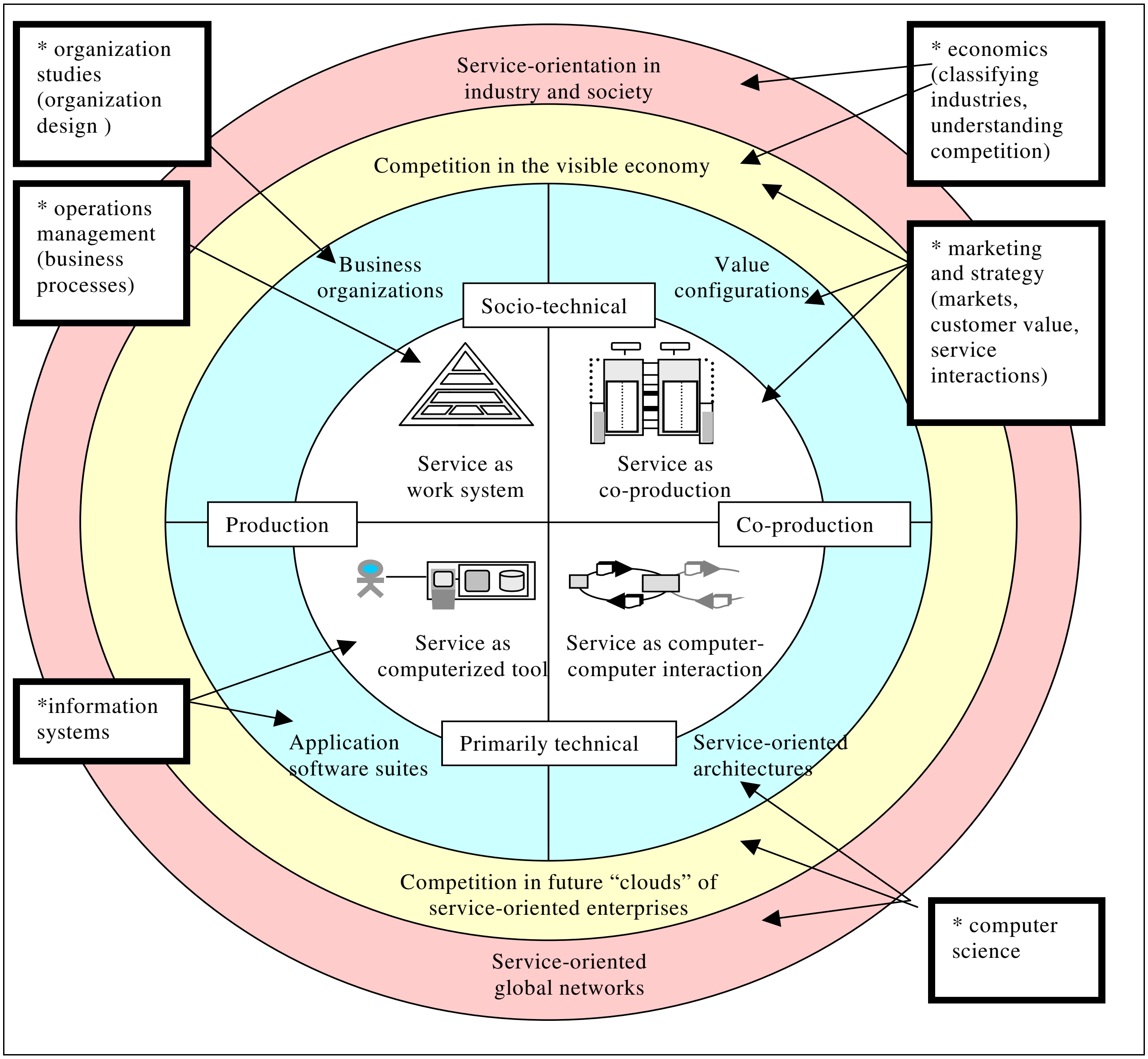


Figure 2: Disciplines often associated with particular areas of the service domain framework

Organization of this paper. This paper's main goal is to summarize the service domain framework (Figure 1) and to demonstrate its potential usefulness in mapping disparate ideas that the existing service science literature has not yet mapped in a comprehensive and useful way. The remainder of the paper defines service, summarizes the four layers of the service domain framework, and provides examples illustrating how the framework can be used.

Length limitations make it impossible to say a great deal about any particular area of framework, or to cite more than a few authors whose work is relevant to specific parts of the framework. Instead, consistent with its exploratory nature, the paper focuses on summarizing the framework and providing a series of illustrative examples.

DEFINITION OF SERVICE

Any framework related to service should be premised on a definition of service, a topic that has been discussed at length by many authors. (e.g., Sampson and Froehle, 2006) This paper's approach to service starts by assuming that every purposeful action performed for the benefit of someone else can be viewed as a service independent of whether the result is customized, intangible, or consumed as produced (characteristics often associated with services). For example, in the *Harvard Business Review* article "Marketing Myopia," Leavitt (1960) noted, "People don't buy a quarter-inch drill. They buy a quarter-inch hole. You've got to study the hole, not the drill. The drill is just a solution for it." According to Vargo and Lusch's (2004) "service-dominant logic," value to the customer is the primary issue; whether that value is delivered through goods or services per se is secondary. "Goods are distribution mechanisms for service provision." (p. 8) Thus, distinctions between products and services may not be fundamental for understanding how value is delivered. If one defines service is an act performed for someone else, then the production of physical things can be viewed as services.

We adopt a simple, dictionary-like definition (Alter, 2008b) by which the following are all services: performing surgery, installing networks, producing customized software, providing Internet-based search capabilities, accepting orders through an ecommerce web site, building houses, producing televisions, providing leisure opportunities on golf courses, performing legal work, and selling groceries.

Services are acts performed for someone else, including the provision of resources that someone else will use.

This definition covers special cases such as self-service and automated services for people. In self-service, service providers provide resources that are used by customers performing self-service activities. In automated services for people, machines perform the service activities.

In the realm services computing "another entity" replaces "someone else."

Services are acts performed by one entity for another, including the provision of resources that another entity will use.

--- --- ---

The next four sections discuss each layer of the service domain framework in turn.

LAYER #1: THE ACTION LAYER

The action layer is an updated version of the four-lens framework in Alter (2008b). The four quadrants of the action layer include:

- Service as work system (sociotechnical with production orientation)
- Service as co-production (sociotechnical with co-production orientation)
- Service as a computerized tool (technical with production orientation)
- Service computing (technical with co-production orientation)

Brief discussions of each of these four quadrants include citations of readily available references that provide much more detail.

Service as work system. This involves thinking of a situation as a work system in which human participants use information, technology, and other resources to produce products and services for internal or external customers.

The triangular icon in the upper left quadrant of the action layer of Figure 1 represents the work system framework, which emphasizes business rather than IT concerns and was developed to help business professionals recognize and understand IT-reliant systems in organizations. That framework identifies nine elements included in even a rudimentary understanding of a work system. Inclusion of customers as the element at the top of the framework implies that work systems can be considered service systems. Almost all significant sociotechnical systems in today's business world are IT-reliant and therefore fall within the scope of the IS field. For detailed discussions of the work system framework see Alter (2003, 2006, 2008a, 2008c)

Service as co-production. The bilateral icon in the upper right quadrant of the action layer represents the service value chain framework, which emphasizes thinking of a situation as co-production or co-creation of value by service providers and service consumers (customers). The assumption of co-production implies that activities and responsibilities of both providers and customers should be included in the description and analysis of a particular situation.

The service value chain framework is a generic value chain model that highlights components of service. Its form and content incorporate ideas often associated with services, such as: co-production of value, internal and external customers, customer experience, negotiated commitments and service level agreements, preparation prior to service instances, service requests, front stage and backstage, follow-up after service instances, and value capture. The motivation and terminology of that framework were explained in Alter (2008a, 2008b).

Service as a computerized tool. The icon in the lower left quadrant of the action layer of Figure 1 represents a user issuing a command to a computerized tool through a user interface and then receiving a response. In addition to a user interface, such tools often contain or link to databases or models. Uses of IT artifacts constitute the predominant focus of the IS field in general and HCI in particular. This focus emphasizes how the form and nature of IT artifacts, including user interfaces, affect the usage of those artifacts. Its disadvantages include deemphazing the business situation (the work system) and assuming that the topic of greatest significance is the usage of technology rather than the successful accomplishment of business process goals.

Service computing. The icon in the lower right quadrant of the inner layer of Figure 1 represents service computing which is organized and described around client programs or devices that request services from server programs or devices. The requests and responses are formal messages expressed using unambiguous formats. The icon for services computing starts with a request sent from the client to the server, which may request information from other devices in the background. Eventually the server provides the requested data or confirms that the request was carried out (as in printing). In other words, service computing is based on "(a) a collection of services that communicate with one another; (b) the services that are self-contained and do not depend on the context or state of other services; and (c) the services that work within a distributed system architecture." (Zhao et al 2008)

Synergies within the action layer. Figure 3 highlights the synergies between different quadrants in the action layer. Strong synergies between quadrant A and quadrant B implies that someone using the viewpoint and analysis tools of quadrant A can usually benefit from thinking about the same situation in terms of ideas from quadrant B, and vice versa. The arrows in Figure 3 represent the likely strength of specific synergies, which were explained in Alter (2008b).

LAYER #2: THE ARCHITECTURE LAYER

In relation to systems and services, architecture can be defined as the system or service's operational structure, in effect, a summary of the various components and how they are interconnected. For our purposes, the architecture later is distinct from the action layer. Frameworks and analyses within the action layer focus on the execution of activities and the quality of performance related to those activities. In contrast, the architecture layer primarily concerns the specification and location of components and their interfaces. Examples of architectural specifications include organization charts, entity-relationship diagrams, and technical blueprints. Such architectural descriptions are essential for analyzing and improving processes and activities that occur in the action layer.

Just as the action layer is divided into four quadrants, the architecture layer can also be divided into four quadrants. Figure 4 highlights two types of relationships:

- Relationships between the quadrants in the action layer and corresponding sections of the architecture layer.
- Relationships between adjoining sections of the architecture layer.

Business organizations. The upper left-hand portion of the architecture layer is devoted to the architecture of business organizations. The link in Figure 4 between "service as work system" and "business organizations" represent the fact that business organizations can be viewed as a summation of the work systems that they contain.

Value configurations. Stabell and Fjelstad (1998) observed that different types of firms may have fundamentally different ways of providing value for their customers. They identified three value configurations: value chains, value shops, and value networks.

- Value chains transform inputs into product outputs. Examplars are manufacturers. Primary value chain activities tend to be sequential, and include inbound logistics, operastions, outbound logistics, marketing, and service.
- Value shops (re)solve customer problems. Exemplars are professional service firms in consulting, medicine, law, architecture, and engineering. Primary value shop activities tend to spiral, and include problem finding and acquisition, problem solving, choice, execution, and control/ evaluation.

- Value networks mediate between suppliers and customers. Exemplars are retailers, banks, insurance companies, and brokers. Primary activities tend to be simultaneous and parallel, and often include network promotion, contract management, service provisioning, and infrastructure operation.

The link in Figure 4 between “service as co-production” and “value configurations” combines two assumptions. These are the action layer’s assumption that value is co-produced by providers and customers and the architecture layer’s assumption that a value configuration is a summation of a firm’s back-stage activities and its front-stage co-production activities.

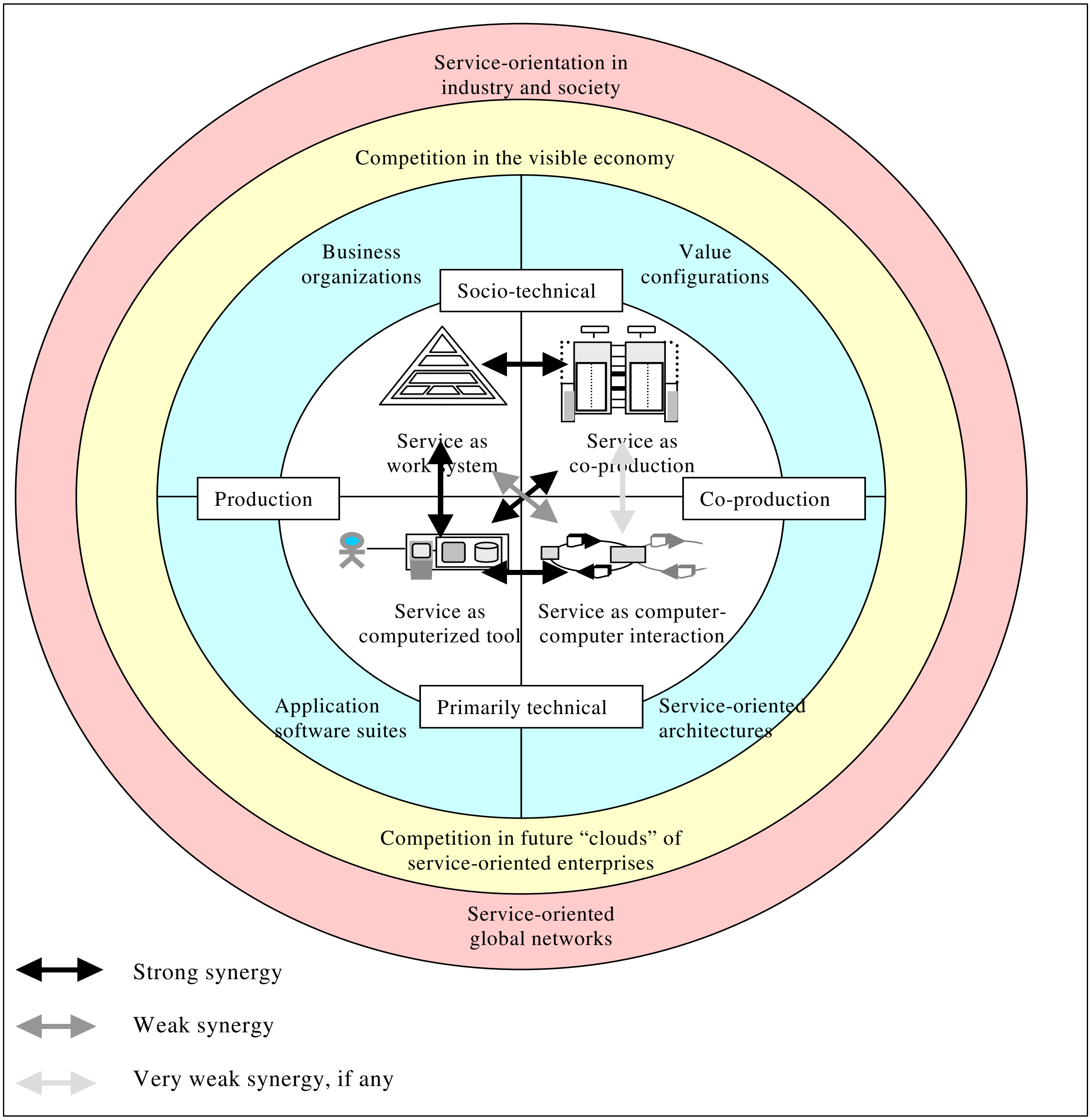


Figure 3: Synergies between quadrants in the action layer of the service domain framework

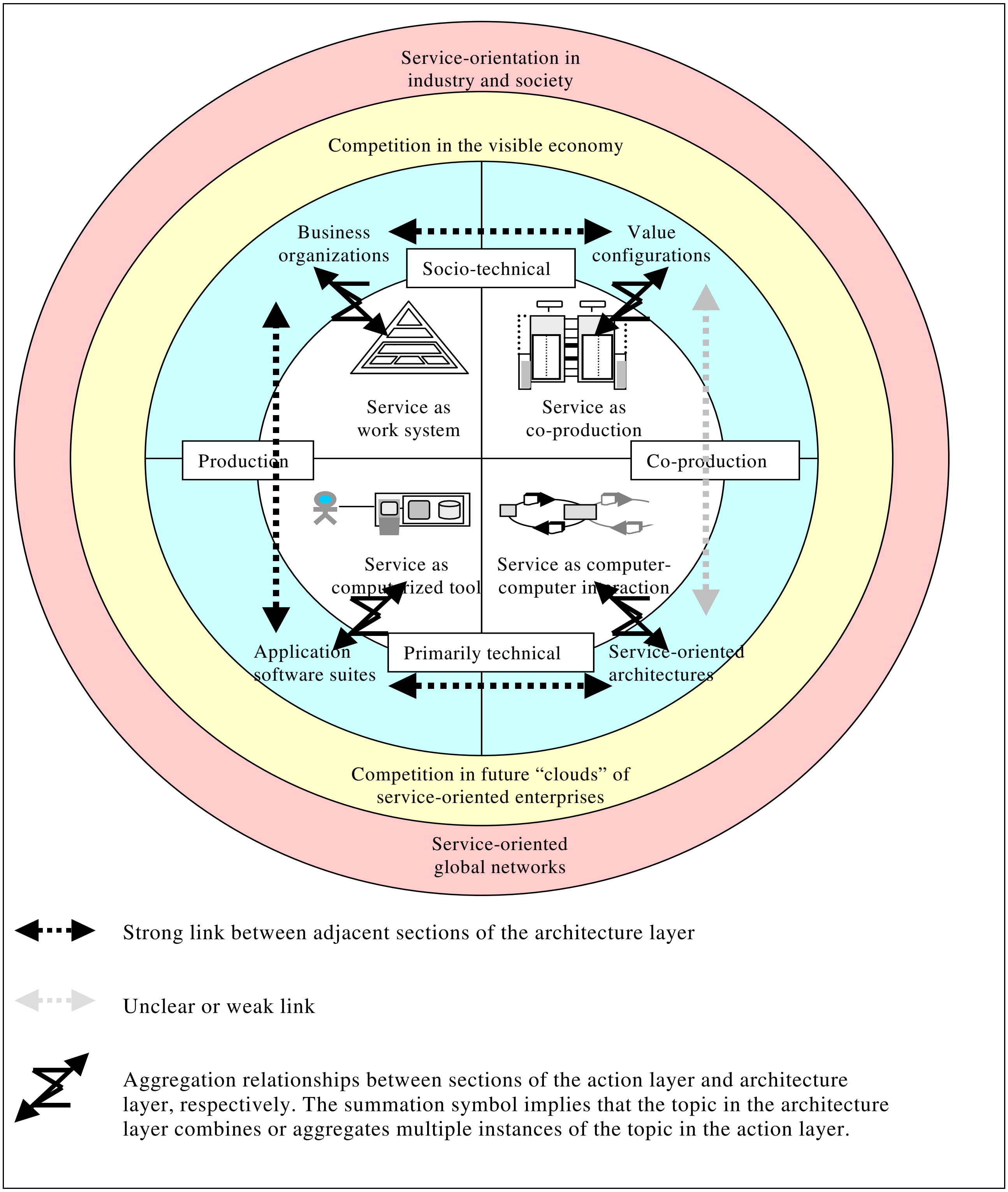


Figure 4: Links between the action layer and architecture layer, and between different parts of the architecture layer in the service domain framework

Application software suites. The tools that appear in the "service as computerized tool" quadrant of the action layer are often individual components of commercial application software suites such as ERP or CRM suites, or are modules within home-grown software systems. Links in Figure 4 between "application software suites" in the architecture level, and "computerized tools" in the action level say that many computerized tools are components of application software suites or home-grown software systems.

Service-oriented architecture (SOA) applies services computing as a programming architecture for building applications from software services that are self-contained and are unaware of the context or state of other services. The message-based loose coupling between services facilitates building software systems from software modules that may have been programmed at different times for different purposes. Often touted as promoting organizational flexibility, SOA facilitates use of legacy software in conjunction with newer software developed according to current programming practices.

There is a tantalizing similarity in certain fundamental concepts from (1) service computing in the action layer, (2) service-oriented architectures in the architecture layer, and (3) services as co-production in the action layer. For example, Figure 1 in zur Muehlen et al (2005), which concerns web services choreography standards, represents inter-organizational process integration as occurring through messages (analogous to service interactions) between two organizations (analogous to provider and consumer), each having both private and public processes (back-stage and front-stage). Despite the surface similarity, the computing approaches differ substantially from service as co-production in context and terminology. For example, a description of IBM's "business architecture for a service-oriented enterprise" (Nayak et al, 2007) explains how services are "exposed" through a catalog, "discovered" by searching a catalog, and invoked (automatically) only if a service agreement exists. Typical business professionals would not use such terms to describe service provision by human providers for human customers.

Links within the architecture layer. Several adjacent components of the architecture layer have important relationships.

- If form truly follows function in organizations, then the architecture of business organizations should closely mirror the value configuration that the business uses. Conversely, changes or improvements in a firm's value configuration should generate changes in its organizational architecture.
- Similarly, the configuration and architecture of application software suites should mirror the architecture of business organizations that use them. Misalignments increase the likelihood that the business will be unable to use the software fully, or even worse, that the software will impede efficient operation of the business.
- The link between service-oriented architectures and application software suites is important in current practice, as vendors increasingly apply service-oriented architectures to increase the flexibility of their software.
- The link between value configurations and service-oriented architectures is not nearly as clear, and may not exist in any meaningful sense (see Figure 7 later in the paper).

LAYER #3: THE ECONOMIC EXCHANGE LAYER

Layers #1 and #2 represented actions and architectures, respectively. Layer #3 represents economic exchange and competition. Layer #3 is based on the assumption that economic exchange will occur between service providers and their external customers mainly where the service provider's value propositions for its customers are superior to those of its competitors.

Competition in the visible economy. The nature of competition has been discussed for decades. We use the term "visible economy" to distinguish between traditional views of competition, e.g., Porter (1985) and Prahalad and Ramaswamy (2004), and future possibilities on the other side of this layer.

Competition in future "clouds" of service-oriented enterprises. The purposefully vague term "clouds" highlights the speculative nature of the bottom side of this layer. The idea of service-oriented enterprise is largely a technical vision. For that technical vision to have competitive impact for companies other than IT vendors, its impact must be realized though real world systems and services in the action layer. Steps toward that vision are reported in Demirkan and Goul (2006).

LAYER #4: THE INDUSTRY AND SOCIETY LAYER

The summation of the action layer, architecture layer, and economic exchange layer has impacts on the nature of industries, nations, and society in general. The sociotechnical parts of those impacts occur at the top of layer #4; the technical parts occur at the bottom.

Service-orientation in industry and society. Economists have written extensively about economic transitions between agricultural, industrial, and service economies. On-going economic research continues to explore the reality and impacts of trends toward more of a service economy.

Service-oriented global networks. It is safe to say that the Internet's current capabilities pale in comparison with the possible form and power of global networks of the future. Such networks might realize parts of the vision of the semantic web (Berners-Lee, Hendler, and Lassila, 2001). Although there is a place for global networks of the future at the bottom of the service domain framework, this paper will not speculate about those networks or about whether and how those networks will affect the service-orientation of industry and society (the part of the layer #4 at the top of the framework).

SAMPLE APPLICATIONS OF THE FRAMEWORK

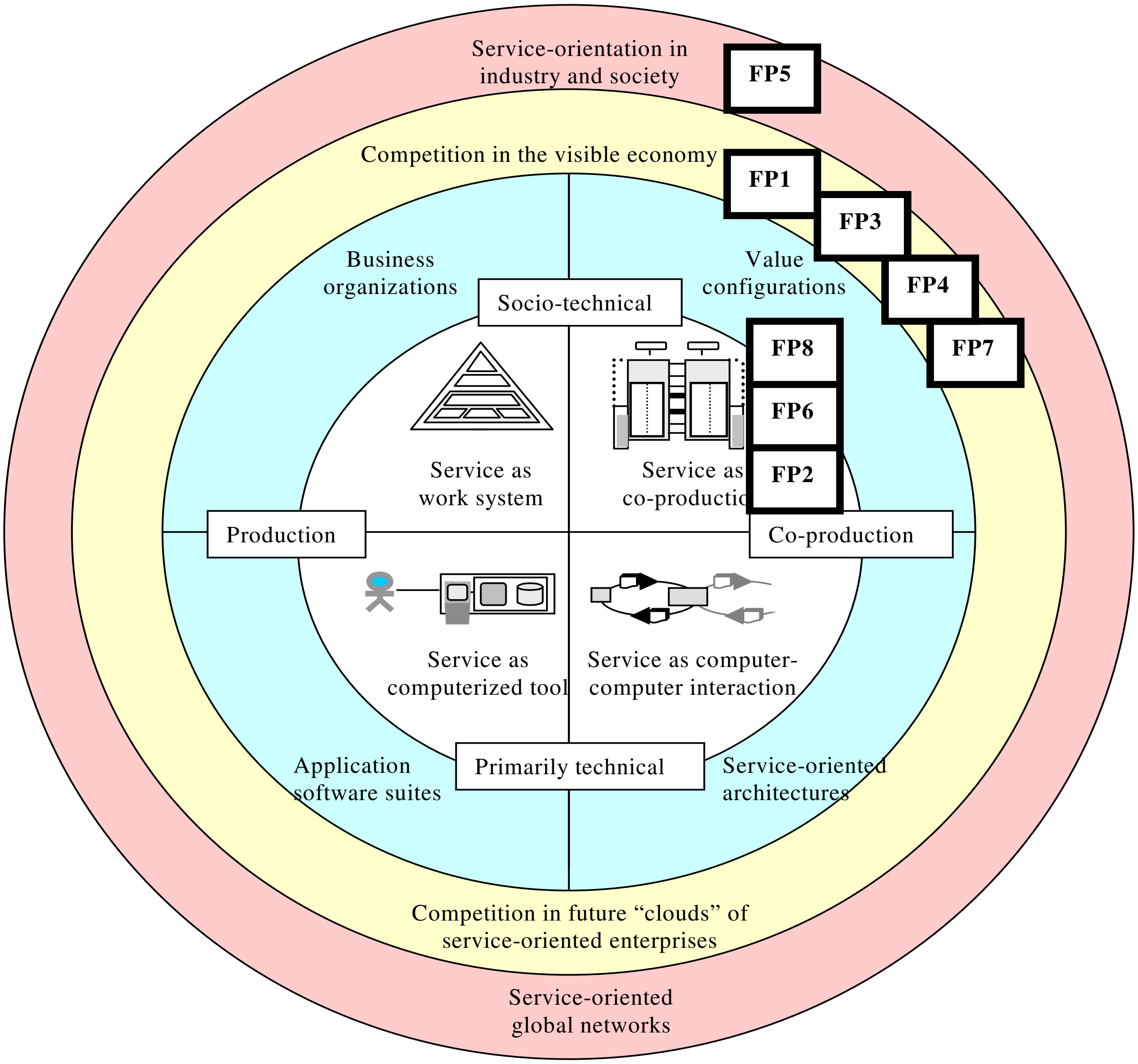
Figures 2, 3, and 4 illustrated several ways in which the framework might help in linking topics within service science and possibly in helping people understand, analyze, and research service-related situations. To illustrate further applications, this section looks a where service-dominant logic and software as a service belong in the framework, and mentions other topics whose various facets could be located, interpreted, and explored.

Service-dominant logic. The service-dominant logic introduced by Vargo and Lusch (2004) argues that traditional goods-dominant logic is insufficient for understanding current markets, economic exchange, and marketing. The final white paper from a service science symposium attended by leading service researchers stated "Service Science embraces the world view of the service-dominant logic." (IfM and IBM, 2008, p. 17)

Figure 5 attempts to place each of eight foundational premises of service-dominant logic in an appropriate location in the service domain framework. Those foundational premises touch portions of four different layers, but are restricted to the upper right quarter of the framework. None of the foundational premises touch other parts of the framework that contain important topics such as business organizations, service as work system, service as computerized tool, service-oriented architecture, and so on. As shown in Figure 2, research related to those parts of the framework applies disciplines such as organization studies, operations management, information systems, and computer science. In other words, the IfM and IBM (2008) white paper embraces the worldview of service-dominant logic, but the worldview of service-dominant logic seems quite distant from the worldviews of important disciplines that are obviously relevant to service provision. One can argue about whether this observation has any important implications. The point here is that the framework could help in visualizing and debating implications of an assertion concerning service science.

Software as a service. SaaS is another topic that is often considered within the domain of service science. Figure 6 identifies different parts of the service domain framework that are related to SaaS. Clearly, many different parts of the service domain framework are relevant when thinking about SaaS. Most interesting from a pure service viewpoint is whether any particular SaaS offering exploits the possibilities of co-producing value to the greatest extent that is economically feasible.

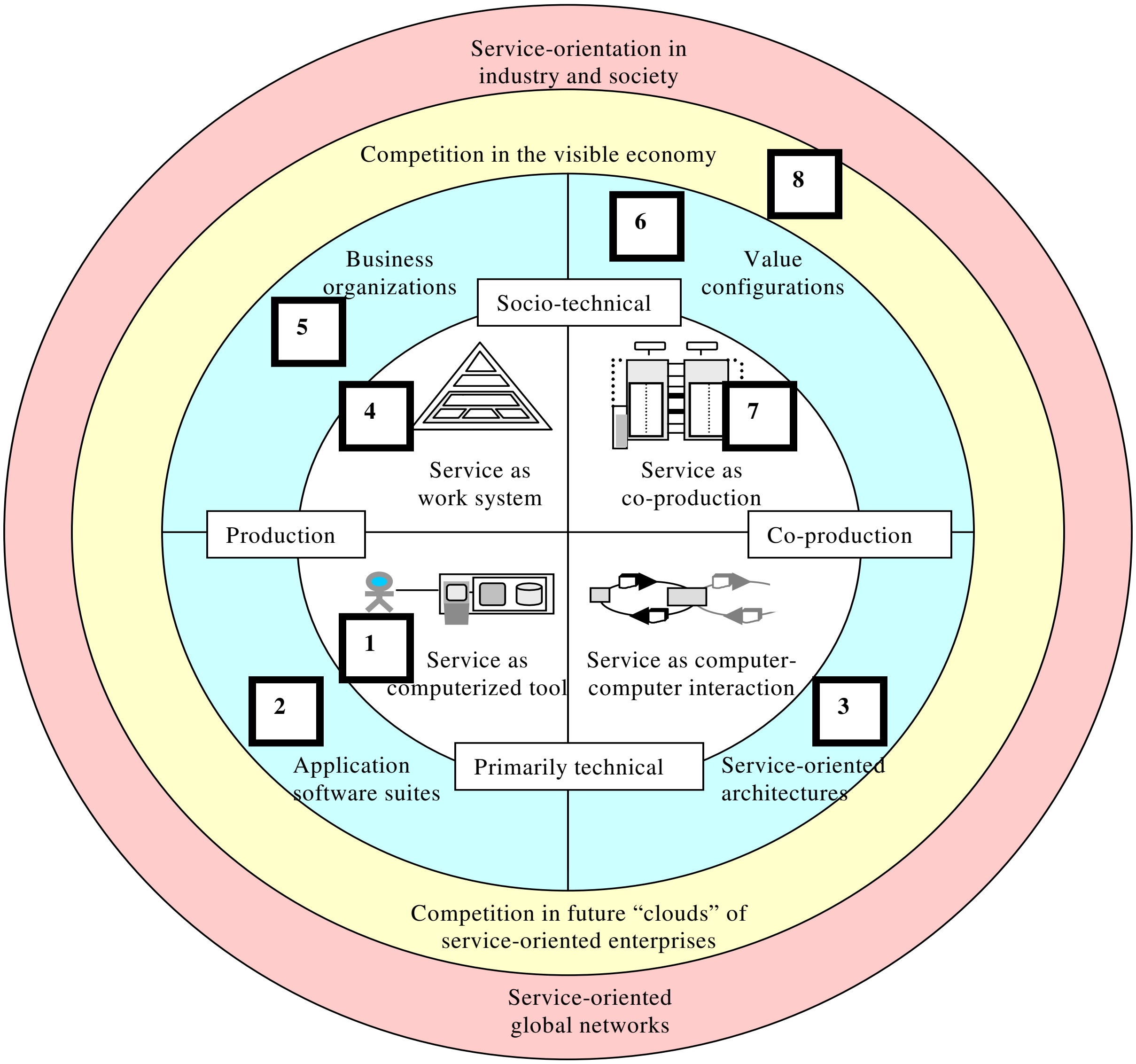
Open source, IT service management, ERP, and CRM. These four are among many additional service-related topics whose various facets can be located, interpreted, and explored using ideas in the service domain framework. For example, the software development and testing practices in open source projects might reveal new possibilities and models for co-producing value. On the other hand, nothing about open source software *per se* implies that open source software will be more amenable to supporting co-production of value by users of that software. Similarly, use of the service domain framework and related ideas to study existing ERP and CRM suites might reveal that they mostly reflect a production mindset. That type of analysis might motivate attempts to build more of a co-production mindset into future versions.



Eight foundational premises of service-dominant logic (Vargo and Lusch, 2004)

- FP1. The application of specialized skills and knowledge is the fundamental unit of exchange.
- FP2. Indirect exchange masks the fundamental unit of exchange.
- FP3. Goods are distribution mechanisms for service provision.
- FP4. Knowledge is the fundamental source of competitive advantage.
- FP5. All economies are service economies.
- FP6. The customer is always a coproducer.
- FP7. The enterprise can only make value propositions.
- FP8. A service-oriented view is customer oriented and relational.

Figure 5: Location of foundational premises of service-dominant logic in the service domain framework



Issues and observations related to software as a service (SaaS) fall in many different areas of the service domain framework.

- 1. SaaS often provides access to numerous computerized tools that operate on the vendor's servers.
- 2. SaaS can provide capabilities in the form of an application software suite.
- 3. SaaS can apply service-oriented architectures to increase flexibility.
- 4. SaaS is a delivery mechanism for technology that it used in in one or more of the client's work systems. In addition, the processes and activities related to setting up and monitoring SaaS arrangements are work systems on their own right.
- 5. An interesting issues about SaaS is whether it affects the operation of business organizations in any way other than merely using different means for providing essentially similar technical capabilities.
- 6. It is interesting to think about which category of value configuration applies to SaaS.
- 7. Co-production of value is inherent in SaaS because the vendor provides the means that allows a client to perform data processing more economically and conveniently. It might be interesting to analyze the extent to which the service value chain framework (Alter, 2008a, 2008b) illuminates negotiations and other processes related to SaaS.
- 8. SaaS represents a competitive threat to traditional software vendors doing business in a traditional way.

Figure 6: Location of various aspects of Software as a Service (SaaS) in the service domain framework

Bridging the gap between human and machine services. The horizontal rectangle superimposed on the service domain framework in Figure 7 represents an important practical and theoretical gap. There are many claims and speculations about how service computing, service-oriented architectures, and service-oriented enterprises will improve business performance and might provide competitive advantage. While those approaches might improve the flexibility of software development and modification and the efficiency of programming groups, it is not clear how those improvements lead directly to competitive advantage.

The arrows in Figure 7 show the general direction of a path from SOA toward competitive advantage. Efficiencies from SOA could make it easier to develop computerized tools that will be used within work systems. Those tools might help in improving internal productivity (a producer view) and/or might help in improving the co-production of value for customers. The performance improvement in either case might lead to competitive advantage, at least until a competitor attains similar capabilities.

CONCLUSION

The introduction noted questions about whether all topics containing the word service (e.g., service interaction, customer service, SOA, and web service) actually belong together in a single discipline called service science. This paper assumed that the answer is yes, proposed a framework, and explored the framework's applicability. Many other uses of the service domain framework and much more detailed explanations could be included in a longer paper.

The most basic test of the framework involves whether all service processes and all research under the general umbrella of service science fit somewhere and whether the framework helps in locating, comparing, understanding, and recognizing synergies and conflicts between service science topics. These issues were explored by looking at seven figures, each of which introduced a different facet of the discussion:

- Figure 1: the service domain framework's form and major topics
- Figure 2: the location of topics covered by disciplines related to service
- Figure 3: synergies between quadrants in the action layer
- Figure 4: links involving the architecture layer
- Figure 5: location of foundational premises of service-dominant logic
- Figure 6: location of various aspects of SaaS
- Figure 7: path for bypassing the gap between human and machine services

While it is impossible to prove that any framework is optimal, the discussion of these figures illustrated that the service domain framework locates many important topics in service science, and potentially provides insights about many issues that are worthy of exploration. A challenge for the future is to develop other frameworks with the same goals, and to compare the service domain framework with the others based on criteria such as coverage, understandability, and fruitfulness.

Since the service domain framework covers topics in many disciplines, another challenge involves using it to explore possibilities for multi-disciplinary research in services. For example, the emerging concept of servitization might be explored by considering how servitization in a particular industry domain might benefit from combining knowledge from multiple disciplines.

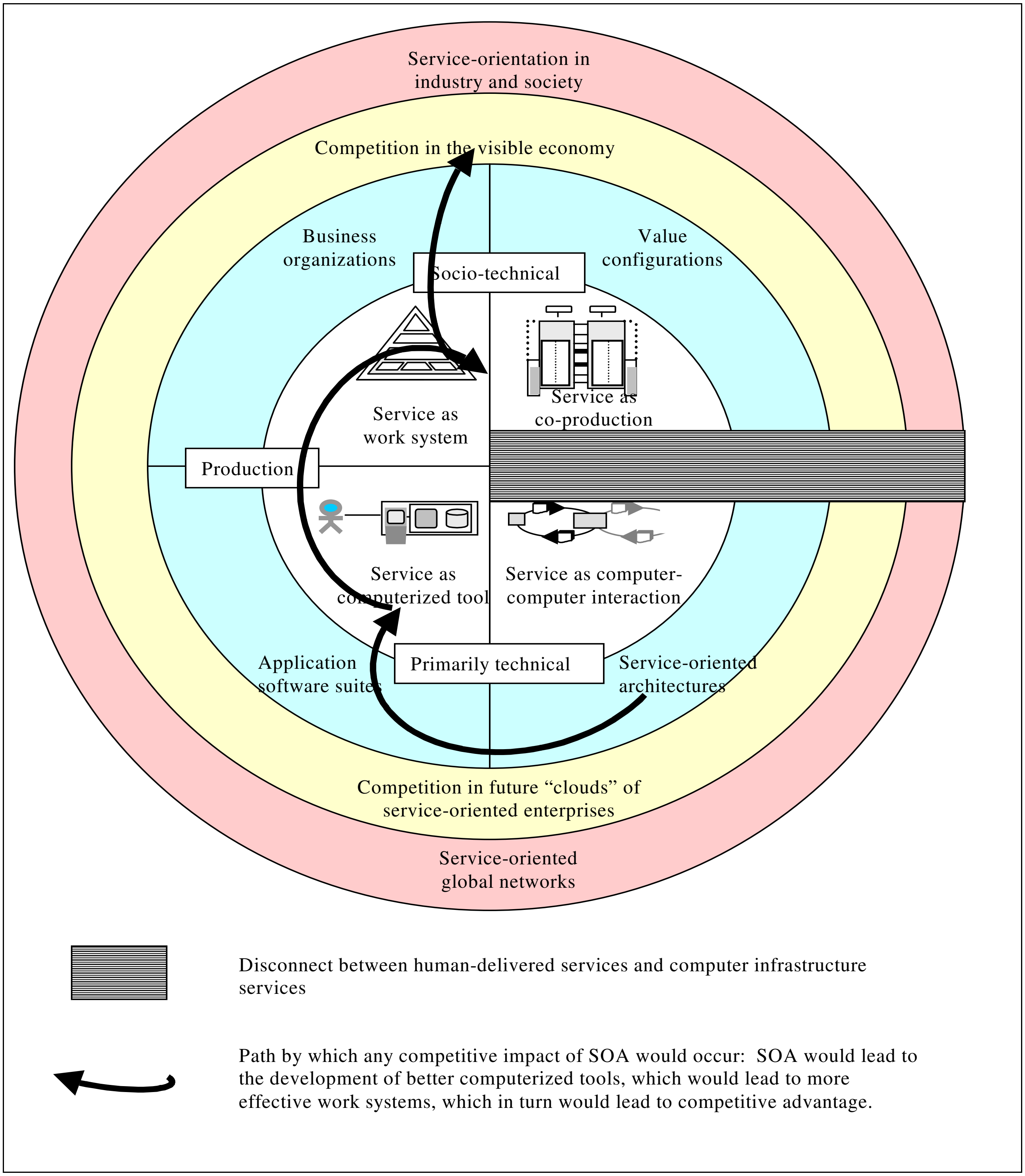


Figure 7: Bypassing the gap between human and machine services

REFERENCES

Alter, S. (2003) "18 Reasons why IT-Reliant Work Systems Should Replace the IT Artifact as the Core Subject Matter of the IS Field," *Communications of the Association for Information Systems*, 12(23). http://aisel.aisnet.org/cais/vol12/iss1/23

Alter, S. (2006) *The Work System Method: Connecting People, Processes, and IT for Business Results*, Larkspur, CA: Work System Press.

Alter, S. (2008a) "Service System Fundamentals: Work System, Value Chain, and Life Cycle," *IBM Systems Journal*, 47(1), pp. 71-85. http://www.research.ibm.com/journal/sj/471/alter.pdf

Alter, S. (2008b) "Seeking Synergies between Four Views of Service in the IS Field," *Proceedings of AMCIS 2008, the 14th American Conference on Information Systems*, Toronto, Canada, August, 2008. http://aisel.aisnet.org/amcis2008/76

Alter, S. (2008c) "Defining Information Systems as Work Systems: Implications for the IS Field," *European Journal of Information Systems,* 17(5), October, pp. 448-469.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001) "The Semantic Web," *Scientific American*, May, http://www.sciam.com/article.cfm?id=the-semantic-web

Brown, A.W. et al (2005) "Realizing service-oriented solutions with the IBM Rational Software Development Platform," *IBM Systems Journal*, 44(4). http://www.research.ibm.com/journal/sj/444/brown.pdf

Carlzon, J. (1989). *Moments of Truth*, New York, NY: Harper Collins.

Cherbakov, L.G. et al (2005) "Impact of service orientation at the business level," *IBM Systems Journal*, 44(4), pp. 653-668. http://www.research.ibm.com/journal/sj/444/cherbakov.pdf

Chesbrough, H. and Spohrer, J. (2006) "A research manifesto for services science," *Communications of the ACM*, 49(7), pp. 35-40.

Demirkan, H. and Goul, M. (2006) "Towards the Service Oriented Enterprise Vision: Bridging Industry and Academics," *Communications of the Association for Information Systems*, 18(26). http://aisel.aisnet.org/cais/vol18/iss1/26

IfM and IBM (2008) "Succeeding through service innovation: A service perspective for education, research, business and government," www.ifm.eng.cam.ac.uk/ssme

Leavitt, T. (1960) "Marketing Myopia," *Harvard Business Review*, 38(4), pp. 45-56.

Nayak, N. et al (2007) "Core business architecture for a service-oriented enterprise," *IBM Systems Journal*, 46(4), 723-742.

Porter, M.E. (1985). *Competitive Advantage: Creating and Sustaining Superior Performance*. Free Press, New York.

Prahalad, C.K. and Ramaswamy, V. (2004) *The Future of Competition: Co-Creating Unique Value with Customers*, Boston, MA: Harvard Business School Press.

Sampson, S.E. and Froehle, C.M. (2006) "Foundations and Implications of a Proposed Unified Services Theory," *Production and Operations Management*, 15(2), pp. 329-343.

Spohrer, J, Maglio, P. P., Bailey, J., & Gruhl, D. (2007) "Steps toward a Science of Service Systems," *Computer*, 40(1), pp. 71-77.

Stabell, C.B. and Fjeldstad, Ø.D. (1998) "Configuring Value for Competitive Advantage: On Chains, Shops, and Networks," *Strategic Management Journal*, 19, pp. 413-437. www.value-networks.com/Articles/configuring_value.pdf

Teboul, J., (2006). *Service is Front Stage: Positioning Services for Value Advantage*, INSEAD Business Press, Palgrave MacMillan, UK.

Vargo, S.L., and Lusch, R.F. (2004) "Evolving to a New Dominant Logic for Marketing," *Journal of Marketing*, 68, pp.1-17.

Wood, J.B. (2007) "Major Service Challenges Facing Technology Companies in 2007," Frontiers of Service Conference, San Francisco, CA.

Zhao, J. L. et al (2008) "Impact of Service-Centric Computing on Business and Education," *Communications of the Association for Information Systems*," 22(16), pp. 295-310. http://aisel.aisnet.org/cais/vol22/iss1/16

zur Muehlen, M., Nickerson, J.V. and Swenson, K.D. (2005) "Developing web service choreography standards – the case of REST vs. SOAP, *Decision Support Systems*, 40, pp. 9-29.