A Management and Enterprise Architecture Framework for Comprehensive Structure Design of Complex Services

Oscar Barros, University of Chile, Chile

ABSTRACT

This paper centers on the full structure design of firms providing complex services, like health, which include the production flow, its management, and flows components. This work proposes a full design solution, improving on current partial designs. The proposed solution integrates disciplines—modularity, enterprise architecture, analytics, and economics—in a methodology that learns from real-life services in several domains to discover common features for them. Thus, it develops a descriptive proposal of a general service management and enterprise architecture framework using an architecture pattern. The pattern generates a prescriptive method to generate design alternatives, intelligence provides options for supporting or automating production and management, and economics helps to choose a design alternative for users. The expected benefit in services explicit, systemic, and innovative design is to generate a large impact on their scope, quality, and efficiency.

KEYWORDS

Complex Services, Design Evaluation, Design Framework, Enterprise Architecture, Intelligence, Modularity, Service Design

1. INTRODUCTION

This paper studies the design of the structure of complex services provided by multilevel enterprises. It begins by characterizing the problem under consideration and reviewing cases that show the complexity of providing comprehensive structure design solutions, to discover research opportunities for improving such designs.

Multilevel enterprises providing complex services have several organizational levels that take the form of a central management component and several structure levels such as divisions, products, service lines, and locations. Such levels usually form a sort of hierarchy, with several branches that may be completely independent of each other, but there are cases in which they interact by exchanging goods, services, and clients. The problem of interest is how to design such services founded on knowledge generated by the design, implementation, and evaluation of real-world cases. The purpose is to provide new design ideas, based on such evidence, further characterized subsequently.

Specific real examples of services of the type just outlined, which show the design challenge and different design approaches, are as follows.

a) Services To the Elderly in Holland that have a design of bundles of services, such as (de Blok et al., 2014):

DOI: 10.4018/IJSSMET.290335

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.
Psychiatric care, including components as “monitoring, scaling up, intensify care and crisis intervention”.

Medication care, with components as “long lasting medication collected at preset times and regular medication collected from the pharmacy.”

This case shows the need to assemble a particular service for a user by selecting the proper components according to his needs. For such assembling, they use protocols that guide the different operating and management units to define what is relevant for each situation.

b) **Shipping Services** provided by a European company (Pekarinen and Ulkuniemi, 2008) have two business lines:

- “Subcase A that moves physical goods in large shipments for the fashion and health care industries by using outsourcing and partnerships”
- “Subcase B that moves physical goods in small shipments for manufacturing industries by using its fleet”.

In Subcase A, the services’ design provides customized versions for several industries. Since the logistics needs for them are different, “similarities in customer needs are considered to build cost-efficient solutions”. For example, service modules for “offerings in order management, supply chain management, and vendor inventory management” designs are the same for all industries. In Subcase B, the design defines “highly standardized services with over 50 modules, but complemented with personalized value-added services that provide some customization”. Examples of the different standardized process modules that the design contains are: “goods pick up, loading, transporting, unloading and delivery, sending and receiving orders, tracking shipments, and control and monitoring processes”. The design includes an IT service platform that facilitates the creation of particular assemblies of services for different clients.

c) **Integrated Practice Units**, defined by Lee & Porter (2013) based on the experience of several large USA hospitals that successfully innovated in the redesign of their medical production flows. The design of a practice unit is for a given clinical condition, e.g. low back pain, and uses a well-designed specialized medical flow, which clearly defines what to do for a patient’s condition. Some work of this type has also been done for emergencies, including the introduction of split flows specialized by type of patients in hospitals in the USA (Saghafian et al., 2012), and Chile (Barros, Riffo, & Paredes, 2019).

The cases presented allow identifying the subsequent design levels.

The first is the design of the **service production flow**, i.e. the sequence of activities directly related to the generation of the service provided for a user. Examples are services to the elderly and health emergencies, given previously.

A second level is designing the structure to **manage the production flow**, including processes for operational, resource and capacity management, and strategic issues such as innovation. A set of management’s design process needs for health services, detailed in Barros (2017), are emergency capacity management, operating room scheduling, ambulatory waiting list management, and innovation management for a hospital network.

The third level of **comprehensive structure design** includes the service-production flow design and the enterprise structure needed for its execution and management at all levels; for example, the organizational setup—divisions, locations, departments, operating units—, the process architecture, and the supporting IT structure, as the shipping case shows. This design is necessary when a new complex service starts or radical innovations in an existent service changes the business model, including new services and/or a new way of delivering the current one, such as in the case of integrated practice units.
In the next section, a review of current service design methods will show that there are not integral solutions for the comprehensive structure design need established above. Thus, the research gap will be defined and the need for a new design approach established.

1. LITERATURE REVIEW

Proposals for the design in cases defined in the previous section do not provide a complete solution for comprehensive structure design, as summarized subsequently. Thus, there is a research opportunity that this paper defines in Section 2.2, based on the integration of disciplines reviewed in Section 2.3.

2.1. Current Service Design Methods

Current Service Design Research (SDR) mostly support the first and second levels of Section 1.3, as it is clear in the review paper by Joly, Teixeira, Patricio, & Sangiorgi (2019) that classifies outcomes from its use at the “micro, meso, and macro” levels. Most of the works reviewed produce outcomes at the micro or meso level, which corresponds to the first and second design levels previously defined. The authors report a few works covering the macro level, equivalent to the third level of comprehensive structure design, which center on contributions to “enable institutional change, by envisioning new services ecosystems that support more sustainable lifestyles and consumption habits, as well new services concepts that change citizens’ practices and routines”. These works provide general ideas and guidelines for designing such ecosystems but do not give specific methods for the comprehensive design issues considered here. One of the proposals reviewed is “multilevel service design” (Patrıcio, Fisk, Falcao e Cunha, & Constantine, 2011) that includes a general model with levels of “service concept, service system, and service encounter design”; these levels apply to “understanding the customer experience” and “designing the service offering”. For each design level, they propose simple graphical models, which work well with simple cases, as the retail grocery management and bank car loan’ cases developed in the paper. Another related proposal is the MINDS method (Teixeira et al., 2017) with modeling constructs such as the “experience blueprints” and the “value constellation and affinity diagram”, applied for designing a new service for watching football and a simple dermatology service. This method also works for these simple cases, but, as the previous one, it does not apply to the design of the comprehensive structure defined here.

Some papers deal explicitly with SS configuration design as defined here. Thus, Trad (2021) proposes a “Business Transformation Framework and Enterprise Architecture Framework” that “can be used to model enterprise architecture (EA) blueprints, business transformation projects, or knowledge management systems; it is supported by many real-life cases of various business domains”. The objectives of his paper are similar to the ones of this work, but methods are very different; the approach here is engineering-oriented providing very precise guidelines for design, while Trad’s method is more oriented to support the “discovery of transformation strategies”, and lacks details on design methods to implement strategies. In another paper, Van Riel et al., (2019) propose configuration variables such as “business model, key partners, key activities, value propositions, customer relationships, channels, customer segments, cost structure, and revenue streams”. Combinations of such variables allow defining alternative configurations, such as “provider, platform, hybrid, and network”, for which the value they provide to the different stakeholders allows selecting among them. An example in the hospitality industry shows how these ideas work in practice. This paper provides an interesting approach for generating new service ideas and configurations, but it needs design tools to make configurations work, the focus of this paper. Finally, Mora et al.(2011) published an “IT Service Engineering and Management Framework (ITS-EMF)”, which was “generated by careful review and examination of the main conceptualizations on IT, ITSE and business services”. The paper claims that ITS-EMF is useful for “mapping services concepts from disparate IT literature, reducing service conceptual confusion from the multiple available sources, and providing conceptual links between service constructs used in business services and IT and ITSE services layers”. Therefore, it is clear
the emphasis is on the design of the IT’s components of services and does not consider the other modules that comprehensive design includes.

Thus, none of the methods reviewed provides a solution to the comprehensive structure design problem defined in Section 1.3.

2.2. Research Gap and Opportunities

The general research gap and opportunities this paper considers are the ones discovered by Frost & Lyon (2017), particularly “applying service science theories and methods to specific domains” in the line of learning from real-life cases, which is a key idea of this paper. Additionally, they found the need “to create a service system framework which offers descriptive, prescriptive and evaluative analysis methods”. In this line, this paper presents a framework, based on patterns, which provides such methods, not currently available as concluded in the previous section.

Thus, the aim is to contribute to the comprehensive structure design need with a theoretical background that uses “artifacts” (Gregor & Hevner, 2013), based on supporting disciplines and a methodology. For this, there is the need to discover and represent alternative structure configurations—in terms of hierarchy levels, decentralization, modularity, innovation, and coordination—an enterprise may have and then be able to evaluate such alternatives for choosing the right one for a particular case (Baporikar & Randa, 2020). The objective is to provide design options by using a Management and Enterprise Architecture Framework (MEAF) based on a general architecture pattern, using experience or previous knowledge, to develop a structure specialized for a case. An architecture pattern synthesizes the knowledge of how an enterprise should design its business and its structure. It details the processes, the people who execute them, the Analytics and IS that support both of these, and the IT layer on which such systems operate.

2.3. Disciplines integrated into the Framework

The approach detailed in Section 3 includes a framework that integrates ideas from several scientific disciplines presenting different views about the nature of services and others that provide concepts and tools for their design. A summary of such disciplines, given subsequently, includes ideas not currently used in SS, which complements the literature on SDR, as the one included in the survey by Joly, Teixeira, Patricio, & Sangiorgi (2019).

a) Modularity

In complex services, modularity has proved very useful to support service production design, as previously defined, when many different components interact in a complex way to provide many variations of a service, which is an important feature of the services considered here. Examples of this idea are in the cases of Section 1.2.

Modularity is the “degree to which components of a system can be separated and recombined to create a variety of configurations without losing functionality” (Pekkarinen & Ulkuniemi, 2008). This approach originated in the design of physical products and manufacturing processes and has been adapted to services (Dorbecker & Bohmann, 2013), mainly for the design of the services provided and the production flow required.

The formalization of modularity starts with the definitions proposed by de Blok et al. (2014), as follows:

- A **Service package** exists for “providing a complete service to a user”, e.g. services to the elderly in Holland.
- **Components** are parts of a package, e.g. medical care, meal on-demand, and heavy housework help for elderly services,
A service provider is a person or automata that executes the service,

Interfaces are “enablers for the substitution or recombination of components”; they may include “people, rules, sequence constrains and devises that determine the service flow”; examples are: “product books that define the flow, physical predefined flows, protocols to evaluate needs that determine the flow, work schedules and service agreements”

The design of components and interfaces has rules –derived from de Blok et al. (2014), and Broekhuis, van Offenbeek, & Eissens-van der Laan (2017) – given subsequently.

i) Design components that have a “specific function and are relatively independent, minimizing interaction and information exchange among them”, also justified by the coordination theory in Section 2.3.

ii) Design interfaces that “facilitate recombination, by coordinating and managing interactions”. The consequence of this rule is that the production flows design determines interfaces since in each step it defines the need to interact with other flows and provides mechanisms for doing so.

iii) As a result of rules (i) and (ii), the design of supporting IT follows, including Intelligence, since IT is the tool to automate or facilitate the work of interfaces.

Thus, a way of representing relationships, managed by interfaces, among the components of a service package is necessary, since their design determine the interactions between such components. For this representation, formal process models for the flows that explicitly show how components are coordinated to allow for substitution are necessary.

Therefore, there is the need to represent the structure of the service provided by an SS, and the production flow for its generation (Yang, Xing, & Ness,2013); also, an approach for its design using modularity is necessary (Becker & Klingner, 2013). The MEAF, which uses the architecture pattern developed in Section 4, provides such representation and design approach.

b) Enterprise Architecture

In this work, an enterprise architecture (EA) pattern synthesizes the knowledge on how an enterprise should structure its business processes and production flows, the people who execute them, and the Analytics and the IT layer that provide support, assuring alignment with given strategic guidelines and a business model. Thus, the formalization of an EA is necessary for the comprehensive structure design defined previously.

Ross, Weil, and Robertson (2006) developed an approach for EA characterization, by linking it with strategy and providing a conceptualization of the different business operating models that determine the structure of an enterprise. They proposed four types of EA based on the degree of business process integration and business process standardization observed in many cases in practices as follows. Diversification focuses on “decentralized organizational structure design with high local autonomy”, as opposed to Unification, which pursues “low costs and standardization of business processes through centralization”. Coordination focuses highly on “integration without forcing specific process standards”, whereas Replication pursues “standardization with low integration among the different units”. Thus, depending on the chosen EA type, a corresponding structure applies.

As concluded in Section 2.3, there is the need to have a formal modeling method to represent alternative structure configurations an enterprise may have. Such a method should allow the formal representation of the EA structures for the general pattern proposed in Section 4 and particular cases. Barros (2019) developed an approach for such modeling with applications to hospitals’ services. It uses the Business Process Management Notation (BPMN) to model activities, the flow, and the
logic of processes configuration (White & Miers, 2009), which serves as a basis for the modeling proposal given in Section 4.

c) Analytics

The importance of advanced versions of Analytics is that they allow giving intelligent support for or to automate service production components and activities that manage them, using models or algorithms, as shown later. Here, Analytics of two types are considered:

- Data-based –traditional statistics and econometrics, data mining, and web mining–to develop predictive models, and artificial intelligence (AI), including machine learning (ML), which may use big data, for predictive and prescriptive models (Sangwan & Bhatnagar, 2020).
- Operations Research and Management Science providing prescriptive models, with techniques such as optimization models –linear, non-linear, and discrete–, heuristics, probabilistic models, and simulation.

d) Evaluation Concepts

In evaluating designs generated with the help of the disciplines just presented, the following Economics concepts provide ideas for evaluating alternative structures.

**Coordination cost** is a production cost that increases at a rate greater than linear as an organization grows, due to diseconomies of scale generated by management’s complexity inherent to large size (Malone & Crowston, 1994). An option to manage coordination costs is to split a service into smaller units to make them operate in a relatively independent way, reducing costs in coordinating the production of several different products or services. Hence, this theory supports defining specialized split health production flows, integrated practice units, and independent components of modularity, as presented previously.

Another evaluation approach is **agency theory**, where the executive of an organization is the principal and subordinates are agents (Arrow, 1985). This theory proposes that a company is a set of related contracts among individuals with their particular interests, instead of the usual assumption that they maximize utility. Therefore, a set of agency contracts allows the principal hiring agents (employees) to perform tasks. Costs in this theory depend on how such contracts define relationships among the principal and agents. One of the key factors that determine such relationships is the degree of decisions’ centralization. Thus, with high centralization, there are information-processing costs that include the information necessary for the principal to make decisions about agents’ behavior, which increases. Additionally, opportunity costs arise due to a lack of or erroneous information for decision-makers. Alternatively, decentralization increases the monitoring cost of agents to assure a behavior consistent with the principal’s interest; but this does not assure expected results, so there is a residual loss. Therefore, the degree of decentralization of decisions is a management design variable in any complex service.

3. RESEARCH APPROACH

This section presents the concepts and justification of the methods used in this research. It also provides ways to evaluate the usefulness of research proposals.

3.1. Service System Characterization

The SS definition adopted here is the one by Wang et al. (2015) because of its alignment with the engineering structure design approach this paper proposes. They define a SS as “a function that is achieved by an interaction between a human and an entity under a protocol. A service system, or
organization or firm consists of three subsystems: (i) an infrastructure, (ii) a substance, and (iii) a management to meet demands of humans or consumers. The infrastructure is of network, and substance flows over the infrastructure. The management plays the roles such as coordinating, leading, planning and controlling, which are applied to both the infrastructure and substance systems”. This definition makes clear the components of a SS, the structure design problem at hand, and the disciplines required to support such design. To show this, an Ontology that formalizes components and relationships is in Figure 1.

Figure 1. Service system ontology

Thus, interpreting the Ontology, the challenge is to design the architecture of the network –actors, resources, and activities- of the SS, and the substance that will flow through it, with the multiple interactions among its components, in providing value for the human or consumer (Cesarotti, Giuusa, Kwan, & Spohrer). Additionally, the “management” in the model of Figure 1 requires coordination, leading, planning, and control of the flows, which must be included in the designs when they are necessary to make modularity work.

For the design of the architecture framework of a SS, there is the need for concepts, tools, and methods to determine the structure of the different components, for which a selection considered relevant here, according to the review in Section 2.3, is the following.

- For the structure of the flow, modularity has a well-defined and proven proposal,
- For the design and modeling of the architecture, EA concepts and process modeling provide ideas and tools,
- For the integration of technology in the flow and its management, modern tools of AI and optimization models –Analytics–, supported by IT, facilitate innovation.
- In generating ideas for the service structure and its evaluation, Economics ideas are available.

3.2. Foundations for Framework Development

As stated previously, the problem of interest in this paper is to develop a design methodology for the comprehensive structure of complex services. In particular, the aim is to develop a MEAF to support the design of the architecture of the components of such services, defined above. Other authors support the importance of architecture design. For example, Bohmann, Leimeister, & Moslein (2014) explicitly consider architecture design within their proposal of “Service Systems
Engineering” (SSE) and provide some ideas for such design; and Pombinho, Aveiro, & Tribolet (2015) propose an “Enterprise Ontology Design and Engineering Methodology for Organizations”, which includes service architecture design. However, these approaches do not consider real-life complex case experiences to support their proposals, which is a key idea of the methods proposed here. This shows the relevance of this problem, the absence of proven solutions, and hence the research gap.

The overall approach of the methods proposed here relates to Frost & Lyon (2017), which concluded that “applying service science theories and methods to specific domains” is a research opportunity, due to a “lack of empirical evidence for design proposals”. Bohmann, Leimeister, & Moslein (2014) also recognize the “lack of evidence-based on real-life design knowledge rooted in the design, implementation, and evaluation of real-world service systems”. The main idea of the proposals of this paper is in the line of learning from real-life cases, including informal knowledge in several domains as the ones reported in the Introduction, to develop patterns based on their common features.

Another research opportunity defined by Frost & Lyon (2017) is to “create a service system framework which offers descriptive, prescriptive and evaluative analysis methods”. Thus the methods proposed in the next section offer a descriptive proposal for a general architecture pattern for complex SS, a prescriptive proposal of how to generate designs for such systems using the pattern, and a method to evaluate design alternatives and chose the one that offers the best value for the “human or consumer” in Figure 1.

The last research opportunity found by these authors is the “introduction of innovation typologies and application of existing ones to help to organize the main types of innovations produced by service systems”. Additionally, Bohmann, Leimeister, & Moslein (2014) recognized the need that SS design includes innovation. The architecture model of the next section explicitly recognizes the role of innovation and provides ways that allow the design of mechanisms that implement such a role.

Another approach used in this paper is “artifacts as a research contribution” in Design Science Research (DSR), as proposed by Gregor & Hevner (2013). They define an artifact is “a thing that has, or can be transformed into, a material existence as an artificially made object (e.g., model, instantiation) or process (e.g., method, software)”. Additionally, “an artifact and its description in terms of design principles and technological rules are steps in the process of developing more comprehensive bodies of knowledge or design theories”. Associated with these ideas, there is a kernel theory that “refers to any descriptive theory that informs artifact construction” and “design knowledge should include a kernel theory because such theory explains, at least in part, why the design works”. A kernel theory “arises in disciplines outside the domain of interest”, which means that it uses or integrates knowledge from such disciplines to develop a design theory. In connection with these ideas, this paper proposes the following methods:

- **Design artifacts** in the form of patterns, models, and design principles and technological rules for complex service design,
- A **kernel theory** that justifies and provides tools for making designs work, which uses ideas from all the disciplines reviewed in Section 2.3.
- First ideas for developing a more comprehensive service structure design theory based on patterns.

For determining the value of SDR, Gregor & Hevner (2013) have proposed criteria such as: “Interest of problem; contribution of problem solution to advance knowledge of theory, methods or applications; solution to problem or contributions toward a solution that improves substantially upon previous work; and new solution methods or solution methods that can be used to solve other problems of interest and clarifies the understanding of an area of research or application”.

Some of these criteria are used for evaluating the contributions of this work in Section 6.
4. REFERENCE ARCHITECTURE AND DESIGN METHODOLOGY

In this section, the main artifact of this paper is developed and a methodology for its use is proposed.

4.1. Main Artifact: Reference Architecture

The first simple aggregated general service architecture pattern considered, an abstraction based on many practical cases (Barros, 2019), defines the basic subsystems that are subject to design in any service. The architecture, shown in Figure 2, features as the key subsystem the “Service Production Flows” defined previously, which design determines all the other subsystems. Over such flows, there are “Flow Management Processes” that take care of, among others, the management of flow’s operation, resources, and innovation. Below “Service Production Flows”, “Information Systems”, and four “Intelligent Support Levels” provide support for the execution and management of the flow. This pattern, although developed independently, is consistent with the SS definition formalized in Figure 1, where the “Service Production Flows”, “Information Systems”, and “Intelligent Support Levels” provide a decomposition of the “Infrastructure Network” in such figure. The management components of the SS definition and the pattern are also equivalent.

The value of the pattern is that provides a first approximation to the relationships that the “Infrastructure Network” of Figure 1 contains, further defined in a more detailed architecture model later. As shown subsequently, relationships include all the “Substance” participants in creating the service and are one of the main design objects. The four components of this aggregated architecture are part of the comprehensive structure design defined previously.

An architecture that decomposes the elements of Figure 2 is necessary to represent designs in situations where there are several “Service Production Flows” with many variations. The experience in the cases aforementioned in this paper allows proposing such an architecture pattern, the main artifact of this proposal, as discussed next.

Figure 2. Aggregated SS architecture pattern
As defined previously, there are several levels in the architecture and an explicit hierarchy produced by the modularized decomposition of an organization’s services into packages, components, and subcomponents, presented in Section 2.3. Besides, other factors observed in practice exist, such as the shared services, other than service components, which can be the ones centralized at the organization level, e.g., financial services; the services that various packages may use, e.g., supplies and IT; and the services shared by components, e.g., examination services used by several components of a medical flow. This structure defines a general hierarchy, made explicit in the multilevel architecture given subsequently.

Levels can be managed in a centralized way, in the style of the Unification structure presented in Section 2.3, or with several degrees of decentralization, up to fully decentralized, where lower-level units have full control over their activities, as in the Diversification structure. Thus, decentralization in architecture design is a key decision. The economic analysis behind this decision uses agency theory, which advises balancing the cost of decentralization, mainly residual loss due to results not according to the principal’s interests, and the cost of all the management machinery necessary to run a business in a centralized way. Modern advanced Analytics, plus ever-decreasing IT cost, have made less expensive the centralization approach. This makes it possible to have structures with processes that follow a centralized Analytics-based business logic, which design comes from central management and includes optimization criteria considering the principal’s interests. However, such processes appear as decentralized because they are directly executed by operators who receive instructions or advice from IS, producing actions according to the principal interests, as shown in the model in Figure 2.

A clear difference exists in managing strategic planning and innovations that improve the service, as compared with the operational issues, so they may have different solutions in the design of the structure. Then, in some enterprises, strategic and business development issues may be naturally centralized, but operational management may be more decentralized. For example, the Replication experience of CEMEX, an international firm in the cement business, is that it centralizes all the innovation for new capabilities development and provides standardized production and management processes to its locations, including IS support, but the operation is decentralized (Ross, Weil & Robertson, 2006). In some cases, where coordination among several operational facilities is necessary to produce global optimum results, some operational decisions may be centralized; for example, the Unification experience of Walmart that centralizes logistics management for several selling facilities in a region using the intelligent support “Level II” of the model of Figure 2 (Barros, 2019).

Based on the characteristics above, the architecture pattern proposed is in Figure 3. This pattern is an artifact that serves as a Management and Enterprise Architecture Framework (MEAF) that is used as a reference to design complex modularized services, with multiple hierarchical levels, as the cases of the next section will show. It is a formal BPMN model, which includes important decompositions of the elements of Figure 2, as defined subsequently.

i) The “Service Production Flows” decompose into several “Package j”, each having various “Component k”, and several “Shared Services Components” represented by pools decomposed into lanes at the bottom of the figure, following BPMN and the modularity definition of components presented in Section 2.3. The model also decomposes the production flow or the “Infrastructure Network” of Figure 1, with independent flows as described in the same section.

ii) All the other pools above the production flow give a detailed view of the “Flow Management Processes” of Figure 2 that are needed to manage operations, and also define and make innovations on the service according to given strategic plans. They also are a decomposition of “Management” of Figure 1.

iii) IS and Intelligent Support are part of the “Shared Service Components”.

A more detailed description of the architecture is as follows.
“Package j” pool is just one typical package, but there may be several. As presented in the cases later, all the components of a package are a partition of the pool into lanes, symbolized by a typical “Component k”, of the many a “Package j” may have. The service “Production Flow k” for a component appears as just one process, later decomposed using the facilities of BPMN into a workflow that specifies subcomponents to execute the service provided by each component; therefore, a package is a hierarchy of components and subcomponents. For each final component or subcomponent of the hierarchy, there is a service provider.
The model also defines a pool for “Operating Unit Service Package j” that takes care of the direct management of Package j. As shown in Figure 3, it includes lanes for:

- **Operations** management, such as operations scheduling, resource provision, performance control, corrective instructions, and detecting the need for production changes,
- **Innovation** in the production flow; for example, managing improvements originated in production observations as continuous innovation or changes determined by a higher level,
- **Planning** of operations based on instructions coming from higher levels or local control information.

A typical “Division i” and “Enterprise” levels are modeled to provide the management necessary to assure that the service is provided according to strategic guidelines and that the right innovations are executed to generate the capabilities necessary for a changing business environment. Additionally, they provide the necessary coordination among Divisions and different Operating Units. For this, basic processes of “Planning”, “Innovation” and “Operation” are replicated and instantiated according to the level needs. These levels will be present in the cases to show how their tasks relate to the design performed over the production components.

“Shared Services Components” give services to all “Package j” components, e.g., supplies and IT, including intelligent support, and there may be many of these. Outsourcing of some of these components to external suppliers is possible and the same representation with a separate pool is adequate in such a case; the coordination with them utilizing interfaces uses information flows for its representation.

The centralization design issue, analyzed in Section 2.3, is present in Figure 3 in the flows between structure levels or pools. Such flows may go from very precise and detailed plans, programs, and instructions, in the centralized option, to general policies and guidelines in the decentralized alternative. Thus, some of the key design issues considered in applying this model are as listed subsequently.

i) How to distribute “Operations”, “Planning” and “Innovation” among levels of Figure 3 is an important design decision, which may be the responsibility of the “Enterprise” level in some situations; but in some cases, where the Diversification model applies, they can be decentralized. Economic analysis is a way to decide on this issue. Centralized management in the line of Unification of some operations is a possibility, since there may be coordination and share of experience issues among “Divisions” and “Operating Units” that well managed, with appropriate Analytics, can generate large benefits, with moderate costs of developing processes and systems needed.

ii) The design of interfaces is the key to assure that recombination works. The model in Figure 3 allows two alternatives for this. First, the arrow “Improvement to flows” allows including very detailed service specifications that define the flow, physical predefined flows, and protocols to evaluate needs that determine the flow. These instructions, supported by IS, means that the service provider in “Production Flow k” can route the flow in a decentralized way. The other possibility is that “Operations management Unit j” performs explicit coordination locally and decides about the flow; this may include the use of tools to support such decisions, as in the decentralized case, but also explicit analysis of the state of the customers and service provided may be included. These alternatives are not mutually exclusive and both may be part of a particular design, as shown in the next section.

iii) The design in the previous item requires continuous innovation, due to the formalization of relationships and coordination contained in interfaces that determine the flow. Thus, performance must be evaluated and necessary changes discovered to correct flow behavior when necessary. The model considers this in the “Innovation” lanes at several levels; for example, at the “Operating Unit Service Package j” pool, in the “Innovation” lane, there is the process “Innovation in
production flow” that performs continuous observation of the flow and corrections in its design. However, at higher levels, there are also “Innovation” lanes, which participation and importance depend on the dynamics of the service. If the service is very stable, necessary changes are minor and managed in a decentralized way, but in highly dynamic services – due to frequent demand changes or the need to innovate the service to stay competitive – all the levels should be involved in innovation. The roles and processes dedicated to such innovations are present in the case of elderly services of the next section.

iv) Share of services is another design decision because there are usually several services possible to centralize in the proposed structure. Supply management is an obvious example because there are many benefits in its centralization in enterprises that consume similar items. Logistic services are another possibility, including outsourcing. Another obvious candidate for centralization and outsourcing are IT services, an option taken by many enterprises all over the world.

Notice the integration in the kernel theory behind the pattern artifact of all the disciplines presented in Section 2.3, providing descriptive, prescriptive, and evaluative methods. Thus, modularity describes the structure of the production flow or “Infrastructure Network”, enterprise architecture ideas are present in providing guidelines for alternative management structures and a formal modeling approach, Intelligence provides alternatives for supporting or automating the production and management, and economic evaluation provides ways for deciding the degree of management centralization.

4.2 Design Methodology

The methodology is part of the kernel theory that prescribes the use of the artifact MEAF in complex services, providing principles and technological rules for complex service design. The steps for its application are the ones listed below.

i) Definition of the structure and design of the modular service packages and components, including its decomposition into subcomponents, which define the production flow. The components’ design includes innovations that may change in a fundamental way the service and the value provided, according to a new business model as shown in the cases of shipping services (Pekarinen and Ulkuniemi, 2008) and “integrated practice units” (Lee & Porter, 2013). Additionally, the use of variants of Analytics, as presented before, for their support or automation is part of this design.

ii) Definition of the relationships and their management by coordinating packages and components using interface design, for which guidelines given in Section 2.3 apply.

iii) Definition and design of activities at the operational, division, and enterprise levels to manage service packages’ coordination and innovation, and alignment with strategic objectives. For this design, the pattern in Figure 3 provides predefined activities and relationships that should be instantiated for the cases under study. Structure’s levels and their responsibilities should be determined. Besides, for each relevant level, operation, innovation, and planning activities designs are necessary, according to centralization or decentralization design guidelines; these designs should consider using variants of Analytics, as defined before, to automate or support such activities.

iv) Finally, for designs defined above, IT requirements for their support or automation needs definition and specification.

5. APPLICATION OF THE MANAGEMENT AND ENTERPRISE ARCHITECTURE FRAMEWORK

Next, the task is to show that the proposed architecture design pattern is general enough to represent the designs of three cases, sufficiently documented in the literature, as instantiations of the model
proposed in the previous section. Additionally, to apply the design methodology and appraise the contribution it makes to improve the designs of the cases.

5.1. Services to the Elderly Case

Three papers describing designs of services to the elderly in Holland, the most documented case of design using modularity, are considered. First, a paper by Soffers, Meijboom, van Zaanen, & van der Feltz-Cornelis (2014) that centers on mental care presents the basic components for such care. Another by de Blok et al. (2014) proposing a typology for interfaces, based on home care services. Lastly, one by Broekhuis, van Offenbeek, & Eissens-van der Laan (2017) providing a component classification that considers mental care and physical problems. From these papers, this work concentrates on mental problems, including medication, and some home services, to synthesize their activities definition in a service production flow. To simplify representation, some of the details described in the papers are not considered, but a further decomposed architecture may model them. With this characterization, it is possible to represent the services package and its components as shown in Figure 4, including service providers, where the “Service Package for the elderly components” pool is included with lanes “Medical care”, “Heavy housework” and “Meal on demand”. Only the first component is decomposed, showing subcomponents “Psychiatric care” and “Medication care”. For the first, a service production flow exists, with a set of tasks that act on the patient as needed, executed by the service provider “Care Person”, including the possibility of intervention of a “Care Person Superior” to perform the task “Manage crisis”, if necessary. The component “Medication care” executes “Provide medication”, as coordinated by “Production Flow” using the interface “Medication request”.

The management activities associated with the pool “Operating Unit Service package to the elderly” in Figure 4 are “Operation” with the main process “Needs assessment to define package, continuous care coordination”, which represent papers’ emphasis on the interface needed to assure a correct definition of the package required for a particular user. Nurses that visit users perform this process and determine the mix of services needed by each of them, generating the interface “Selected components-Work schedules and instructions” that defines the role and coordination of each of the package components. This process permanently evaluates the package assigned to each person with the information provided by “Control information-Need for package reformulation”, generated by care coordinators that visit users every week. The other key process of this pool, in the lane “Innovation”, is “Innovation in production flow (continuous improvement)” that performs a work of structure definition in connection to package design and interfaces. It generates “New flow design including rules-Product Book and pre-combined elements”, which reconfigures a package and components to guide the selection of component combinations to satisfy needs, and rules for the order in which to provide them, or the selection of individual components for particular cases. All these definitions determine the design of the elderly service production flow, designing practices, including health care and coordination utilizing interfaces. To support this effort, the flow “Innovation needs” reports problems and suggestions from “Operation”. Additionally, this process receives instructions from the lane “Innovation” of the pool “Government Social Services” through “New services and improvements to implementation”. This innovation is important since users’ needs may change and new technologies and procedures may become available; for example, formal learning about elderly behavior using AI models can provide support for the decisions about the best actions to take on patients.

At the lowest pool of the model, there are some shared components identified as “Pharmacy” that delivers medications needed by other components, and “IT support” providing a digital “Care Dossier” to all other components, which are not explicitly identified in the papers describing these services.

The papers describing this case do not differentiate among operating and innovation activities and define them as stages of a unique process. Innovation is a management activity, but the papers mix it up with the tasks that execute the service. Separating them, as done here, is to recognize that they occur in different periods: operation is short time oriented to manage the day-to-day, using given rules and procedures, and innovation has a longer-range perspective, including structural changes.
on how to do the work. This has consequences in defining the organizational setup, including who performs the service, manages and redesigns it as needed, and the methods and abilities required.

Economic analysis justifies this solution since agency theory favors decentralization when it is possible to design the service following modularity, with precise rules and procedures that define work in a way that interprets the principal; in this case, government. Additionally, innovation activities at two levels favor continuity, since they guaranty that the design stays current when changes in the demand or the political environment happen. The additional value provided by the methodology is discussed in Section 6.
For more details, further needed decomposition into subcomponents, not detailed by the model, is necessary, which the modeling approach facilitates with hierarchical decomposition, exemplified in the emergency case. All the logic behind service execution is formal and should be detailed using BPMN, as will be also exemplified in the emergency case.

5.2. Services to the Unemployed

A case developed for municipal services in Denmark that provide job support to the unemployed (Harmon, 2010) is considered. Modularity is included in the design, which structure is in Figure 5, with components as listed below.

i) Service package “Job Support” has components “Support unemployed with jobs and education” and “Support sick return to the job market”.

ii) Component “Support unemployed with jobs and education” has a production flow that includes the processes shown in Figure 5 in the order in which they occur; but, for a particular case, selecting some of them according to user’s requirements is required, which is to be detailed by further decomposition of the flow.

iii) Component “Support sick return to the job market” has processes “Evaluate readiness” and “Evaluate working ability”.

iv) “Shared Services components” are used; a sample of the “Support component” processes is in Figure 5. The “IT support” component is also present as explained below. The arrows “Request” and “Service” act as interfaces to coordinate these processes, to be further detailed by decomposition.

In the above components, the design of the service production flows is present in “Support unemployed with jobs and education” and “Support sick return to the job market”, which defines the processes that may be used, in different combinations, to perform any of the services provided.

The “Shared Services components”, in different configurations, allow executing several service variations that determine customer’s flow. They use the interface flows “Request” and “Service”, which provide a logic defining the relevant shared component for a case and how to provide them.

An operations management component must be present for the “Job Support” package to perform properly, included in the pool “Job Support Package operation”. The lane “Operation” of this pool contains the process “Evaluate need- Manage cases”, which receives “Activity and control information” to assure the satisfactory provision of all requested services, detecting problems that may require cases corrections or activation; also discovering the need of service flow redesign is part of the job. Instructions in the interface “Programs and instructions” define the service. The information on “Innovation needs” informs the requirement to do a redesign to the process “Innovation in production flow”. Such redesign generates “Improvements to flows” that are implemented in each “Service Flow”. These processes are regulated by the “Danish Municipalities Government” pool, which provides “Coordination among services”, “Plans” and “Changes to operations” to assure that services perform as defined by the government. These two management levels are not included in Harmon’s (2010) paper, but they are derived from the pattern; it is clear that their explicit design is necessary to correct short-term operational malfunction and perform continuous evaluation and innovations to adapt the service to changing conditions.

Additionally, explicit IT support is present to have the possibility of knowing the state of each user, processing the “Activity data”, allowing for control and activation of the flow, using “Activity and control information”, which is not considered in the author’s design.

Economic analysis justifies this solution since coordination theory and modularity justify specialized production flow lines for different service flows –unemployment and sick support– given to the same population, which can share, as the original paper discovered, several components and processes coordinated using interfaces.
5.3. Services in an Emergency Department

This case uses a paper (Barros, Riffo & Paredes, 2019) that reports the work of a project developed for the complex San Juan de Dios Hospital for adults in Santiago, Chile; this is a large hospital with 62,000 emergency department (ED) yearly admissions. The paper covers the first part of the work that concentrates on the production flows design. A second part, reported in Barros (2019), explicitly considers the architecture design problem covered in this paper.
For the first part, the hospital had digital clinical records data for July-December 2016, which allowed the authors to perform an analysis of the patient flow behavior to determine the best opportunities for design innovations. This hospital uses a classification of patients according to disease severity (Triage) from M1 to M5, where M1 is the most serious. The conclusion was that “the demand and amount of service hours were very concentrated M2 and M3 patients who used 90% of the resources, justifying the focus on them”. Regarding the most relevant diagnoses, “three diagnostic groups accounted for 74% of the bed days in the ED: neurology, cardiology, and broncho-pulmonary”; this justifies formal consultation protocols to accelerate the diagnosis for these groups. Complementarily, other analyses of the flow showed that “waiting to confirm the diagnosis resulted in attention stagnation, reevaluation of the patients produced unnecessary delays, merging of M2 and M3 patients was common during the reevaluation process, and traceability of patients in the interior of the ED was a constant unresolved issue”.

After this identification of problems, the authors determined the design modifications, including facility architectural changes, necessary to solve them and obtain a better structure for the patient’s care flow. Thus, based on modularity proposals, they designed components for “independent medical and surgical flows for each of the groups of patients categorized as M2 and M3”. The designed flows “needed exclusive areas for patients M2 and M3 with dedicated staff: waiting room, specialist consultation, patient reevaluation, treatment room, and hospitalization box”. This required “significant remodeling of facilities, including stations for nurses in each flow”. Following modularity, they found that “the interactions among the new flows and other service components, such as examinations and inter-consultation (IC) specialists, required the design of interfaces, which took the form of complementary practices”. For example, the new interface design for “specialist consultation that includes a manager role to handle requests and executions of IC” and formalizing various “IC medical specialties requests using service agreements properly managed and controlled”.

Additionally, they developed “clinical protocols for IC specialist consultation for selected pathologies”. The protocols emphasize “medical workflows that allow solving the problems arising in the gray areas of clinical decision-making. They incorporate the experience of the best MD, formalized as a logic diagram”. This is a good example of Intelligent “Level I” support of Figure 2 and opens the way to the use of a more complex predictive Machine Learning model of “Level II”, which is possible to develop with the data collected.

Modularity is present in the proposed design, which structure is in the model of Figure 6 that is based on a model taken from Barros (2019), complemented to include management components not originally included. Details are explained subsequently.

i)  “Emergency Service Package” has the independent components “Triage”, “Line M1”, “Line M2”, “Line M3” and “M4-M5 component”,

ii) For the “Line M2” component, there is the process “M2 flow”, which has a production flow that is detailed in the original paper (Barros, Riffo, & Paredes, 2019) and replicated in Figure 7, modeled in BPMN that represents the logic defining its execution. Other components may have a similar decomposition to show the details of their flow.

iii) The related “Inpatient care Service Package” is included, which function is receiving patients that ED cannot cure.

iv) Activities shared by the previous components are present; they are “Shared Services components” and include, as explained previously, “Exam Service” and “Inter Consultation Service”. Also included is the shared activity “IT support” since there is an IS that collects all the information generated in the production flow components. This information has potential use in the “Emergency Operation” and “Hospital Management” pools to improve service performance.

v) Service providers are included for each component.
In Figure 6, management is present with an “Emergency Operation” pool, including an “Operations management” process, which, at least, should perform resources’ management. An important process is personnel management since production lines’ design determines human resources based on peak demand. Thus, a process is necessary to monitor the load on these resources, to adapt them to current requirements. For other resources—such as supplies, beds, and equipment—a similar process should exist to assure operations’ continuity. This process generates “Schedules and instructions” that guide the “Lines” and performs control to assure their satisfaction. The next level is the pool “Hospital Management” with the lane “Operation” and the process “Central operation” in charge of defining “Programs, goals, and budgets”. This process guides “Operations management” and perform control to assure that directives are met; besides, it generates “Coordination instructions and resource definition” for the “Lines, Inpatient Care and shared services components”. Additionally, to perform continuous adaptation and innovation of the production design, the lane “Innovation” is defined with the process “New capabilities development” for redesigning the production flows as needed, according to “Policies, Plans, and Programs”, “Requirements” from “Operations management” and “Control and evaluation information” coming from “IT Support”. This process generates the flow “New flow designs, protocol changes and instructions” defining new capabilities for “Lines, Inpatient Care and shared service components”.

Not all these management processes were included in this case’s original papers. Thus, the value of the pattern is to show the need of having them in the design to assure good management of resources, and innovation needed to assure that production operates as specified and changes needed to adapt to new situations occur. As to design justification, this case validates the economics theory value of specialized production flows; the original paper evaluates savings associated with efficiency improvements of the new design as US$250,000 annually, not including the value of the better service provided, which reduced patients’ waiting time by 30%.

6. EVALUATION AND CONCLUSIONS

Here, the evaluation of the proposed research approach, using the guidelines presented in Section 3.2, is performed.

First, the evaluation of compliance with the research opportunities established by Frost and Lyon (2017) is evaluated. They discovered a “lack of empirical evidence for design proposals”. The proposal presented previously uses this opportunity because architecture patterns come from formalized real-life case experience. Another research opportunity they found is to “create a service system framework which offers descriptive, prescriptive and evaluative analysis methods”. The methods proposed and applied to the cases offer a descriptive proposal for a general architecture pattern for complex SS, a prescriptive methodology of how to generate designs for such systems using the pattern, and a method to evaluate design alternatives and chose the one that offers the best value for users. Thus, it complies with this opportunity. Additionally, they recognized the need that SS design includes innovation. The architecture pattern and cases explicitly recognize the role of innovation and provide ways that allow the design of mechanisms to make such innovation work.

Of the factors proposed by Gregor and Hevner (2013) to evaluate research, the ones relevant for this work are the following.

A “solution to problem or contributions toward a solution that improves substantially upon previous work” is present. Thus, learning from real-life cases in several domains allowed proposing architecture patterns for the problem of a comprehensive design of a SS. Such patterns integrate modularity, enterprise architecture and modeling, Analytics, and evaluation methods. Converted into models, they were applied to three representative situations of such a problem using the methodology proposed in Section 4.2. In each of the cases, the conclusions, as detailed previously in this section, are that formal modeling of the situation under design proved feasible, with clear design guidelines, which allows discovering several important issues not considered in original papers. In particular,
Figure 6. Architecture for emergency services
Figure 7. Decomposition of the M2 flow
modularity considers interface design, very much related to the design of coordination mechanisms for the interactions of the quasi-independent production flows modularity proposes and shared services they have. Additionally, the coordination with management levels is part of this design as shown in the cases. This includes a design of the degree of management decentralization, for which specific guidelines, based on economic theory, are available. An important management factor discovered in the cases is innovation, for which a good design, with clear responsibilities at different management levels, assures that the service is continuously adapted to new situations. The formal design of all these coordination issues allows the explicit design of Analytics and IT support. The issues just described were only considered informally or not considered at all in the original cases. Besides, no current method, as reviewed in Section 2.1, considers all the design aspects included in the methodology proposed here. Therefore, the value of the proposed approach is the design of solutions that consider explicitly and formally several issues, not integrally incorporated in other methods, which favor given better service and eliminating inefficiencies.

The second contribution of this work is “advance knowledge of theory, methods or applications”. Thus, the development of design patterns using real-cases experience, well known in software and processes, has proved feasible for complex services comprehensive structure.

Finally, this work advances in another contribution, which is the “understanding of the area of research and application” of SDR, by proposing a new challenge: the design of a comprehensive structure of complex services, only suggested in the literature reviewed in Section 2.1. The problem definition and formal design approach presented in this paper open the way to improved definitions and alternative or complementary design methods.

Thus, the proposed MEAF approach for complex SS design makes a significant research contribution and meets the criteria proposed by Gregor & Hevner (2013). Additionally, the advantage of the proposed architecture pattern and methodology for designing modularized production flows in services of this type is clear. Besides, the design of the management of such flows, for several organizational levels, follows from the production design. A closing conclusion of this paper is the value of having a frame of reference which can be instantiated for a particular case, like the one proposed here, for producing results in much shorter times than the ones usually experienced in projects of this type and the possibility to evaluate designs based on economic theory.

As to limitations of the proposed approach, the formalization of knowledge derived from actual cases requires more validation using a representative set of other complex services, for which its application starts from current situations with no design. Work has already started in this line with services to children and education to confirm the results presented here or improve the proposed methods.
REFERENCES

Arrow, K. (1985). The Economics of Agency. In J. W. Pratt & R. J. Zeckhauser (Eds.), Principals and Agents: The Structure of Business Cambridge. Harvard Business School Press.

Baporikar, N., & Randa, I. O. (2020). Organizational Design for Performance Management in State-Owned Enterprises. International Journal of Service Science, Management, Engineering, and Technology, 11(4), 1–25. doi:10.4018/IJSSMET.2020100101

Barros, O. (2017). Service Design with Applications for Health Care Institutions. Business Expert Press.

Barros, O. (2019). A process architecture pattern and its application to designing health services: Emergency case. Business Process Management Journal, 26(2), 513–527. doi:10.1108/BPMJ-08-2018-0210

Barros, O., Riffö, R., & Paredes, I. (2019). Improving service in an emergency department by designing the health production flow. Health Services Management Research, 33(2), 76–85. doi:10.1177/0951484819860325

Becker, M., & Klingner, S. (2013). A Metamodel for Component-Based Service Modeling. International Journal of Service Science, Management, Engineering, and Technology, 4(2), 60–77. doi:10.4018/jssmet.2013040105

Boehmann, T., Leimeister, J. M., & Moslein, K. (2014). Service systems engineering: A field for future information systems research. Business & Information Systems Engineering, 6(2), 73–79. doi:10.1007/s12599-014-0314-8

Broekhuis, M., van Offenbeek, M., & Eissens-van der Laan, M. (2017). What professionals consider when designing a modular service architecture? International Journal of Operations & Production Management, 37(6), 748–770. doi:10.1108/IJOPM-05-2015-0306

Cesarotti, V., Giuiusa, A., Kwan, S. K., & Spohrer, J. (2014). Designing Multichannel Value Propositions to Enhance Value-Cocreation Phenomenon. International Journal of Service Science, Management, Engineering, and Technology, 5(1), 14–44. doi:10.4018/ijssmet.2014010102

de Blok, C., Meijsboom, B., Luijkk, K., Schols, J., & Schroeder, R. (2014). Interfaces in service modularity: A typology developed in modular health care provision. Journal of Operations Management, 32(4), 175–189. doi:10.1016/j.jom.2014.03.001

Dorbecker, R., & Boehmann, T. (2013). Concepts and Effects of Service Modularity—A Literature Review. In 41th Conference on System Sciences. IEEE Computer Society. doi:10.1109/HICSS.2013.22

Frost, R., & Lyons, K. (2017). Service Systems Analysis Methods and Components: A Systematic Literature Review. Service Science, 9(3), 219–234. doi:10.1287/serv.2017.0180

Gregor, S., & Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. Management Information Systems Quarterly, 37(2), 337–355. doi:10.25300/MISQ/2013/37.2.01

Harmon, P. (2010). Danish Municipalities Case Study. BPTrends. www.bptrends.com

Joly, M. P., Teixeira, J. G., Patricio, L., & Sangiorgi, D. (2019). Leveraging service design as a multidisciplinary approach to service innovation. Journal of Service Management, 30(6), 681–715. doi:10.1108/JOSM-07-2017-0178

Lee, T. H., & Porter, M. E. (2013). The Strategy That Will Fix Health Care. Harvard Business Review, 91(12), 1–19.

Malone, Th. W., & Crowston, W. H. (1994). The Interdisciplinary Study of Coordination. ACM Computing Surveys, 26(1), 87–119. doi:10.1145/174666.174668

Mora, M., O’Connor, R., Raisinghani, M. S., Macías-Luévano, J., & Gelman, O. (2011). An IT Service Engineering and Management Framework (ITS-EMF). International Journal of Service Science, Management, Engineering, and Technology, 2(2), 1–15. doi:10.4018/jssmet.2011040101

Patricio, L., Fisk, R. P., Falcão e Cunha, J., & Constantine, L. (2011). Multilevel Service Design: From Customer Value Constellation to Service Experience Blueprinting. Journal of Service Research, 14(2), 180–200. doi:10.1177/1094670511410190
Oscar Barros (PhD, U. of Wisconsin) is Full Professor at the Department of Industrial Engineering of the U. of Chile. He created the first group of Operations Research in Chile and the first Master in the discipline. He also developed the discipline of Information Systems, including a Master on the subject. Dr. Barros is now dedicated to the development of an original methodology for the design of businesses and services, including the creation of the novel and recognized Master in Business Engineering. He also directs an applied research program that is applying the ideas of Business Engineering to health services, which has produced significant impacts on productivity and quality of service. He has written 15 books, with more than 130,000 copies sold on the subjects of Operations Research, Information Systems, Information Technologies, Process Reengineering, Business Engineering and Service Engineering. He has also published widely in international scientific journals.