An Approach for Automated Service Selection and Ranking using Services Choreography

Firmino Silva, Claudia-Melania Chituc and Paul Grefen
Eindhoven University of Technology, Eindhoven, The Netherlands

Keywords: Service Selection, Service Ranking, Service Choreography, SOC, Adaptive System.

Abstract: In today's highly competitive market, it is critical to provide customers services with a high level of configuration to answer their business needs. Knowing in advance the performance associated with a specific choreography of services (e.g., by taking into account the expected results of each component service) represents an important asset that allows businesses to provide a global service tailored to customers' specific requests. This research work aims at advancing the state-of-the-art in this area by proposing an approach for service selection and ranking using services choreography, predicting the behavior of the services considering customers' requirements and preferences, business process constraints and characteristics of the execution environment.

1 INTRODUCTION

Changing market conditions, increased competitive pressure, and growing demands and expectations of clients make companies rethink the way they perform business to provide customers with services that offer a high level of configuration in order to parameterize their particular needs. Allowing customers to configure the requested service(s) and providing an accurate estimation of the results (e.g., in terms of cost, time to complete the service, level of quality) is of great importance to them. However, designing, developing, deploying and monitoring a system to support such characteristics is challenging. Service-oriented computing (SOC) and cross-organizational business processes provide the means to build and run dynamic business environments addressing the constantly evolving customers' requirements (Papazoglou et al., 2006) (Di Nitto et al., 2009) (Wetzstein et al., 2010).

The work-in-progress presented in this paper aims at advancing the state-of-the-art in this area by proposing a method for service selection and ranking using services choreography, predicting the behavior of the services considering customers' requirements and preferences, business process constraints and characteristics of the execution environment. Based on previous work (Silva and Chituc, 2013), the proposed approach relies on a mechanism for monitoring different metrics measured at different levels of the choreography, considering the prospect of satisfaction of the customer and provider. We enhance our existing framework by including a control mechanism based on closed life cycles (Hellerstein et al., 2004) derived towards productive responses. This mechanism enriches the historical collection of results of past events to estimate, through decisional elements, the behavior of the choreography. The control mechanism follows a hierarchical model that consists of three different levels of control responsibility: strategic, tactical and operational level, considering the functional scope of each framework element (Kaplan and Norton, 2008). This work contributes to provide a conceptual adaptive (Landau et al., 2011) framework of services that ensures a high degree of predictability for the services' choreography

This research project started with an extensive literature analysis where relevant topics for this research project were selected. In-line with the results of the analysis pursued, a framework was designed, which aims at supporting the selection of the most suitable set of services available at a given time to answer customers' requirements and preferences, taking into consideration business process constraints and the characteristics of the execution environment. The paper is organized as follows. The adaptive framework for automated service selection and ranking using service choreography is presented in Section 2. The matrix

Silva F., Chituc C. and Grefen P..
An Approach for Automated Service Selection and Ranking using Services Choreography.
DOI: 10.5220/0005461202590266
In *Proceedings of the 11th International Conference on Web Information Systems and Technologies* (WEBIST-2015), pages 259-266
ISBN: 978-989-758-106-9

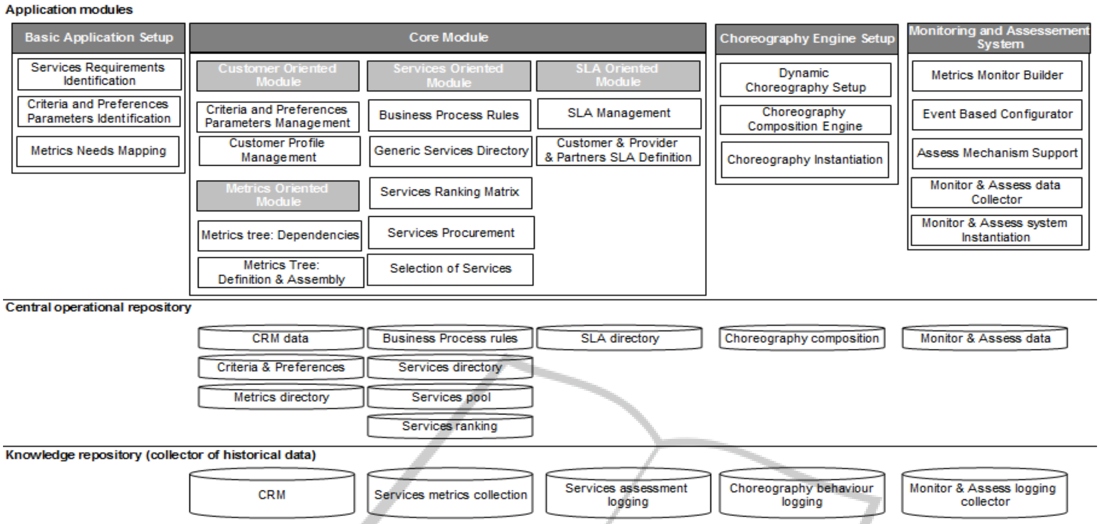

Application modules
Basic Application Setup
Services Requirements Identification
Criteria and Preferences Parameters Identification
Metrics Needs Mapping
Core Module
Customer Oriented Module
Criteria and Preferences Parameters Management
Customer Profile Management
Metrics Oriented Module
Metrics tree: Dependencies
Metrics Tree: Definition & Assembly
Services Oriented Module
Business Process Rules
Generic Services Directory
Services Ranking Matrix
Services Procurement
Selection of Services
SLA Oriented Module
SLA Management
Customer & Provider & Partners SLA Definition
Choreography Engine Setup
Dynamic Choreography Setup
Choreography Composition Engine
Choreography Instantiation
Monitoring and Assessement System
Metrics Monitor Builder
Event Based Configurator
Assess Mechanism Support
Monitor & Assess data Collector
Monitor & Assess system Instantiation
Central operational repository
CRM data
Criteria & Preferences
Metrics directory
Business Process rules
Services directory
Services pool
Services ranking
SLA directory
Choreography composition
Monitor & Assess data
Knowledge repository (collector of historical data)
CRM
Services metrics collection
Services assessment logging
Choreography behaviour logging
Monitor & Assess logging collector

Figure 1: Adaptive framework schema - adapted from (Silva and Chituc, 2013).

The matrix of services used for ranking the services is described in Section 3. Related work is discussed next. The paper concludes with a section summarizing the results and addressing the needs for future research.

2 ADAPTIVE FRAMEWORK

The framework presented in this paper (Figure 1) is an enhancement of the approach presented in (Silva and Chituc, 2013). It consists of four main modules and two levels of repositories:

- Basic Application Setup module. This module prepares and defines the basic structure of the information introduced by a customer. The interpretation of the criteria and preference parameters chosen by the customer, the identification of the needed services and metrics requirements are initially mapped for use by other modules.
- Core module. This module centralizes a set of core elements for handling the information collected by customer input and conjugates it with existing knowledge in order to subsequently trigger actions for the implementation of the requested service. It consists of four sub-modules that process guidelines: customer, services, SLA and metrics.
- Choreography Engine Setup module. After the identification and selection of the services better positioned in the ranking matrix, this module will gather the data required to assemble and instantiate the choreography.
- Monitoring and Assessment System module. The definition and mounting of the dynamic event based monitoring and assessment mechanism, in order to measure the selected metrics upon the service acquired by the customer, is instantiated by the elements of this module.
- Central Operational Repository. It stores information that allows the daily management of all the modules of the framework, such as: customer data management (e.g., criteria and preferences), service metrics mapping for a specific customer request, or the data structure composition for monitoring and assessment system assembly.
- Knowledge repository. It stores information reflecting results from various runs from customer requests, e.g., metrics assessments and choreography execution results. This data is collected and organized to enrich the knowledge in the framework.

2.1 The Hierarchical Control Model

As the basis of the control model, we use the hierarchical structure of three levels of control: strategic, tactical and operational (STO) (Mintzberg et al., 1996) (Ackoff, 1990) and the Balanced Scorecard (BSC) (Kaplan and Norton, 1992, 1996, 2008). The BSC model is an example of a closed-loop controller applied to the management of the implementation of a strategy. Basically, a closed-loop control (Hellerstein et al., 2004) is where actual performance is measured, the measured value is compared to an expected value and based on the difference between the two actions are taken as required.

Based on the STO principles and the BSC approach, a central hierarchical structure has been identified that is favoured by a closed-circuit of interactions between policy development and practical implementation. Isolating the central core, the scope to evolve this research work was defined, creating an analogy with overlapping layers of responsibility in a pyramidal shape, allowing to compare the reference STO and BSC models with the approach described in (Silva and Chituc, 2013). Figure 2 illustrates the layers of the proposed hierarchical pyramid.

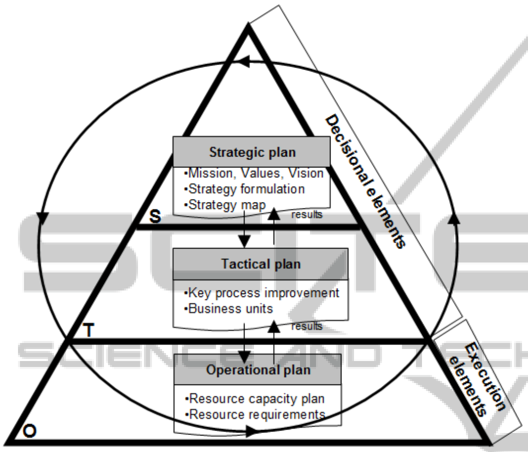


Figure 2: Adopted control model.

The basic execution of this hierarchical structure is driven by a closed life-cycle that adds knowledge and autonomy each time a choreography is invoked. The life-cycle is based on two steps: firstly, by executing elements at the operational level, a response to customer requirements is produced with the performance desired by a customer; secondly, by considering the results of past events, the behaviour of the choreography that will address customers' expectations is estimated, considering the decisional elements residing in the strategic and tactical levels.

2.2 Control Levels and Role Definition

This section describes the roles of each control level for the proposed framework, given its functionality and contribution to the overall performance of the framework. The alignment chosen in order to address the roles of responsibility is the hierarchical model of Figure 2.

Hierachical levels differentiate from each other according the dimensions: content (level of detail of information), time extension (decision influence in time) and scope (impact of the decision) (Mintzberg et al., 1996) (Ackoff, 1990). Elements belonging to the Strategic block have greater range both in terms of scope of influence as in terms of duration of that impact. In order to prepare structures and guidelines for future implementation, elements of the Tactical block are focused on more detailed information to identify the module or specific area of competence, *what* needs to be implemented and *how* the implementation is made. The elements of the Operational block require a critical level of information detail to process and execute services considering customer requirements and preferences.

Strategy tends to flow from the top down and is concerned with long-range objectives and ways of pursuing them that affect the system as a whole. The strategy formulation requires examining where and how the status is now, determining where we want to go, and then determining how to get there (Ackoff, 1990). The main focus of strategic configurations is to growth and enrich the knowledge. Thus, the definition of objectives, rules and basic principles (policies and orientations) resides at this level of the pyramid. The elements of the framework that allow configurations that affect the behaviour of the whole framework belong to this level. Each of the sub-modules of Core module addresses each of the most relevant area of the framework: Customer, Metrics, Services and SLA. These areas of competence are crucial in managing configurations that influence the performance of the framework.

The tactical segment is concerned with shorter-run goals and means for reaching them. Tactics involve the intermediary steps needed to achieve the strategy vision. In this case, the role of these elements is focused on converting the strategic settings (at the highest level) in each of the areas of the framework, either at the level of the services modules (choreography), as at the level of the metrics modules (monitoring and assessment). All the elements which enable the setups and settings oriented to customer requests based on the configurations of the elements of strategic level should belong to this level. The elements included here (e.g., criteria and preferences parameters management, metrics tree definition, service ranking matrix, dynamic choreography setup) are related to the preparation of the conditions, based on the strategic definitions, to be implemented at the operational segment.

The operational level focuses on the systems and procedures to provide the immediate response to the definitions and configurations previously aligned and become operational in this segment of the pyramid. All the elements linked with the instantiation of actions involving the implementation of a specific request received from a customer, at the level of each service, as the evaluation metric of each service, as well as the instantiation of a choreography should belong to this segment, e.g., service requirements identification, metrics definition and assessment. The duration of each activity is sized by the instantiation of the service or choreography and ends when the activity is completed (different compared to previous layers).

2.3 Life-cycle Controls

By linking internally and externally the decisional and operational group of elements (as portrayed in Figure 3), cycles are designed to support the flow the information generated in each instance of each choreography and contributes to increase the knowledge associated with the framework, making it autonomous and adaptable, with higher reliability to future requests.

Decisional blocks comprise elements which can affect (e.g., with their configurations and setups) the whole or part of the framework behaviour. Operational blocks include operational elements of the framework related to specific actions. Each block (Strategic; Tactical; Operational) comprises input data, decisions and target actions, and output results.

Input data. Each framework elements receive information to be processed internally. Input data is received by two types of cycles: internal and external. The internal data flows are those between elements belonging to the same block, which allow feeding different competences. The external life cycles derive from interactions between blocks. They feed the results of the elements of the blocks positioned hierarchically below and contribute to the development of new processing at higher and adjacent blocks.

Decisions => Target actions reflect decisions and resulting actions, e.g., the definition of a specific metrics tree for a given customer request is decided upon criteria and preferences, and as a result of that a structured metrics tree is build according to customer needs.

Output results. Each block receives, as input data, output results from lower blocks. In addition to the information they produce (internal output results), they support the creation of new information, e.g., a specific metrics tree structure which responds to a given customer request, should empower the knowledge of each block so that subsequent interactions are optimized.

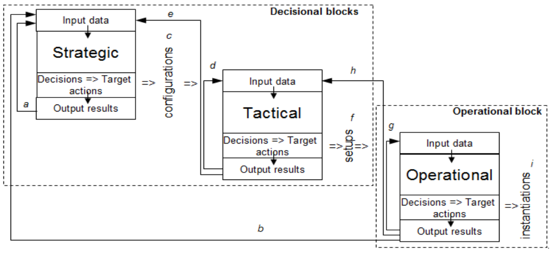


Figure 3: Control dynamic life-cycles.

3 POOL OF SERVICES

3.1 Elements of the Services Oriented Sub-Module

This section describes how services are organized in individual sets (functional pools), the procedures to manage those pools, and the selection of a service to be integrated in a specific choreography.

The Services Oriented sub-Module (Figure 4) supports these activities. It includes elements of relevance for the composition of services and their integration in the choreography. A brief description of each element is below:

- *Generic Services Manager*: This element identifies the needed characteristics as catalogues of each of the services according to the definitions and requirements of the business process rules. Those characteristics will be fundamental to build specific pools where services with the same characteristics will compete.
- *Services Procurement*: Searching and identification of services needed to meet the objective outlined by the client are managed by this element.
- *Services Ranking Matrix*: The functional scope of each service is defined to add each service in the same pool to "compete" in terms of performance within the same type of functionality. Each pool of services is associated to a matrix that stores the ranking of services. Scoring algorithms are then executed to calculate the rating for each service, according to customer preferences.
- *Selection of Services*: The needed services with the best performance indicators are chosen from the ranked pools databases. The algorithm to classify services' performance in previously interactions support the service choreography engine so that it can dynamically build the services better ranked.

This sub-Module actually compiles a list of generic services necessary to address customers' requirements and preferences, manages services and calculates a ranking matrix, based on pools which collect features and services with identical functionalities; identifies and locates services that match the needs of the choreography requested by the client (this function of procurement is crucial to match the characteristics of the services to be integrated in choreography); identifies and selects services whose indices ranking are better positioned to integrate the choreography as potentially giving the best response to the customers' request.

In the Basic Application Setup Module, the basic structure of the information inserted by customer is prepared and defined, to be later used by other modules. Based on the Business Process Rules (from the Core module) and the available Generic Services Directory (from the Core module), the element Services Requirements Identification is responsible to create and identify a Generic Services Requirements List. This list will be used to identify the set of services that will be mapped to the matrix in order to select the best ranked services with those characteristics.

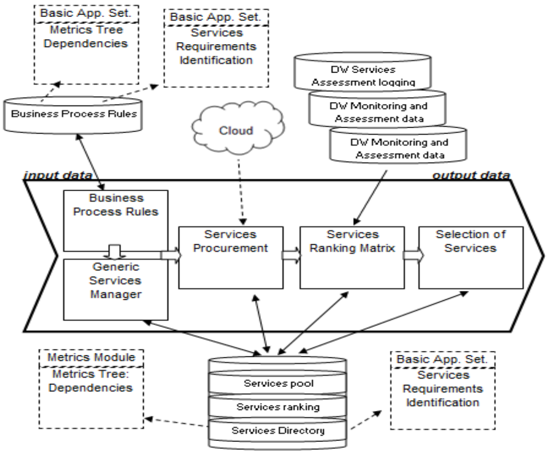


Figure 4: Services oriented sub-Module schema.

An activity of a business process may correspond to a sub-process, with a specific set of attributes and requirements for which a pool of services is identified.

The business process to be assembled (e.g., considering pre-defined business rules) includes all the services that will be executed and contribute to accomplish the customer's request. Thus, as the customer selects the overall service - composed of services provided by business partners, a reference to each of these services is added to the global business process with a set of requirements to identify the specific pool of services.

3.2 Ranking of Services

Each pool of services has associated a matrix (Figure 5) that is built to determine the ranking of the services for a specific function. The matrix accumulates over time the result of measurement of metrics, from all the iterations resulting from their use in choreographies, and is managed so it can be adapted to the customer request by assigning weights (Silva and Chituc, 2013).

The weights correspond to the customer's preferences. Each service has a set of metrics to assess its performance. The weights allow these assessments to be measured at a given time. Thus, the matrix rank can be oriented to what the customer business needs. Different scenarios may be constructed generating different matrices.

In Figure 5, three services are illustrated (services *A, B, R*), and metrics associated to each pool of services are referred (e.g., *m1, m2, m3, mp).* Each metric supports a set of values which represents the domain of values that can result from the evaluation at a given time for a service - m represents the minimum value and M the maximum. Measured values are registered in the matrix for each of its iterations in choreography (*iter 1, iter 2, iter n*). Each service has a number of iterations that match its selection for integrating the choreography, and the number of iterations of services in the same pool (examples: *n, u* and *v*).

3.2.1 Performance Coefficient

Figure 5 shows a pool of services of type ϕ, with four metrics whose range of values is indicated in parentheses. For each metric, a natural average of values from previous iterations is obtained.

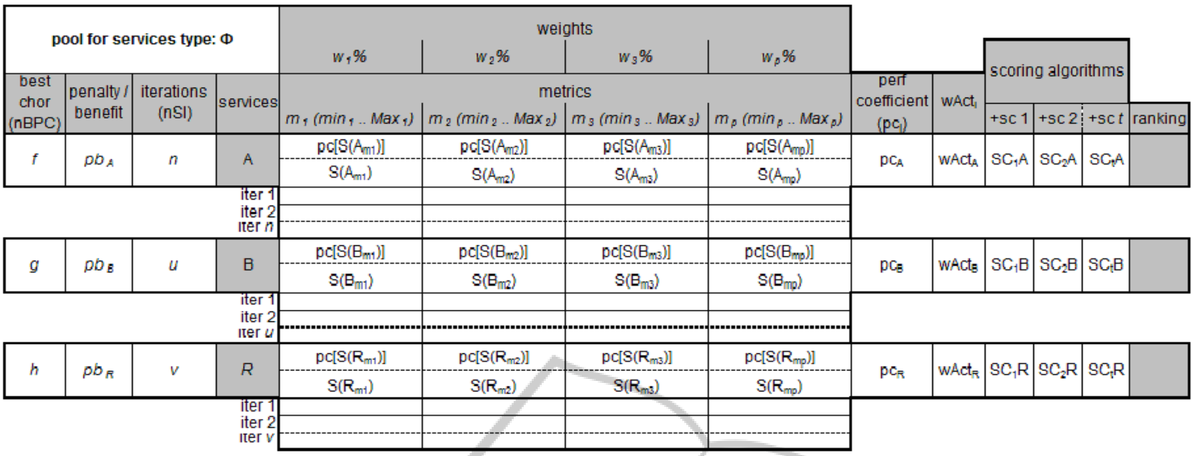
The expressions for calculating each scoring rule is presented using the service A as an example.

$$S(A_{m1}) = (\sum_{i=1}^{n} iter.Value_{(i)}) / n \qquad (1)$$

n: number of runs of service *A* in choreographies;

iter.Value$_{(i)}$: is the assessment value measured on each iteration of service *A*;

S(A$_{m1}$*)*: is the average of occurrences (n) of measurements values (iter.Value (i)) for a metric m_1 for service *A*.



pool for services type: Φ				weights									
				w_1%	w_2%	w_3%	w_p%			scoring algorithms			
best chor (nBPC)	penalty / benefit	iterations (nSI)	services	metrics				perf coefficient (pc_i)	$wAct_i$				
				m_1 (min_1 .. Max_1)	m_2 (min_2 .. Max_2)	m_3 (min_3 .. Max_3)	m_p (min_p .. Max_p)			+sc 1	+sc 2	+sc *t*	ranking
f	pb_A	*n*	A	$pc[S(A_{m1})]$ $S(A_{m1})$	$pc[S(A_{m2})]$ $S(A_{m2})$	$pc[S(A_{m3})]$ $S(A_{m3})$	$pc[S(A_{mp})]$ $S(A_{mp})$	pc_A	$wAct_A$	SC_1A	SC_2A	SC_tA	
			iter 1										
			iter 2										
			iter *n*										
g	pb_B	*u*	B	$pc[S(B_{m1})]$ $S(B_{m1})$	$pc[S(B_{m2})]$ $S(B_{m2})$	$pc[S(B_{m3})]$ $S(B_{m3})$	$pc[S(B_{mp})]$ $S(B_{mp})$	pc_B	$wAct_B$	SC_1B	SC_2B	SC_tB	
			iter 1										
			iter 2										
			iter *u*										
h	pb_R	*v*	*R*	$pc[S(R_{m1})]$ $S(R_{m1})$	$pc[S(R_{m2})]$ $S(R_{m2})$	$pc[S(R_{m3})]$ $S(R_{m3})$	$pc[S(R_{mp})]$ $S(R_{mp})$	pc_R	$wAct_R$	SC_1R	SC_2R	SC_tR	
			iter 1										
			iter 2										
			iter *v*										

Figure 5: Matrix structure overview (Source: adapted from (Silva and Chituc, 2013)).

The performance coefficient (*pc*) is than obtained by the expression:

$$pc_A = \Big(\sum_{j=1}^{p} S[A_{m(j)}] / M_{(j)} \Big) / p \qquad (2)$$

p: number of metrics associated to the service;
M(j): domain maximum value that a assessment can achieve for each metric;
S(A_{mj}): natural average of assessments of each metric for service *A*;
pc_A: the average value of each metric for each service is addressed to the maximum possible value that can be achieved and therefore is obtained a proportional coefficient of performance.

3.2.2 Weighted Performance Averaging

Figure 5 shows four generic metrics, with four weights used (w1%, w2%, w3% and wp%). Weighted averaging activity (*wAct*) reflects that each client has different demands. The matrix results are constructed according to client preferences. *wAct* allows customer to configure the pool service to the levels customer wants to get.

$$wAct_A = \sum_{j=1}^{p} S[A_{m(j)}] \times w_{(j)} \qquad (3)$$

p: number of metrics in the pool
S(A_{mj}): metric average value of the service *A*;
$W_{(j)}$: the weight for each metric defined by client;
$wAct_A$: is the sum of the average values obtained for each metric, weighted by weights assigned directly by the client for service *A*.

3.2.3 Service Ranking Rules

This section refers to service ranking rules that allow, from the data reflecting the behaviour of services, to create a rating for each service by adding other variables that are important to assure that the global service offered is in accordance to customer requirements and preferences.

A) SC1: Oriented Coefficient of Performance

This rule uses the “penalty / benefit” parameter to determine the Oriented Performance Coefficient. The "penalty / benefit" column of the matrix (Figure 5) shows the value that characterizes the service behaviour of the last execution and superimposes an index, according to a table (that is configured by the owner) which aims to penalize – if the behaviour was below the expected performance coefficient of the service, or benefit – if it was above, adjusting the ranking of each service. The positive or negative deviation (resulting from its last behaviour) is classified by ranges with therefore a corresponding value which allows to benefit or to penalize the classification of the service ranking.

$$SC_{1A} = wAct_A + [wAct_A * (pc_A + (pc_A * pb_A) / 100)] \qquad (4)$$

SC_{1A}: adjusted value form performance coefficient and weighted averaging with the benefit or penalty parameter;
$wAct_A$: weighted performance averaging;
pc_A: performance coefficient;
pb_A: penalty or benefit value according to last run performance.

B) SC2: Service Utility

The number of times that a service is called to be integrated in choreographies is relevant and the scoring algorithm uses a factor that enhances the ranking according to the number of times the service is used. In addition to calculating the performance coefficient and the weighted average of each metric, evaluating the importance degree in terms of the utility of the service for the various choreographies instantiations is addressed by this rule. The utility factor table which supports this rule is parameterized according to the strategy of the global service provider. SC2 is given by the expression:

$$SC_{2A} = SC_{1A} + SC_{1A} * utilityFactor \quad (5)$$

SC_{1A}: adjusted value form performance coefficient and weighted averaging with the benefit of penalty parameter;
utilityFactor: by the number of times the service integrates choreographies a factor is returned.

C) SC3: Service Participation in High Performed Choreographies

Each time a service is called in a High Performed Choreography is important in order to add that contribution of the service performance for the high ranked choreography. SC3 follows the same principle of the previous rule and is given by the expression:

$$SC_{3A} = SC_{2A} + SC_{2A} * bestChorUtilityFactor \quad (6)$$

SC_{2A}: contains the value from previous calculations which gives the ranked value for the service;
bestChorUtilityFactor: by the number of times the service integrates high performed choreographies a factor is returned.

D) SC4: Ratio between Service Participation in High Performed Choreographies and Service Utility

This rule depends of a ratio between the number of times the service integrates a high performed choreography (with results in practice above the initially expected) and the total number of times a service was selected to common. Choreography. SC4 follows the same principle of the previous rules and is given by the expression:

$$SC_{4A} = SC_{3A} + SC_{3A} * ratioFactor \quad (7)$$

SC_{3A}: contains the value from previous calculations which gives the ranked value for the service ;
ratioFactor: is the ratio between the number of times the service was called by a high performed choreography and the total number of times a service was invoked in a choreography, a factor is returned.

4 RELATED WORK

Several approaches for service monitoring and assessment exist. (Garg et al., 2013) address the issue of monitoring services in the cloud through a framework that supports SMI attributes (Service index measurement). It Consists of a set of business-relevant KPIs that provide a standardized method for measuring and comparing business services.

(Whaiduzzaman et al., 2013) focus on a theoretical work about service selection for cloud computing in multicriteria decision analisys (MCDA) situations. They describe the MCDA types and characteristics and present a taxonomic categorization. Also summarize several of the advantages and disadvantages, and present several applications of these MCDA methods in the selection of cloud services.

A monitoring, predicting and adaptation approach for preventing KPI violations of business process instances is presented in (Wetzstein et al., 2012). A decision tree learning to construct classification models (which are then used to predict the KPI value of an instance while it is still running) is also discussed.

(Baresi et al., 2005) advance an approach towards monitoring WS-BPEL processes focusing on runtime validation, focusing on the identification of services delivering unexpected results, and not on monitoring process performance metrics.

Different from previous approaches, the monitoring and assessment approach described in this article focuses on service ranking rules related to service choreography. The scientific contribution of this research lies on the conceptual framework that supports the selection of the most suitable set of services available at a given time to answer customers' requirements and preferences, taking into consideration business process constraints, and the characteristics of the execution environment. Featuring a high level of learning acquired based on historical data, solutions with a high degree of predictability of the behavior of the overall service in terms of time, cost and quality could be developed.

5 CONCLUSIONS AND FUTURE WORK

Nowadays, a complex service is developed and managed by a consortium of companies that jointly contribute to its development, sharing costs and risks. Different parts of the process are carried out at different organizations. Each company has its own responsibility of the part of the choreography of processes in which it participates. A choreography approach is than used as coordination across many domains of control/visibility is required. In this, assessment and monitoring of cross-organizational business processes are fundamental. However, an extensive literature review has revealed that the combination of SOC, complex adaptive systems and adaptive control systems has not been addressed in a conceptual and systematic way. Closing this research gap is a focal point in this paper.

Based on complex adaptive systems theory, an adaptive framework of services has been constructed, including hierarchical levels of control to enable predictability considering cost, time and quality characteristics. The closed loops for controlling the behaviour of the framework are supported by the measurements of metrics at different levels allowing establishing individual pools of rankings of services. A choreography is then assembled with the most adequate available services according to customers' requirements and preferences.

The proposed framework and the approach for services selection allow to dynamically identify the appropriate set of services to target customer requirements and preferences, making this offer available to the customer before (s)he decides to acquire the whole service. This approach will benefit the relationship between customer and the provider in the sense that will assign to the relationship a favorable degree of reliability, facilitating new interactions. Thus, this research work contributes to the state-of-the-art by advancing a conceptual adaptive framework of services that will ensure a high degree of predictability for the services' chorography.

Future work will focus on the overall assessment of the choreography and validation of the proposed framework. The scalability and adaptability of the framework will also be targeted in future work.

REFERENCES

Ackoff, L., 1990. *Redesigning the Future - Strategy*, Systems Practice, Vol. 3, No. 6, 1990, pp. 521-524.

Baresi L., Guinea, S.: Towards Dynamic Monitoring of WS-BPEL processes. In: Proc. 3^{rd} Int. Conf. on Service-Oriented Computing (ICSOC'05), Springer, pp. 269-282 (2005).

Di Nitto, E. et al., 2009, *S-Cube: Addressing Multidisciplinary Research Challenges for the Internet of Services, Towards the Future Interne*t, G. Tselentis et al. (Eds.), IOS Press.

Garg, S. et al., 2013. *A framework for ranking of cloud computing services*, Future Generation Computer Systems, vol. 29, no. 4, pp. 1012–1023.

Hellerstein, J.L., Diao, Y., Parekh, S., Tilbury, D. M., 2004. *Feedback Control of Computing Systems*, IEEE Press /Wiley Interscience.

Landau, I. D., Lozano, R., M'Saad, M., Karimi, A., 2011. *Adaptive Control - Algorithms, Analysis and Applications*, Communications and Control Engineering, Springer.

Kaplan, R. S., Norton, D. P., 1992. *The Balanced Scorecard: Measures that Drive Performance*, Harvard Business Review, (January-February): 71-79.

Kaplan, R. S., Norton, D.P., 1996. *Linking the Balanced Scorecard to Strategy*, California Management Review, Vol. 39, Nº 1, FALL 1996.

Kaplan, R. S., Norton, D. P., 2008. *Mastering the Management System*, Harvard Business Review, January 2008, pp. 62-77.

Mintzberg, H., et al., 1996. *The Strategy Process:Concepts, Contexts, Cases*. Prentice Hall.

Papazoglou, M. P. et al., 2006. *Service-oriented computing: A research roadmap*, In Service Oriented Computing (SOC), Dagstuhl Seminar Proceedings, no. 05462. IBFI, Schloss Dagstuhl, Germany.

Silva, F., Chituc, C.-M., 2013. *Towards the Definition of a Framework Supporting High Level Reliability of Services.* International Conference of Service Oriented Computing 2012, Shangai, China, Springer Lecture Notes in Computer Science Volume 7759, 2013, pp. 143-154.

Whaiduzzaman, M. et al., 2013, *IT. Cloud Service Selection using Multi-Criteria Decision Analysis*. The Scientific World Journal.

Wetzstein, B. Et al., 2012. *Preventing KPI violations in business processes based on decision tree learning and proactive runtime adaptation*. Journal Systems Integration 3(1), pp. 3-18.

Wetzstein, B., et al., 2010, *Cross-organizational process monitoring based on service choreographies*, In proceedings of the 2010 ACM Symposium on Applied Computing, March 22-26, Switzerland.