



Dynamism in Inter-Organizational Service Orchestration - An Analysis of the State of the Art

Paul Grefen & Stefanie Rinderle-Ma

Beta Working Paper series 497

BETA publicatie	WP 497 (working paper)
ISBN	
ISSN	
NUR	
Eindhoven	Maart 2016

Dynamism in Inter-Organizational Service Orchestration - An Analysis of the State of the Art

Paul Grefen¹ & Stefanie Rinderle-Ma²

¹*School of Industrial Engineering, Eindhoven University of Technology*

²*Computer Science Department, University of Vienna*

Abstract. Modern-day business is increasingly service-oriented. In this context, delivered services are increasingly complex in functionality, relying on a composition of underlying basic services. In many business domains, the basic services underlying a single composed service are provided by a set of autonomous business organizations, each delivering part of the functionality of the composed service. This requires inter-organizational service orchestration. To follow market dynamics, orchestrated complex services have to be dynamic as well: their specification changes to accommodate changing customer requirements, changing economic circumstances, and changing offerings of collaboration partners. To understand these dynamics in service orchestration, a well-structured framework is required. In this paper, we provide such a framework, based on a combination of developments in inter-organizational business process management and service management. We use the framework to assess the state of the art in research by analyzing a set of general research approaches as well as a collection of concrete research projects in the domain of inter-organizational business process management and service orchestration.

1 Introduction

In modern-day business, we see an increasing level of service-orientation in many business domains. This can lead even up to service-dominant business [Var04, Lüf12, Kry13, Lüf14], in which service-orientation is the main business paradigm. One of the main characteristics of service-oriented business is the fact that it relies on the composition of individual services into complex services. The individual services encapsulate the core capabilities of service providers, but the complex services provide the actual value (solution) to the customer. Typically, the individual services in a composed service are provided by multiple autonomous service providers. As an example, an integrated travel service may be provided by the composition of individual services from a travel service integrator, several providers of transport, several providers of travel accommodation, and a travel insurance company [Gr13a, Gr15a]. Complexity in inter-organizational orchestration may be great. As an example, we can take supply chain management as an example domain of orchestration. A survey published by the Aberdeen Group in 2012 shows that growing supply chain complexity is identified as the top business pressure [Hea12].

In most business domains, current market dynamics demand that service compositions are agile: their definition and enactment must be able to follow changes in customer requirements, changes in provider offers, changes in economic circumstances, and changes in legal and technological settings. As an example for the integrated travel service, the level of service expected by travelers may change (typically increase) quickly over time. Also, the definition of an integrated travel service may change as new transport providers become available. The execution of an integrated travel service may change when an existing service provider goes bankrupt. In other words: service compositions must be dynamic in their specification (at design time) and possibly in their execution (at run time). Adaptiveness and dynamism is also heavily studied in research¹. Dynamism adds to the complexity of service orchestration observed above. To relate this to the

¹ For example, collective adaptive systems were the topic of the plenary panel discussion at ICSOC 2014.

prior example: it has been observed that the largest cause of business complexity is greater expectation on the part of the customer [Eco11].

To support dynamic compositions of business services provided by sets of autonomous providers, we require dynamism in inter-organizational service orchestration. We focus on orchestration here because of the business context: one party always has the end responsibility towards the customer for a composed business service, making an orchestration paradigm more natural than a choreography paradigm. Dynamism in inter-organizational service orchestration relies on being able to handle changes in the control flow that defines the sequencing of the individual services. As such, this is very similar to dynamism in inter-organizational business process management - a domain that is already slightly older than that of service orchestration.

Even though a range of approaches exist that deal with dynamism in inter-organizational service orchestration or business process management, the topic is still poorly understood. Many approaches originate from a specific business context or depart from a specific choice of technologies and hence have an ad-hoc contribution to the field. To arrive at a consolidation of research efforts, an integration of existing approaches and an agenda for problems to be solved, a framework is required that places all individual developments into a well-structured perspective. Describing such a framework is the goal of this paper. In doing so, we integrate developments from business process management and service management.

The structure of this paper is as follows. In Section 2, we start with laying the foundation for the development of the framework in terms of basic concepts. Next, we describe the framework in Section 3, arriving at a three-dimensional space for classifying dynamism. In Section 4, we use this classification space to analyze the state of the art from a general research approach point of view in the domains of inter-organizational business process management and service management. In Section 5, we analyze a set of specific research projects in these domains. We conclude the paper and point towards future work in Section 6.

2 The basic concepts

In this section, we present the basic concepts that we use for the analysis of the domain of dynamism in inter-organizational service orchestration.

Orchestration dynamism requires (1) the ability to apply changes to an existing orchestration specification at the type and/or instance level during the life cycle of a service composition and (2) the ability to switch to another orchestration specification at the type and/or instance level during the life cycle of a service composition. In this paper, we are interested in the structural application of changes to a specification, but not in the correctness of these changes. Design-time correctness of changes is dealt with elsewhere, for example with respect to the aspects of consistency [Aal00] and transactional correctness [Rin14].

We refer to a changed version of an orchestration specification as a *variant* of that specification. A variant can exist before a switch is triggered or can be constructed on-the-fly after a switch is triggered. We call the first type a *pre-existing variant*, the second type an *on-demand variant*.

Switching to a process variant can be triggered because of two classes of reasons. The first class of reasons is *internal* to the service orchestration, i.e., it is part of programmed exception management. The second class of reasons is *external* to the service orchestration, i.e., they are reactions to non-covered environmental events. In this paper, we are mainly interested in externally triggered dynamism because of dynamic business orientation.

Orchestration dynamism can appear at the global and the local level. Orchestration dynamism at the *global level* includes changes that are visible at the inter-organizational level, i.e., changes that affect the way organizations collaborate to realize a complex service. Orchestration dynamism at the *local level* is restricted to changes that are within the scope of a single organization, i.e., do not

affect the interaction between organizations. In this paper, we are mainly interested in orchestration dynamism at the global level.

3 A classification framework

In this section, we describe a classification framework for analyzing syntactical and semantic aspects of change of service orchestrations, i.e., on a tool to describe what kind structural changes exist and what they mean. The framework does not cover the analysis of pragmatic aspects, i.e., a classification of reasons why changes are required. The reason for this choice is to keep the number of dimension in the framework small and the use of the framework practical. We revisit this issue when we discuss future work at the end of this paper.

In the construction of the framework, we have used the interrogative-based approach (as used also for example in the design of the Zachman framework of enterprise architecture [Zac02, Wi16b]) for the identification of the main characteristics of dynamic change of orchestrations. We have identified the following main interrogatives:

1. *When* is a change applied to an inter-organizational service orchestration, i.e., what is the change moment?
2. *What* is changed by the application of a change to an orchestration, i.e., which objects are affected by a change?
3. *How much* of the structure an orchestration specification is changed by the application of a change, i.e., how severely is a specification altered?

The *when* interrogative leads to what we call the *timing of change* dimension. The values in this dimension are based on the life cycle of a service orchestration: *definition time*, *deployment time*, *instantiation time*, and *execution time*. Note that absolute time values to measure frequency of change (like once per week) are not very meaningful here to analyze the state of the art, as they are heavily dependent on the application domain at hand: in ship building processes, a week may be a valid ‘clock tick’ to measure business process execution progress, whereas in stock trading, a millisecond may be the proper ‘clock tick’.

The *what* interrogative leads to what we call the *scope of change* dimension. The values in this dimension are based on the amount of orchestration instances that is affected by a change: *all instances* of an orchestration type, *batch of instances* using an explicit instance set qualification, and *single instance* using an explicit instance identification. Comparable to the absolute time values for the *timing of change* dimension, absolute numbers of instances are not meaningful in the *scope of change* dimension, as they are heavily dependent too on the application domain at hand: in ship building processes, ten process instances may be many, whereas in stock trading, a hundred thousand may be a quite regular number.

The *how much* interrogative leads to what we call the *intensity of change* dimension. The set of values in this dimension is less obvious than for the other two dimensions: there is no generally accepted set of levels of the amount of change to an orchestration specification. Therefore, we have developed a set of values in a bottom-up way by inspecting projects and approaches. This has led to the following values: *parameterization of service* (without making a change to the control flow), *substitution of service* (making a local change to the control flow), *interlinking of services* (making a global change to the control flow, assuming service compatibility), *assembly of services* (making a global change to the control flow, using mechanisms to deal with service incompatibility), and *construction of orchestration* (replacing the entire global control flow). The five values in this dimension are illustrated in Figure 1. In this figure, the change to an orchestration is indicated by the elements with dashed lines.

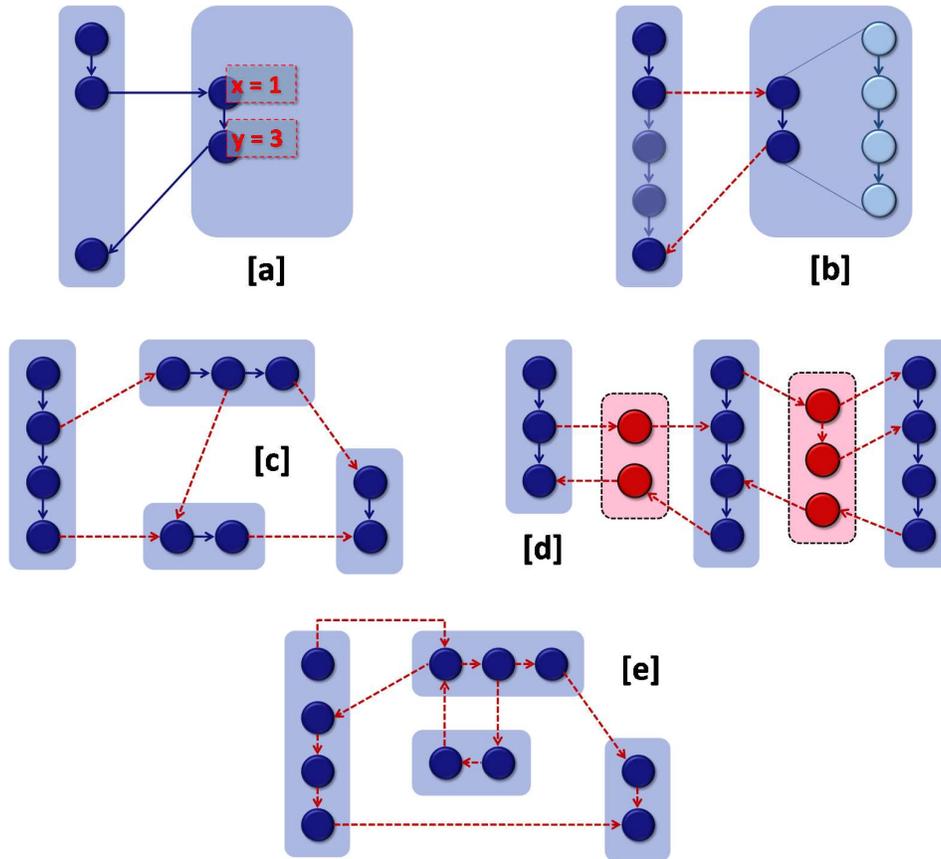


Figure 1: illustration of values of intensity of change dimension:
 [a] parameterization [b] substitution [c] interlinking [d] assembly [e] construction

An overview of the classification framework consisting of the three dimensions discussed above is shown in Figure 2.

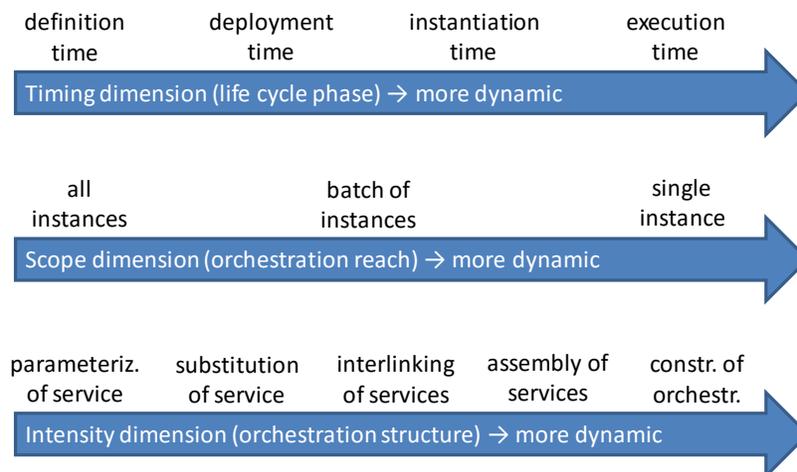


Figure 2: overview of dimensions in framework

All three dimensions in our framework have an ordinal scale [Ste46, Wi16a]. The ordinal type of scale allows for rank order by which elements (approaches to orchestration dynamism) can be sorted, but do not allow for measuring relative degree of difference between elements. This means

for example that we can state that an element that has an *instantiation time* timing is more dynamic in the timing dimension than an element that has a *deployment time* timing, but that we cannot state that it is twice as dynamic. When we discuss the central tendency of an ordinal scale, the *mode* is the most common item, the *median* is the middle-ranked item. So, we can state for example that the mode of the timing dimension in classic approaches to service orchestration is *definition time*.

Note that the values we have chosen for the scales of the three dimensions can be discussed: different values can be chosen and the chosen scales can be refined by splitting values. For example, in the *timing of change* dimension, we can split the value *execution time* into the values *planned at execution time* and *ad-hoc at execution time*. Here, the second value indicates more dynamism than the first: with the *planned* variant, changes are applied at a predefined point in the execution of an orchestration (e.g., the second half of an orchestration is constructed after the execution of the first half has been completed); with the *ad-hoc* variant, changes can be applied arbitrarily during execution (e.g., an orchestration can be constructed completely in a free incremental way). Changing the values in the scales of the dimensions does not change the overall approach that we pursue in the paper, however.

To visualize combinations of values in the three dimensions, we plot these combinations in a three-dimensional space for dynamism created by the three dimensions and their values. This space is shown in Figure 3. As indicated in the figure, the front-left-bottom corner of the space represents approaches with a low level of dynamism, the back-top-right corner represents approaches with a high level of dynamism.

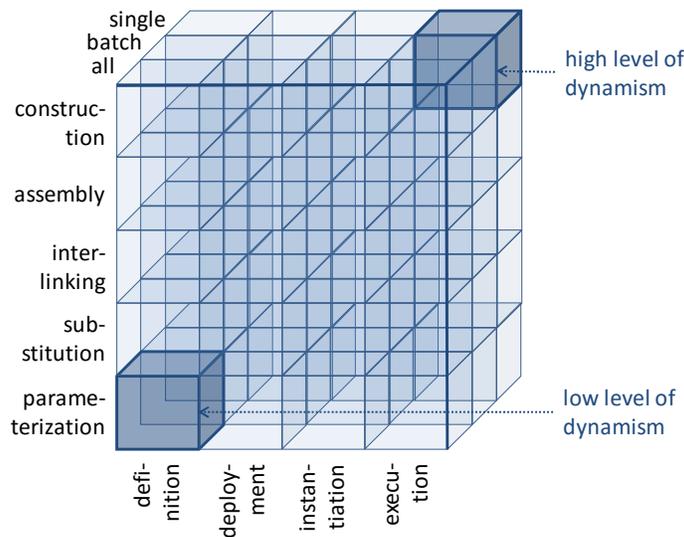


Figure 3: three-dimensional space for dynamism

In the next two sections, we use the three-dimensional space to analyze and classify the state of the art in research of inter-organizational service orchestration (and inter-organizational business process management as its predecessor). In doing so, we plot approaches (Section 4) and projects (Section 5) in the space as shown in Figure 3.

4 Analysis of general research approaches

In this section, we analyze general research approaches in the field of dynamism in service orchestration. We do this using the classification framework described in the previous section. In the first subsection, we analyze general research approaches (or research subfields) from the

business process management (BPM) perspective. In the second subsection, we do the same from the service management (SM) perspective. We end this section with a few concluding observations.

4.1 Positioning of general research approaches from the BPM perspective

As discussed earlier in this paper, research into business service orchestration has its roots in the business process management (BPM) field. Therefore, we start the analysis of research approaches into orchestration dynamism in the BPM field. We can distinguish four important general approaches that we can position in our classification space:

- business process reengineering;
- evolution by escalation;
- single-use processes;
- ad-hoc processes.

We discuss each of the four approaches in the following subsections. The four general research approaches to change in business processes identified above are (obviously) not all existing approaches – other approaches can be classified in our framework as well.

4.1.1 Business process reengineering

The first general approach is that of ‘classical’ business process reengineering (BPR). In BPR, process change is typically applied at process *definition time* (in the *timing of change* dimension of our framework). A BPR project consists of several distinct phases or stages [Ket97], one of which is changing a process definition. In BPR, typically *all instances* of a process model (in the *scope of change* dimension) are redefined towards the future. The BPR approach does not constrain the value in *intensity of change* dimension: in a reengineering cycle, a process definition may be re-parameterized or it may be completely re-constructed – and everything in between. This means that the BPR approach can be illustrated by a vertical column of cells in our classification space, as illustrated in Figure 4. To make reading the figure easier, we have only included dimensional value labels that correspond with the chosen approach.

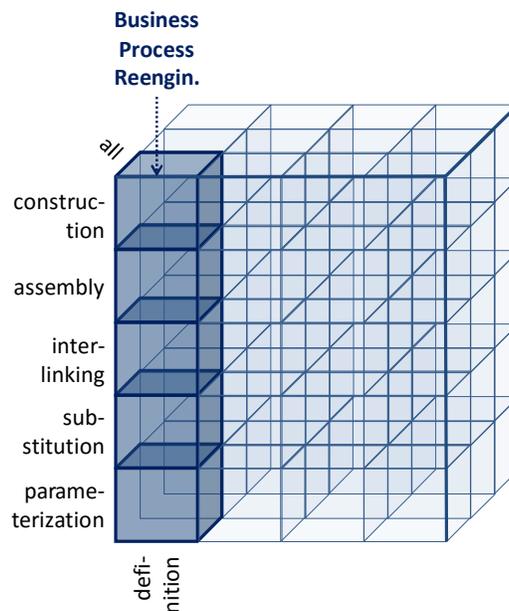


Figure 4: business process reengineering in dynamism space

4.1.2 Single-use processes

The second general approach is that of single-use processes (SUP), i.e., processes the definition of which is only used for a single instance. Single-use processes are deployed in two kinds of situations: unique, long-lasting (and typically expensive) processes and fully automatically composed processes (e.g. by fully automatically composing processes in service outsourcing [Gre00]). In the application of SUP, process change is applied at process *definition time* (in the *timing of change* dimension) for a *single instance* (in the *scope of change* dimension). In the *intensity of change* dimension, we can find any value depending on the change approach: for example, a SUP definition can be created by *parameterizing* an existing process template, by *interlinking* existing process fragments (where each fragment may correspond with a partner in a business network), or by *constructing* an entirely new process definition from scratch. The SUP approach is illustrated by a vertical column of cells in Figure 5 (again omitting superfluous dimensional value labels).

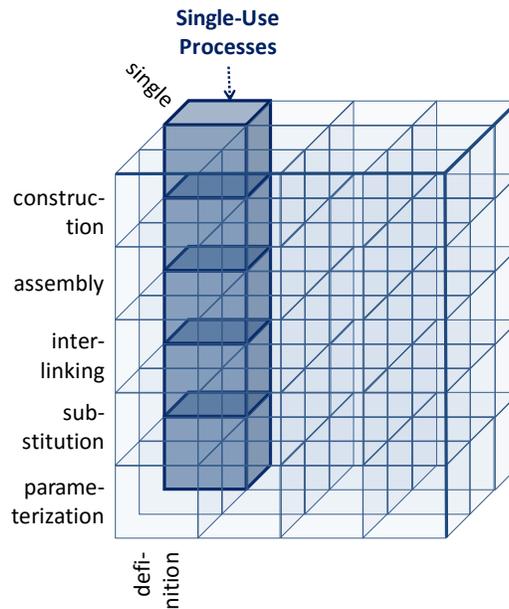


Figure 5: single-use processes in dynamism space

4.1.3 Ad-hoc processes

The third general approach is that of ad-hoc processes (AHP). Ad-hoc processes are processes the definition of which (gradually) emerges during the start-up and execution of the process. Clearly, the AHP approach focuses on the *single instance* value in the *scope of change* dimension. As changes are not restricted by any rules, it is classified at the *construction* value in the *intensity of change* dimension. In the *timing of change* dimension, we can find any value. When the value in this dimension is *definition time*, the approach coincides with the single-use process (SUP) approach as discussed above. The AHP approach is illustrated by a horizontal row of cells in Figure 6. Note that the AHP approach overlaps one cell with the SUP approach discussed before: though working from different principles, similar abstract mechanisms are put in place in this cell.

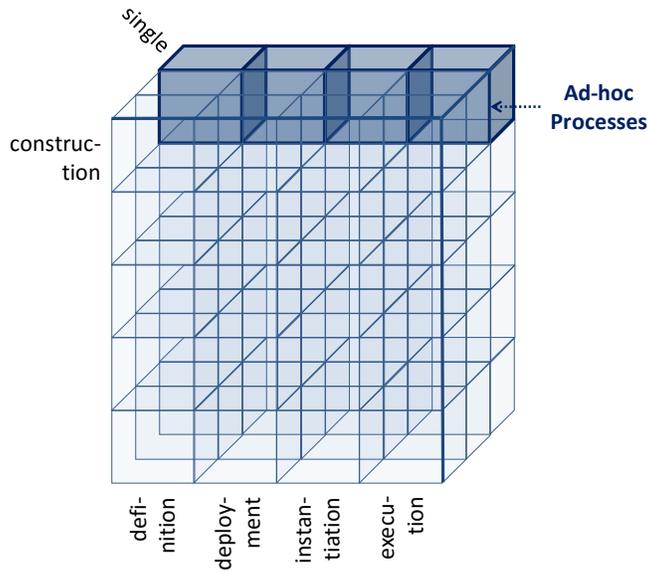


Figure 6: ad-hoc processes in dynamism space

4.1.4 Evolution by escalation

The last general approach we discuss here is that of evolution by escalation (EBE). In this approach, the specification of a running process instance is changed as a consequence of practical insights from the execution, and this this specification is next escalated to the type level of the process (to be used for other instances as well). In the *timing of change* dimension, the EBE approach has the *execution time* value. It is geared towards changing process type specifications, so in the *scope of change* dimension, it has the *all instances* value. In the *intensity of change* dimension, we can find any value. The position of the EBE approach in our dynamism framework is illustrated in Figure 7.

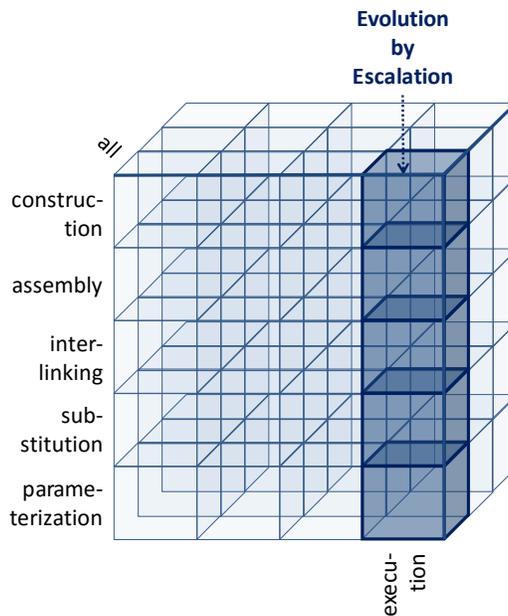


Figure 7: evolution by escalation in the dynamism space

4.2 Positioning of general research approaches from the SM perspective

After positioning business process management approaches in our framework in the previous subsection, we now turn to positioning service management approaches. We first discuss the static and dynamic service outsourcing approaches. Then we turn our attention to the static and dynamic service composition approaches.

4.2.1 Static and dynamic service outsourcing

From a service decomposition point of view, service outsourcing is an important concept in service management. In service outsourcing, we can distinguish a *static* and a *dynamic* variant.

In Figure 8, we position *static service outsourcing* in our classification space. This approach uses *substitution* to allocate part of a service composition to a service provider. This provider can be determined at *definition time* or *deployment time*. After an outsourced service has been deployed, it can be newly *parameterized* at *instantiation time* or *execution time* to change its behavior for a *batch* of service instances. This happens without changing the service provider and hence without changing the outsourcing topology.

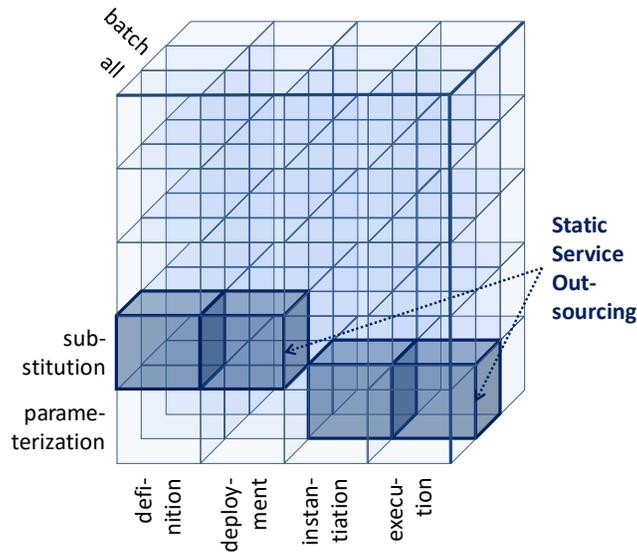


Figure 8: static service outsourcing in dynamism space

In Figure 9, we position *dynamic service outsourcing* in our classification space. This approach again uses *substitution* (like static service outsourcing). The provider is, however, determined for each single service instance at *instantiation time* or *execution time* (in the life cycle of the outsourcing service, i.e., from the point of view of the service consumer).

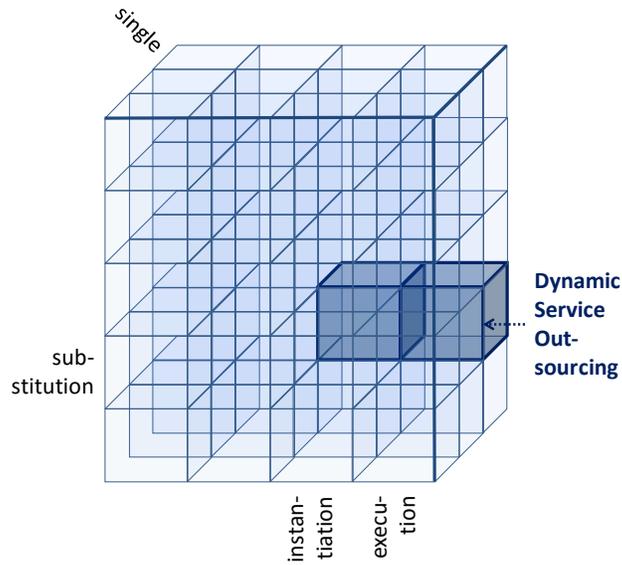


Figure 9: dynamic service outsourcing in dynamism space

In between the approaches of static service outsourcing and dynamic service outsourcing, we have the approach of semi-dynamic service outsourcing. In this approach a service outsourcing topology is determined per *batch* of service instances.

4.2.2 Static and dynamic service composition

Where service outsourcing uses a decomposition approach, *service composition* uses a service integration approach. Service composition exists both in a static and a dynamic variant.

In *static service composition*, a number of services are combined using *interlinking* or *assembly*. With interlinking, the services are directly compatible. With assembly, the services have to be made compatible by the use of service adapters. The combination takes place at *definition time* or at *deployment time*. The combination is made for *all* service instances. We show the position of the approach in Figure 10.

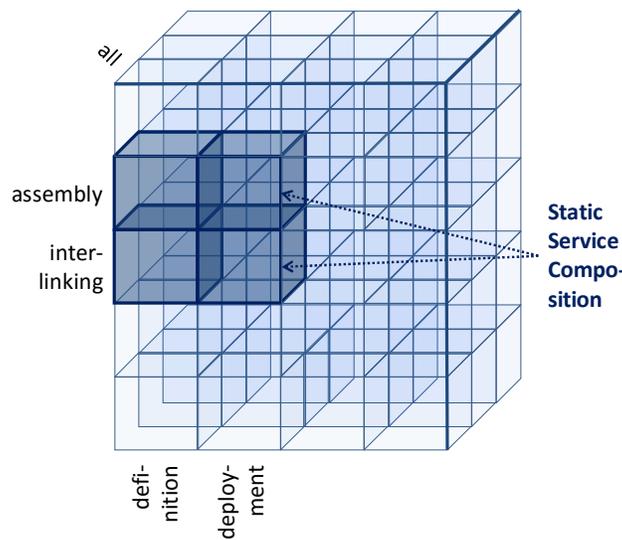


Figure 10: static service composition in dynamism space

In *dynamic service composition*, we again use *interlinking* or *assembly*, but now at *instantiation* or *execution time*. A combination is made for a *single* service instance. Note that the *assembly* case typically means that service adapters have to be constructed on-the-fly, e.g., through the use of automated adapter generation [Bro06]. Note also that the *execution time* case means that combination of services is performed incrementally during execution (otherwise, *execution time* would reduce to *instantiation time*) - an approach that is rarely used in practice to the best of our knowledge, but that is possible in principle. We show the position of dynamic service composition in the dynamism space in Figure 11.

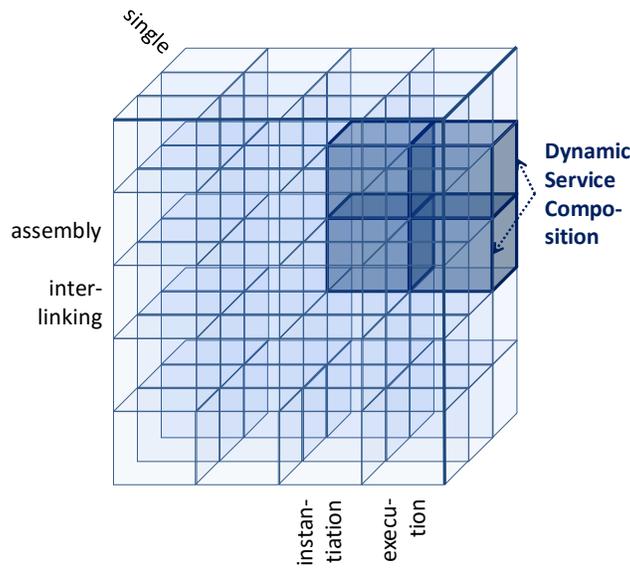


Figure 11: dynamic service composition in dynamism space

4.3 Conclusion with respect to positioning general approaches

When we look at the positioning of the general BPM approaches in our design space, it may be considered striking that they all are at the edges of the cube that represents the dynamism space. This is caused, however, by the fact that we have modeled ‘pure conceptual’ approaches in flexible BPM here. In practice, amalgamations of the ‘pure’ approaches can be used, which may be positioned away from the edges. For the general SM approaches, the positioning is less ‘extreme’ – they are not positioned along the edges of the cube. This is caused by the fact that these approaches have a more operational character, which leads to more ‘balanced’ values in the dimensions.

When we compare the general approaches from the BPM and the SM domains as analyzed in this section, we see that we have obtained rather different patterns in our classification spaces for BPM and SM. This may be caused by the fact that BPM is rooted in control flow that links underlying steps (i.e., top-down composition of functionality), whereas SM is rooted in individual services that are linked by overlying orchestrations or choreographies (i.e., bottom-up composition of functionality). This can also imply that where we combine BPM and SM to achieve business dynamism (agility), we may expect a ‘clash or paradigms’.

5 Positioning of specific research efforts

In this section, we turn our attention to the analysis of specific research efforts and projects in the field of dynamism in service orchestration. We use the classification space of Section 3 like we did

in the previous section for general research approaches. We also cover both the business process management and service management perspectives like we did in the previous section.

5.1 Positioning of specific research efforts from the BPM perspective

In this subsection, we put the focus on research efforts that depart from a process-centric view in the design of support for flexible inter-organizational collaboration. We analyze the following projects: WISE, CrossFlow, CrossWork, ADVENTURE, GET Service and C³Pro.

5.1.1 WISE

The WISE project (Workflow based Internet Services) is a project in the field of inter-organizational business process management at ETH Zürich, started in 1998. It aims at providing a software platform for process based business-to-business electronic commerce [Alo99, Laz00, Laz01]. The project focuses on support for networks of small and medium enterprises.

The software platform used in WISE is based on the OPERA kernel [Alo97]. WISE relies on a central workflow engine to control inter-organizational processes, called virtual business processes. A virtual business process in the WISE approach consists of a number of black-box services linked in a workflow process [Alo99]. A service is offered by an involved organization and can be a business process controlled by a workflow management system local to that organization – but this is completely orthogonal to the virtual business process. Specification of virtual business processes in WISE is performed using the Structware/IvyFrame tool [Lie98]. This tool and its specification technique are used to construct both the conceptual structure of inter-organizational processes and the specifications of services exchanged between organizations in a virtual enterprise.

Defined virtual business processes in WISE are static, hence in the *timing* dimension of our framework, WISE is classified as *definition time*. The WISE approach links black-box processes, so WISE classifies as *interlinking* in the *intensity* dimension. A virtual business process definition holds for all its instances, so in the *scope* dimension, we have the *all* value. The corresponding position of WISE in our framework is shown in Figure 12.

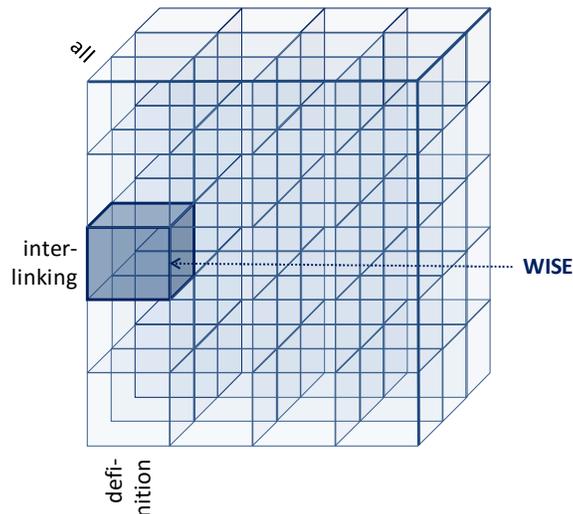


Figure 12: position of WISE project in dynamism space

5.1.2 CrossFlow

CrossFlow [Gre00, Hoff01] is a European research project that explores the use of dynamic service outsourcing from an inter-organizational process management perspective. The project

started in 1998. In the CrossFlow approach, an organization selects parts of its business processes that it considers non-core with respect to its competences and outsources these sub-processes to service providers for whom these sub-processes are covered by their core competences. The CrossFlow approach has been prototyped in the service industry, more specifically business-to-consumer telecom logistics and business-to-business insurance claim handling.

CrossFlow is strongly rooted in BPM, using a commercial BPM system for the execution of local business processes. Dynamic outsourcing is based on a CORBA-based broker and fully automated electronic contracting [Koe00, Hoff01]. Dynamic linking of processes is supported by an ad-hoc software solution based on business process abstraction.

CrossFlow follows two approach variations where it comes to dynamism in business process management. In the most dynamic approach, each outsourcing is performed (and contracted) completely dynamic during the execution of the outsourcing process. Hence, in the *timing* dimension of our framework, CrossFlow is classified as *execution time* and in the *scope* dimension, CrossFlow is classified as *single instance*. In the outsourcing paradigm, placeholders for sub-processes are replaced by actual outsourced processes, so CrossFlow is classified as *substitution* in the *intensity* dimension. In Figure 13, this variation is shown as the right-hand side highlighted cell.

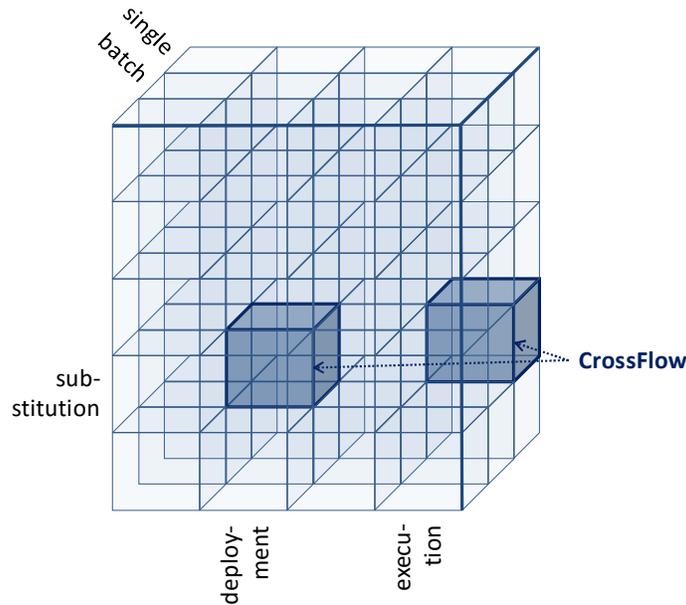


Figure 13: position of CrossFlow project in dynamism space

The second variation to dynamism used in the CrossFlow project uses umbrella contracts: a party to execute an outsourced sub-process is selected for a batch of process instances (instead of for a single instance as above). This means that for the *scope* dimension, we now have the *batch of instances* value. In the *timing* dimension, things move to the *deployment time* value: a contract is established after the process is defined and before instances are instantiated. In the *intensity dimension*, there is no difference between the two variations. This leads to the left-hand side cell highlighted in Figure 13.

5.1.3 CrossWork

CrossWork [Gre09, Meh10] is a European research project that aims at developing semi-automated support for process-based instant virtual enterprises in the high-tech manufacturing industry. The project started in 2004. The main idea is that manufacturing of series of complex

products requires the dedicated collaboration between a set of organizations that form a temporary virtual enterprise for that purpose in an agile way. The overall business process in this instant virtual enterprise is created by connecting the local business processes of the collaborating organizations with global control flow links. The CrossWork approach has been prototyped in the automotive industry.

In the CrossWork approach, global processes are completely designed and determined at definition time, which determines the value in the *timing* dimension in our analysis framework. As a virtual enterprise (and hence a global business process) is set up for a series of products, in the *scope* dimension, CrossWork has the *batch of instances* value. As global processes are built from pre-existing local processes, the value in the *intensity* dimension is *interlinking*. The resulting position of CrossWork in the classification space is shown in Figure 14 as ‘CrossWork’.

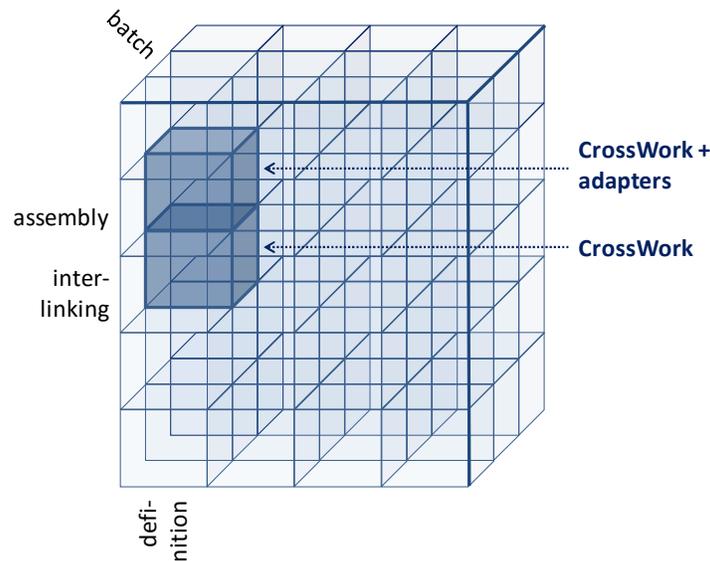


Figure 14: position of CrossWork project in dynamism space

In follow-up research in the ASCI Ph.D. project, the CrossFlow approach is extended with protocol adapters that enable handling of process incompatibilities between collaborating partners [Seg14]. This extension allows the CrossFlow approach to also cover the *assembly* value in the *intensity* dimension. This is indicated by ‘CrossWork + adapters’ in Figure 14. This is an example of an extension towards additional dynamism.

5.1.4 ADVENTURE

ADVENTURE [Sch12] is a European research project aiming at providing dynamism in manufacturing processes of highly customized products, which started in 2011. Dynamism is provided by incrementally filling in process templates and by reacting to changes that occur during process execution.

In the ADVENTURE approach, process definitions are developed during *execution* time (including ad-hoc changes to a definition), starting from a basis at process *instantiation* time. The approach is classified accordingly in the *timing* dimension of our classification space. Process definition can be on the basis of *assembly* of existing sub-processes or by *construction* of new parts, which determines the values in the *intensity* dimension. In the ADVENTURE approach, a product and the associated production process is developed for each individual customer order, which sets the value *single instance* in the *scope* dimension. The position of the project in the dynamism space is shown in Figure 15.

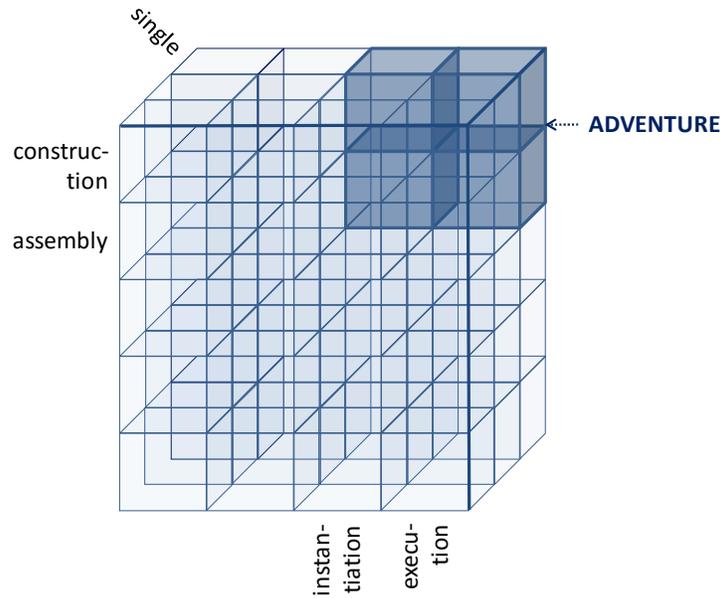


Figure 15: position of ADVENTURE project in dynamism space

5.1.5 GET Service

GET Service [Gr13b, Ba15a] is a European research project aimed at the development of an approach and infrastructure for the use of real-time data to increase the quality of multi-modal logistics processes with respect to their execution time, execution cost and carbon footprint. The project started in 2012. It does so by allowing logistics processes to be planned (called *static planning*) and created from existing transport sub-processes for individual business orders, taking into account real-time event information [Ba15b]. It also allows changing business processes during execution when exceptional situations call for this. Here, real-time information is used to re-plan (called *dynamic planning*) and re-specify an overall business process from existing sub-processes. A typical example is replacing the transport modality (e.g. train) for a transport leg by another transport modality (e.g. truck) - a principle called *synchro-modality*. Note that the name of the project suggests that we should have put it in the service management perspective - the approach in the project is, however, strongly process-oriented.

When we position GET Service in our classification space, we arrive at the following. The approach generates an overall business process for each customer order, so in the *scope* dimension the value is *single instance*. An overall business process is created from sub-processes (called *snippets* in the GET Service project), hence the value in the *intensity* dimension is *interlinking*. In the *timing* dimension, we find the value *definition time* for static planning and *execution time* for dynamic planning. This results in the positioning as shown in Figure 16.

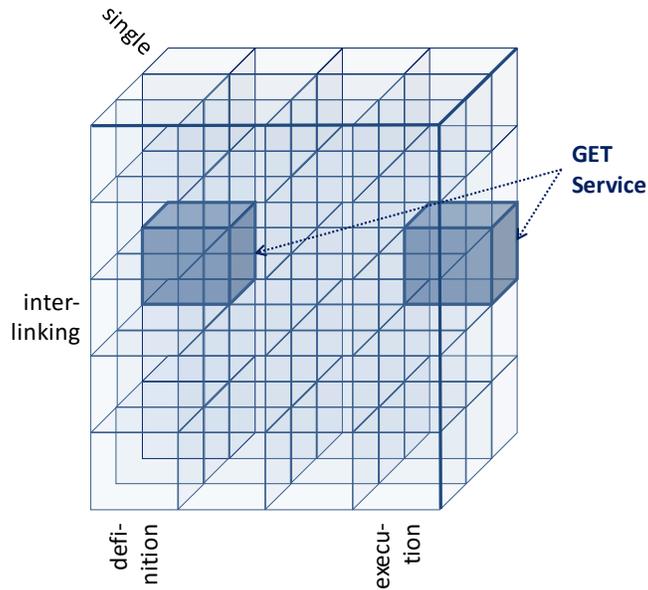


Figure 16: position of GET Service project in dynamism space

5.1.6 C³Pro

C³Pro is a project aiming at the support of change definition and propagation in collaborative process scenarios, independent of the particular process specification language. The project started in 2011. If a change is defined and applied at one partner's side, C³Pro provides a methodology including correctness / soundness criteria to propagate and possibly negotiate the change impacts to the other partners of the collaboration [Fdh12]. Further on, [Fdh12] presents three patterns for defining changes on collaborative processes that can be applied at build time and run time aiming predominantly at supporting adjustable processes.

The project supports dynamism at the level of individual process instances, which places it at the *single instance* value in the *scope* dimension. The approach focusses both on build time and run time issues, which places it both at the *definition time* and *execution time* values of the *timing* dimension. The approach allows for *construction* of processes in the *intensity* dimension. The resulting classification of the project is shown in Figure 17.

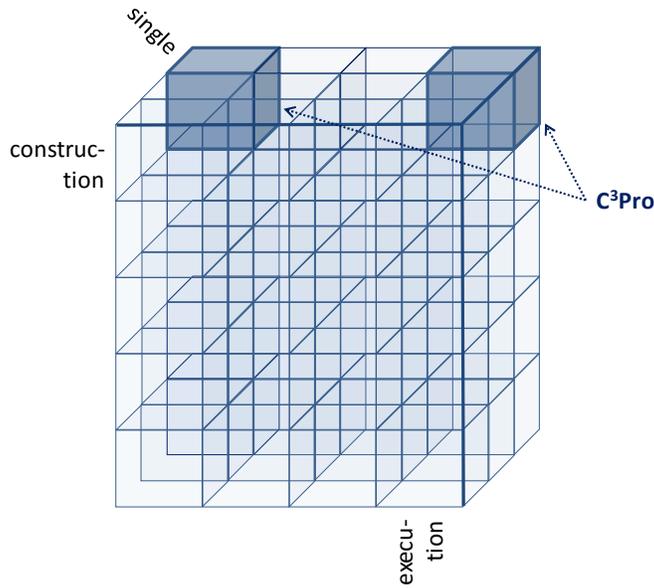


Figure 17: position of C³Pro project in dynamism space

5.1.7 ComVantage

ComVantage is an EU project aimed at leveraging heterogeneous data interoperability technology enablers for enterprise mobile collaboration [Sal11]. The project has a data-oriented focus, but places business process management in the center of its operational approach. The project started in 2011.

ComVantage is aimed at the business process type level, so we place it at the *all* value in the *scope* dimension. Processes are specified at *definition time*, which determines the value in the *timing* dimension. The approach allows *interlinking* of local capabilities in business networks, which determines the value in the *intensity* dimension. The resulting classification of the ComVantage project is shown in Figure 18.

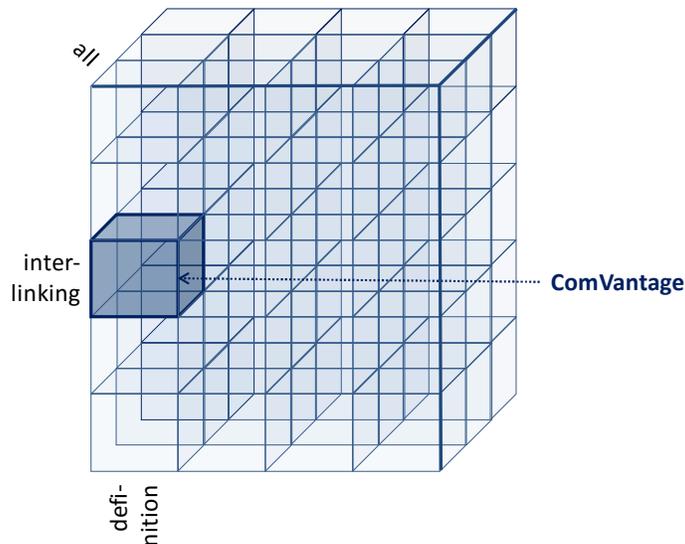


Figure 18: position of ComVantage project in dynamism space

5.2 Positioning of specific research efforts from the SM perspective

In this section, we analyze research projects that depart from the service management perspective. Like we have done in the previous section, we use the analysis to position projects in our three-dimensional classification space. We analyze the following projects: XTC, SYNERGY and CoProFind.

5.2.1 XTC

XTC [Wan06, Wan08] is an inter-university academic research project in the Netherlands, which started in 2003. Aim of the project is the development of a framework for the combination of service composition and transactional business semantics [Pap06], such as to support service-oriented e-business applications with proper operational semantics.

XTC focuses on definition-time issues, which determines its position in the *timing* dimension of our classification space. Service-oriented applications are built by service composition, which places the project at the *interlinking* value in the *intensity* dimension. No explicit support is envisioned in the project for dynamism of the level of individual service instances or batches of instances, so we place the project at the *all instances* value in the *scope* dimension. The classification of the XTC project in the dynamism space is shown in Figure 19.

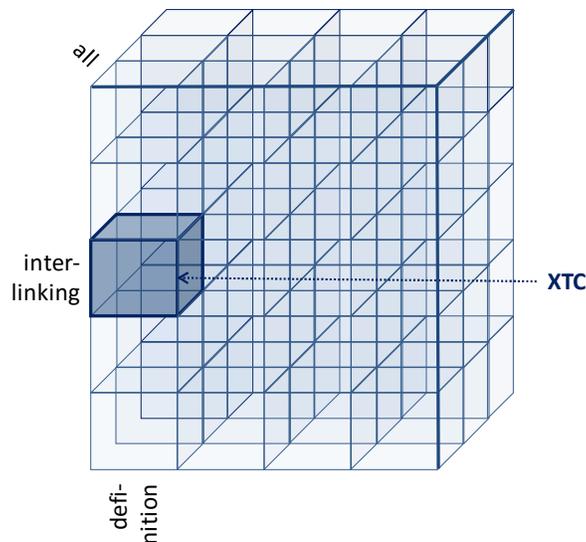


Figure 19: position of XTC project in dynamism space

5.2.2 SYNERGY

SYNERGY [Lor10] is a European FP7 research project aiming at the development of dynamic and adaptive knowledge management systems and services to support flexible collaboration in virtual organizations. SYNERGY started in 2008.

Dynamism in SYNERGY is supported through the use of collaboration patterns (called *CPats* in the project) that can be dynamically changed through so-called *reorganization patterns* (implemented in the Maestro dynamic orchestration engine). The SYNERGY approach supports reconfiguration of running process instances [Lor10], which places it at the *single* value in the *scope* dimension and at the *execution time* value of the *timing dimension*. The reorganization patterns allow changing the actual control flow of collaborations - this places the approach at the *construction* value of the *intensity* dimension. This analysis is summarized in Figure 20.

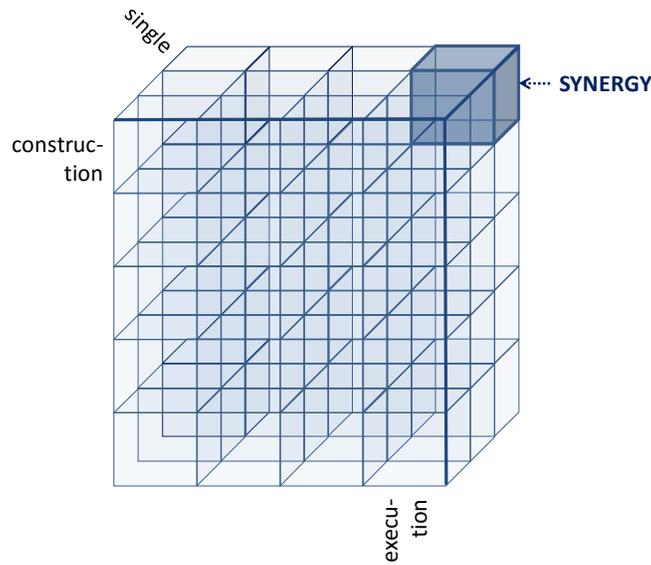


Figure 20: position of SYNERGY project in dynamism space

5.2.3 CoProFind

CoProFind [Gr13a, Gr13c] is an academy-industry collaboration project in the Netherlands, which started in 2009. Aim of the project is the development of a framework for agile, service-dominant, network-based business [Gr15a]. Agility is created by composing business models from existing business services in business networks. Business models are mapped to service compositions [Tr15a]. The approach has been applied among others in various contexts in the mobility domain [Lüf14,Gr15b,Tr15b].

Dynamism is supported in CoProFind by making the definition of business models and their operationalization in service orchestrations light, such that business models can be easily replaced. This places the project at the *definition time* value in the *timing* dimension of our classification space. New business models pertain to all customer orders in a collaboration, so the appropriate value in the *scope* dimension is *all instances*. The fact that the operationalization of new business models is created by combining existing services determines that the value in the *intensity* dimension is *interlinking*. This results in a position as shown in Figure 21.

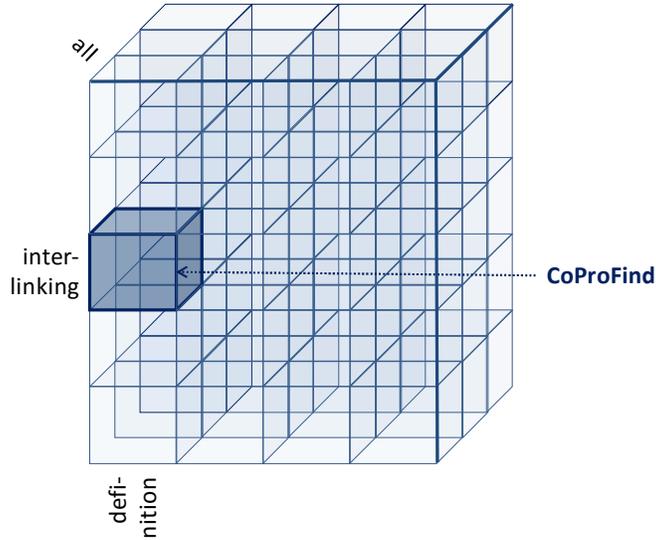


Figure 21: position of CoProFind project in dynamism space

5.2.4 GloNet

GloNet is an EU FP7 project aiming at designing, developing, and deploying an agile virtual enterprise environment for networks of SMEs involved in highly customized and service-enhanced products through end-to-end collaboration with customers and local suppliers [Glo16]. The approach of the project is based on semi-automated software service integration [Afs15]. The GloNet project started in 2011.

GloNet is aimed at the orchestration type level, so we place it at the *all* value in the *scope* dimension. Service orchestrations can be changed at *deployment time*, which determines the value in the *timing* dimension. The approach allows *assembly* of local services in business networks, which determines the value in the *intensity* dimension. The resulting classification of the GloNet project is shown in Figure 22.

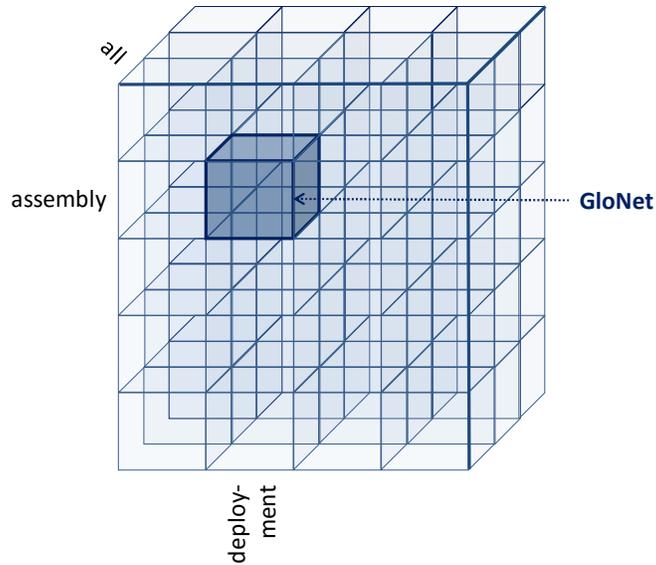


Figure 22: position of GloNet project in dynamism space

- In the *scope* dimension, we observe a light trend: old projects appear at all values, but most recent projects are at the *single instance* value. A very preliminary conclusion may be that there is a trend towards instance-based approaches.
- In the *intensity* dimension, there is a slightly clearer trend: older projects are positioned towards the low end of the dimension (*parameterization*), whereas newer projects are positioned towards the high end (*construction*). Again a very preliminary conclusion is that more recent research efforts concentrate on more advanced mechanisms to obtain dynamism.

We can also observe that there are not many projects addressing the *batch of instances* value in the *scope* dimension - CrossFlow and CrossWork are the only projects at this value in our set. Most projects concentrate either at *all instances* (i.e., the type level) or at *single instance* (i.e., the instance level). Likewise, we see a tendency of projects to concentrate on the extreme values in the *timing dimension*: most concentrate on *definition time* or *execution time* (CrossFlow and ADVENTURE are the exceptions). In the *intensity* dimension, we see a more even spread - although the *parameterization* value does not seem to attract much attention from research in our selection - this may be caused, however, by the fact that we have selected projects that explicitly aim at support for dynamic collaboration.

6 Conclusions and future work

In this paper, we have developed a three-dimensional framework for positioning and analyzing research in inter-organizational business process management and service orchestration. This framework has been materialized in what we call the *dynamism space*. We have applied this space to a set of well-known research directions and a set of concrete research projects.

There are two main directions of future work to extend what is reported in this paper. Firstly, we aim at extending the set of analyzed research projects, such that we can make observations about the state of the art in a broader setting. Secondly, we want to apply the dynamism cube to industrial application domains to chart the dynamism requirements from business practice. This will allow confronting technology push and requirements pull in the field of dynamic service orchestration and inter-organizational business process management.

Another important direction for future work is the inclusion of pragmatic aspects into our analysis framework, i.e., paying attention to the reasons for switching between variants of orchestration specifications. Most interesting in this context is the *source of change* dimension (indicated by the *by whom* interrogative). In this dimension, we can distinguish between changes caused by the different roles in service management: service providers, service consumers, or service orchestrators.

References

- [Aal00] W. van der Aalst; *Loosely Coupled Interorganizational Workflows: Modeling and Analyzing Workflows Crossing Organizational Boundaries*; Information and Management, Vol. 37, No. 2; Elsevier, 2000; pp. 67-75.
- [Afs15] H. Afsarmanesh, M. Sargolzaei, M. Shadi; *Semi-automated Software Service Integration in Virtual Organisations*; Enterprise Information Systems, Vol. 9, Nos. 5-6; Taylor & Francis, 2015; pp. 528-555.
- [Alo97] G. Alonso, C. Hagen, H.J. Schek, M. Tresch; *Distributed Processing over Stand-alone Systems and Applications*; Proceedings 23rd International Conference on Very Large Databases, 1997; pp. 575-579.

- [Alo99] G. Alonso, U. Fiedler, C. Hagen, A. Lazcano, H. Schuldt, N. Weiler; *WISE: Business to Business E-Commerce*; Proceedings 9th International Workshop on Research Issues in Data Engineering; Australia, 1999; pp. 132-139.
- [Ba15a] A. Baumgrass, R. Dijkman, P. Grefen, S. Pourmirza, H. Voelzer, Mathias Weske; *A Software Architecture for Transportation Planning and Monitoring in a Collaborative Network*; Proceedings 16th IFIP Working Conference on Virtual Enterprises; Albi, France; Springer; 2015.
- [Ba15b] A. Baumgrass, M. Botezatu, C. Di Ciccio, R. Dijkman, P. Grefen, Marcin Hewelt, Jan Mendling, Andreas Meyer, Hagen Voelzer; *Towards a Methodology for the Engineering of Event-driven Process Applications*; Proceedings First International Workshop on Process Engineering; Innsbruck, Austria; Springer; 2015; pp. 277-284.
- [Bro06] A. Brogi, R. Popescu; *Automated Generation of BPEL Adapters*; Proceedings International Conference on Service-Oriented Computing; Chicago, USA, 2006; Springer, 2006; pp. 27-39.
- [Eco11] Economist Intelligence Unit; *The Complexity Challenge: How Businesses are Bearing Up*; The Economist, 2011.
- [Fdh12] W. Fdhila, S. Rinderle-Ma, M. Reichert; *Change Propagation in Collaborative Processes Scenarios*; Proceedings 8th International Conference on Collaborative Computing: Networking, Applications and Worksharing; 2012; pp. 452–461.
- [Glo16] *GloNet - Glocal Enterprise Network Focusing on Customer-Centric Collaboration*; GloNet Project Web Site; <http://www.glonet-fines.eu/>; Inspected March 2016.
- [Gre00] P. Grefen, K. Aberer, Y. Hoffner, H. Ludwig; *CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises*; Computer Systems Science & Engineering, Vol. 15, No. 5; CRL Publishing, 2000; pp. 277-290.
- [Gre09] P. Grefen, R. Eshuis, N. Mehandjiev, G. Kouvas, G. Weichhart; *Internet-Based Support for Process-Oriented Instant Virtual Enterprises*; IEEE Internet Computing, Vol. 13, No. 6; IEEE, 2009; pp. 65-73.
- [Gr13a] P. Grefen, E. Luftenegger, E. v.d. Linden, C. Weisleder; *BASE/X: Business Agility through Cross-Organizational Service Engineering*; Beta Working Papers 414; Eindhoven University of Technology, 2013.
- [Gr13b] P. Grefen, R. Dijkman; *Hybrid Control of Supply Chains: a Structured Exploration from a Systems Perspective*; International Journal of Production Management and Engineering, Vol. 1, No. 1; 2013.
- [Gr13c] P. Grefen; *Networked Business Process Management*; International Journal of IT/Business Alignment and Governance, Vol. 4, No. 2; IGI Global, 2013; pp. 53-81.
- [Gr15a] P. Grefen; *Service-Dominant Business Engineering with BASE/X - Practitioner Business Modeling Handbook*; Amazon CreateSpace, 2015; ISBN 978-1516942176.
- [Gr15b] P. Grefen, O. Turetken, K. Traganos, A. den Hollander, R. Eshuis; *Creating Agility in Traffic Management by Collaborative Service-Dominant Business Engineering*; Proceedings 16th IFIP Working Conference on Virtual Enterprises; Albi, France; Springer, 2015; pp. 100-109.
- [Hea12] B. Heaney; *Supply Chain Visibility Excellence*; Aberdeen Group, 2012.
- [Hoff01] Y. Hoffner, S. Field, P. Grefen, H. Ludwig; *Contract Driven Creation and Operation of Virtual Enterprises*; Computer Networks, Vol. 37, No. 2; Elsevier, 2001; pp. 111-136.
- [Ket97] W. Kettinger, J. Teng, S. Guha; *Business Process Change: A Study of Methodologies, Techniques and Tools*; MIS Quarterly, March 1997; pp. 55-79.

- [Koe00] M. Koetsier, P. Grefen, J. Vonk; *Contracts for Cross-Organizational Workflow Management*; Proceedings 1st International Conference on Electronic Commerce and Web Technologies; London, UK, 2000; pp. 110-121.
- [Kry13] N. Kryvinska, R. Olexova, P. Dohmen, C. Strauss; *The S-D Logic Phenomenon - Conceptualization and Systematization by Reviewing the Literature of a Decade (2004–2013)*; Journal of Service Science Research, Vol. 5, No. 1; Springer, 2013; pp 35-94.
- [Laz00] A. Lazcano, G. Alonso, H. Schuldt, C. Schuler; *The WISE approach to Electronic Commerce*; International Journal of Computer Systems Science & Engineering, Vol. 15, No. 5, September 2000.
- [Laz01] A. Lazcano, H. Schuldt, G. Alonso, H. Schek; *WISE: Process Based E-Commerce*; IEEE Data Engineering Bulletin, Vol. 24, No. 1; IEEE, 2001; pp. 46-51.
- [Lie98] H. Lienhard; *IvyBeans - Bridge to VSH and the project WISE*; Proceedings Conference of the Swiss Priority Programme Information and Communication Structures, 1998.
- [Lor10] J. Lorre, Y. Verginadis, N. Papageorgiou, N. Salatge; *Ad-hoc Execution of Collaboration Patterns using Dynamic Orchestration*; in: *Enterprise Interoperability IV - Making the Internet of the Future for the Future of Enterprise*; Springer, 2010; pp. 3-12.
- [Lüf12] E. Lüftenegger, P. Grefen, C. Weisleder; *The Service Dominant Strategy Canvas: Towards Networked Business Models*; Proceedings 13th IFIP Working Conference on Virtual Enterprises; Springer; 2012; pp. 207-215.
- [Lüf14] Egon Lüftenegger; *Service-Dominant Business Design*; Ph.D. Thesis; Eindhoven University of Technology, 2014
- [Meh10] N. Mehandjiev, P. Grefen (eds.); *Dynamic Business Process Formation for Instant Virtual Enterprises*; Springer, 2010.
- [Pap06] M. Papazoglou, B. Kratz; *A Business-Aware Web Services Transaction Model*; Proceedings 4th International Conference on Service-Oriented Computing; Chicago, USA, 2006; pp. 352-364.
- [Rin14] S. Rinderle-Ma, P. Grefen; *Towards Flexibility in Transactional Service Compositions*; Proceedings 21th IEEE International Conference on Web Services; Anchorage, USA; IEEE, 2014; pp. 479-486.
- [Sab12] R. Eid Sabbagh, R. Dijkman, M. Weske; *Business Process Architecture: Use and Correctness*; Proceedings 10th International Conference on Business Process Management; Springer, 2012; pp. 65-81.
- [Sal11] A. Salmen, T. Münch, S. Buzin, J. Hladik, W. Altmann, C. Weber, D. Karagiannis, R. Buchmann, J. Ziegler, J. Pfeffer, M. Graube, M. Carnerero, O. López, M. Uriarte, P. Órtiz, O. Lázaro; *ComVantage: Mobile Enterprise Collaboration Reference Framework and Enablers for Future Internet Information Interoperability*; in: *The Future Internet - Future Internet Assembly 2013: Validated Results and New Horizons*; Springer, LNCS 7858, 2013.
- [Seg14] R. Seguel, R. Eshuis, P. Grefen; *Architecture Support for Flexible Business Chain Integration using Protocol Adaptors*; International Journal of Cooperative Information Systems; Vol. 23, No. 4; World Scientific; 2014; pp. 1450008-1-1450008-31.
- [Sch12] S. Schulte, D. Schuller, R. Steinmetz, S. Abels; *Plug-and-Play Virtual Factories*; Internet Computing, Vol. 16, No. 5; IEEE, 2012; pp. 78–82.
- [Ste46] S. Stevens; *On the Theory of Scales of Measurement*; Science, Vol. 103, No. 2684; American Association for the Advancement of Science, 1946; pp. 677-680.

- [Tr15a] K. Traganos, P. Grefen; *Hybrid Service Compositions: When BPM Meets Dynamic Case Management*; Proceedings 4th European Conference on Service-Oriented and Cloud Computing; Taormina, Italy; Springer / IFIP, 2015; pp. 226-239.
- [Tr15b] K. Traganos, P. Grefen, A. den Hollander, O. Turetken, R. Eshuis; *Business Model Prototyping for Intelligent Transport Systems: A Service-Dominant Approach*; Beta Working Papers 469; Eindhoven University of Technology, 2015.
- [Var04] S. Vargo, R. Lusch; *Evolving to a New Dominant Logic for Marketing*; Journal of Marketing Vol. 68, No. 1; American Marketing Association, 2004; pp. 1-17.
- [Wan06] T. Wang, P. Grefen, J. Vonk; *Abstract Transaction Construct: Building a Transaction Framework for Contract-Driven, Service-Oriented Business Processes*; Proceedings International Conference of Service-Oriented Computing; Chicago, USA; 2006; pp. 434-439.
- [Wan08] T. Wang, J. Vonk, P. Grefen; *Towards a Contractual Approach for Transaction Management*; Enterprise Information Systems, Vol. 2, No. 4; Taylor & Francis, 2008; pp. 443-458.
- [Wi16a] Wikipedia; *Level of Measurement*; http://en.wikipedia.org/wiki/Level_of_measurement; Inspected March 2016.
- [Wi16b] Wikipedia; *Zachman Framework*; http://en.wikipedia.org/wiki/Zachman_Framework; Inspected March 2016.
- [Zac02] J. Zachman; *The Zachman Framework for Enterprise Architecture*; Zachman International, 2002.