Stra2Bis: a model-driven framework for aligning business strategy and business processes

Rene Noel[1], Jose Ignacio Panach[2], Marcela Ruiz[3], and Oscar Pastor[1]

[1] PROS-VRAIN: Valencian Research Institute for Artificial Intelligence, Universitat Politècnica de València, València, Spain Valencia, Spain
{rnoel@vrain.upv.es, opastor@disc.upv.es}
[2] Escola Tècnica Superior d'Enginyeria, Universitat de València, València, Spain
{joigpana@uv.es}
[3] Zürich University of Applied Sciences, Winterthur, Switzerland
{marcela.ruiz@zhaw.ch}

Abstract. MDA-based initiatives for software development have included computation-independent models to align information system models with business knowledge which is important in the development process. One source of business knowledge is the business strategy, which, traditionally, has had a long-term perspective; changes in the organisational structure and their high-level ends and means were less frequent and arguably not relevant for software development. However, organisations that aim to accelerate their software development cycles define their business strategy and reconfigure their structure on a short-term, continuous basis, fusing, splitting and creating as independent as possible organisation units. These changes directly affect the business processes and the design of software components of the organisation. Based on this approach to business strategy, we propose Stra2Bis, a framework for designing strategically aligned business processes in an MDA-based context. Stra2Bis introduces a business strategy modelling step when redesigning business processes and three transformation guidelines to support the analysis of the alignment of processes with the organisational structure and the measurement of the units' outcomes. We discussed the effect of the guidelines on the software design with five professionals the supported the feasibility and usefulness of the proposal.

Keywords: model-driven architecture · business process · business strategy

1 Introduction

The Model-Driven Architecture (MDA) [28] approach has been used for designing and developing information systems for ensuring that the software products fulfil the business requirements. Computation-independent models (CIM) in MDA-based initiatives have been widely used for specifying business requirements for the system, mainly in terms of stakeholder's goals, business processes and use cases [15]. Other high-level business concepts have been included less

frequently at the CIM level [6], despite their usefulness for helping software developers make the most of the business knowledge.

One important source of business knowledge is business strategy, which addresses high-level organisational ends and the means to achieve them [21]. The scope of business strategy is broad and involves several concepts, such as the definition of organisational goals, and the action plans to achieve them. It also deals with how to structure the organisation units (groups of people such as departments, areas, or teams) for deploying the strategy and the capabilities needed. Most of these concepts have been addressed by enterprise architecture modelling languages [30, 29]; however, to the best of our knowledge, they have not been completely integrated into MDA-based methods for software development.

Traditionally, strategic decisions have had a long term perspective. Suppose an organisation decides to fuse two business areas. In that case, it requires a considerable effort to re-design the organisational structure, processes, and systems and several years for implementation. This drives the need for analysing competing goals from different stakeholders across the organisation and aligning business processes, which has been addressed by goal modelling frameworks and included in MDA-based methods [27, 13, 25, 32, 19, 1, 14, 22, 17, 6]. However, organisations whose value offer depends on software [12, 16] have a different approach to business strategy and alignment. These organisations continuously reconfigure their structure to foster the independence of their organisation units, which is translated to more efficient business processes and information systems [12]. These organisations carefully manage the dependencies among organisation units [2] and define precise strategic objectives, whose measurement is a critical part of their software products [16]. Proof of the importance of including business strategy information in the software development process is the fact that the agile community have considered it a critical part of software delivery [26, 20]. Also, broadly adopted software design techniques take a strategic approach for separating business domains [11] and for designing microservices [34].

Based on the above facts, we believe that it is important to include business strategy information about the structure and dependencies of organisation units and their strategic objectives as a source of business knowledge in MDA-based approaches for software development. This paper presents Stra2Bis, a framework for integrating business strategy information into the CIM level of an MDA-based approach for software development. Stra2Bis proposes 1. Modelling a business strategy scenario before business process design, and 2. Three transformation guidelines from the business strategy model to the business process model elements. The guidelines help to reason about strategic alignment by tackling three strategic concerns of organisations whose value depends on software: 1. Mapping organisation units at the strategic level to independent business processes, 2. Mapping organisation units' dependencies to interactions between their respective business processes, and 3. Mapping strategic objectives to business activities for collecting data about the objectives' status. We performed an initial exploratory evaluation of the proposal through a focus group with software development professionals, who confirmed the proposal's value.

2 Related Work and Motivation

Several initiatives that combine modelling languages have tackled the design of business processes aligned with strategy. Goal modelling languages have been used, for instance, to analyse whether business process activities (modelled using BPMN) support organisational goals (modelled with TROPOS) [14], or to analyse how business processes constraint business goals (modelled using KAOS)[22]. The Goal-Oriented Requirements Language (GRL) has been combined with Use Case Maps to model strategically aligned processes in the last two decades [1] and also to prioritise business processes [17]. MAP models (that define goals and the strategies to achieve them) have been mapped directly to the business processes elements that operationalize them [19] and also served to analyse the purpose behind the creation, modification, and deletion of business process elements [32]. I* models have been used for transforming social dependencies into interactions at the process level [25], validating the consistency of the process interactions [13], and checking whether the business processes have the elements needed to collect information to verify the goal achievement [27].

Besides goal modelling, other initiatives have combined frameworks addressing business strategy concerns. Business plans (modelled in Business Motivation Model [29]) have been used jointly with i* to add intentionality to the process of enterprise architecture construction [33]. Business value models (modelled using the e3Value method) have been used for generating performance requirements for an enterprise architecture [8]. In [4], organisational capabilities, modelled at the enterprise architecture level, are the starting point for the model-driven development of context-adapting software systems.

On the other hand, organisations whose value offer depends on software, such as digital or traditional organisations in an incremental digital transformation process, achieve alignment by managing their organisational structure. As researched by Forsgren et al. [12], independent, cross-disciplinary organisation units or teams yield loosely coupled systems, which improve software development performance and scalability. Most of the agile software development frameworks have adopted this approach [26, 20], which is based on the principle that organisations replicate their communication structure to everything they design, following Conway's Law [5]. Inverse Conway Manoeuvre [12] is an approach for evolving the organisational structure so business architecture matches the desired system architecture. Another key element is to measure the performance of the units' outcomes in terms of well-defined and measurable objectives [7, 16, 12] which are aligned with the customer value offered by the organisation unit.

We conclude from the model-driven initiatives that this approach is still a powerful tool for strategic alignment. However, while stakeholders' strategic goals and actions have been the main driver of alignment, strategic decisions about organisational structure have not been addressed by MDA approaches. Inspired by the idea of evolving organisational and business structure to get to an aligned system design, we aim to contribute to the existing MDA initiatives by aligning organisational structure and business processes at the CIM level, so this knowledge can be exploited for system design at the PIM level.

3 The Stra2Bis Framework

Stra2Bis is a framework to include business strategy information into an MDA-based software development method. Stra2Bis helps business analysts design business processes aligned to business strategy, particularly with the organisational structure and the strategic objectives. Stra2Bis prevents the analysts from designing business processes that couple many organisation units (e.g., departments, areas, teams), harming the software development efficiency [12]. It also prevents analysts from designing processes that miss information critical to measuring the achievement of strategic objectives.

Stra2Bis proposes three model-to-model transformation guidelines. The input for the transformation is a business strategy model that represents the strategic scenario that drives the need for re-designing a business process. The output is a partial business process model, with elements that serve as a scaffold to design strategically-aligned processes. This business process model is meant to be completed by the business analyst according to the problem domain and, through existing CIM-to-PIM transformation techniques [9, 6, 15], to be transformed into the information system model and then generate the code of the software system.

We describe Stra2Bis through a working example as a three-step business process improvement cycle in the following subsections. In Step 1, we introduce the working example and its business process and information system models. In Step 2, we present the business strategy model that will serve as input for the transformation guidelines. Step 3 details the guidelines and the resulting, re-designed business process model. Even though the contribution of Stra2Bis is focused on the CIM level, we also comment on the effects of the business strategy information on the PIM level using a microservices refactoring example[4].

3.1 Step 1: Current Business Process Model (Working Example)

In this step, the current business process is modelled. The notation proposed is from the Communication Analysis (CA) method [9]. We choose this notation because CA, in the same way as BPMN's choreography diagram [24] is not focused on the work performed but on the information exchange between the process actors. Moreover, CA has been integrated into an MDA-based development process, having theoretical consistency and technical feasibility for generating information system models and software code [10].

Working Example: F-FOOD is a software-as-a-service company that allows **consumers** to order food from **restaurants**, for pickup or for delivery. After the restaurant confirms an order, the delivery orders are scheduled to the closest available **courier**. F-FOOD has had exponential growth since its foundation and most of its software development efforts have been focused on mobile applications. However, the back end is still a monolithic application.

Fig. 1.A presents the business process model for the current situation. In order to later discuss the effects of the Stra2Bis guidelines on the design of

[4] https://microservices.io/refactoring/

software components, we also present a class diagram of the current information system in Fig. 1.B. Please note that there is no a `Delivery` class in the domain model and that `scheduledelivery` is a service offered by `OrderService`.

3.2 Step 2: Business Strategy Modelling

This step proposes modelling the strategic scenario that drives the business process re-design. We propose using the LiteStrat [23] method. LiteStrat is our

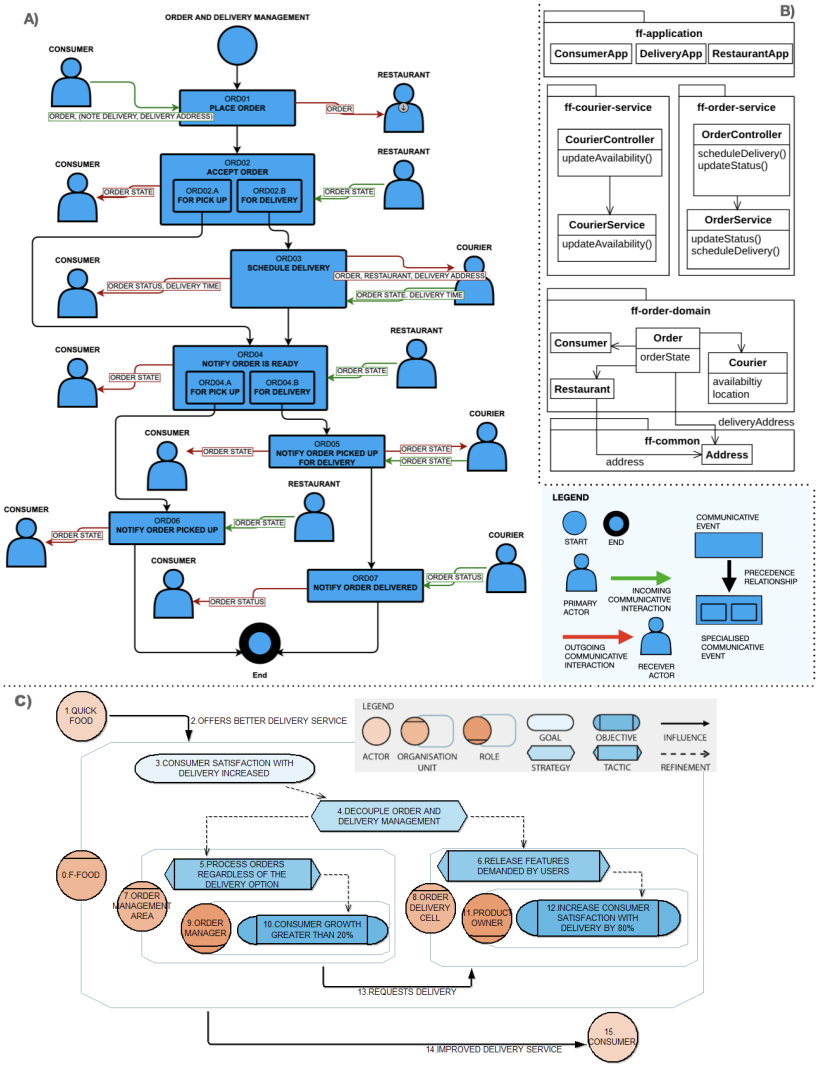

Fig. 1. Current situation models: A) Business process model. B) Class diagram of the Information System. C) Business strategy model.

previous work that proposes a business strategy modelling language to represent the organisational structure, strategy, and goals jointly, as well as a modelling procedure to reduce the variability of models to improve their integration in MDA contexts. However, other goal or enterprise architecture notations can be used, while they support representing: 1. The organisation units that are affected by the strategic definitions, 2. The new dependencies between the organisation units that are generated by the strategic definitions, and 3. The measurable objectives to assess the strategy implementation. Fig. 1.C presents a LiteStrat model for the strategic scenario described below. The parenthesis indicates the model elements associated with the description.

Strategic Scenario: In the last quarter, the growth of consumers in F-FOOD (0) has decreased. F-FOOD's finds out that a new competitor, QUICK-FOOD (1), has a better order delivery service (2). Consumers claim that the F-FOOD app lacks several features for delivery tracking and has a slow response when putting delivery orders. F-FOOD discovers that the Order Management Area (7) constantly gives a lower priority to new delivery features and optimisations, favouring the order management functionality. F-FOOD management has decided that consumer satisfaction with the delivery is the top strategic goal for the next quarter (3). To achieve this goal, the strategy is to decouple the delivery service as an independent service (4), owned by a new cross-disciplinary team called Order Delivery Cell (8) that is meant to release all the features demanded by the customers (6). The Product Owner (11) will track the objective of increasing consumer satisfaction with delivery by 80% (12). The Order Management Area will have a leaner order processing, regardless of their delivery option (5) and will depend on the Order Delivery Cell for delivering the orders (13). New consumers are expected to increase by a 20% (10), which will be tracked by the Order Manager (9). The implementation of the strategy seeks to offer an improved delivery service (14) for the consumers (15).

3.3 Step 3: Transformations Guidelines from Strategy to Process Model

In this step, we take as input the business strategy model from Step 2 and apply three transformation guidelines to generate an initial version of the redesigned business process model. A guideline is a recommendation for designing parts of a business process model, taking into account elements from business strategy. Guideline 1 deals with organisation units, Guideline 2 with organisation units dependencies, and Guideline 3 with strategic objectives. As with other MDA transformations at the CIM level, the guidelines support a semi-automatic, skilled transformation process so that the analysts can change the mapped process parts according to the real-world context.

For each guideline, we first define it and then detail the transformation algorithm. The algorithm is described using some specific terms of the CA and LiteStrat notations, for which we provide a metamodel summarising the transformations in Fig. 2. We briefly discuss the benefits of the guidelines based on

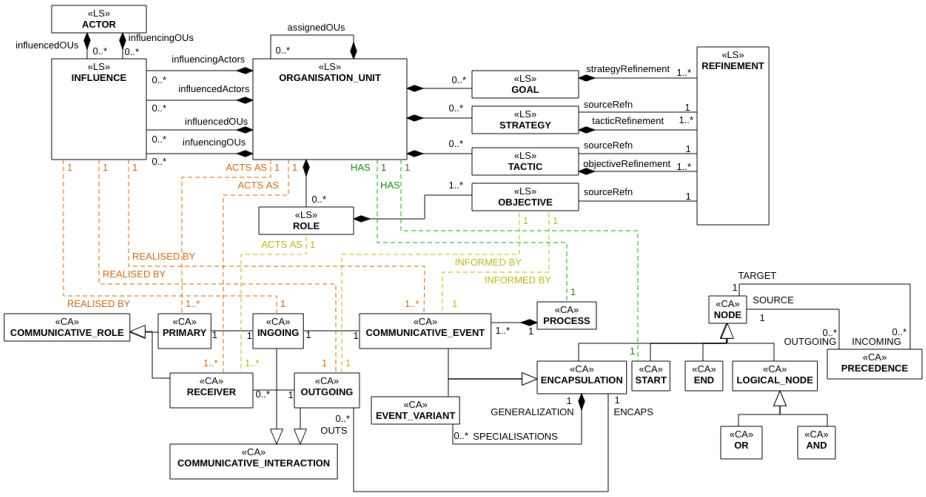

Fig. 2. Metamodel mappings for LiteStrat (LS stereotype) [23] and a simplified Communication Analysis (CA stereotype) [10] metamodels. Relationships for Guidelines 1, 2, and 3 are coloured in green, orange, and yellow, respectively.

existing literature and how they can be applied in cases different from the working example. Finally, we describe the application of the guideline in the working example that produces the model depicted in Fig. 3.A. The algorithm omits the details for naming generated model elements for brevity, but they can be inducted from the generated business process model in the example.

Guideline 1: *For each organisation unit in the business strategy, create a new process for the organisation unit in the business process model.*

Algorithm: Let LS be the business strategy model, and $LS.OU_0$ the organisation unit that is not contained in other organisation units, thus, the organisation. $LS.IOU_i$ are all the organisation units in the strategy model that are contained in the organisation $LS.OU_0$. Let CA be the new business process model. For each organisation unit $LS.IOU_i$, create a new process $CA.Process_i$. For each process created $CA.Process_i$, create a new start event $CA.Start_i$. The name of the node follows the name of the organisation unit. The mappings between the metamodel elements are shown in green in Fig. 2.

Analysis Guideline: this guideline is based on the research by Forsgren et al. [12], who found that the coupling between organisation units processes has been reported as a hindering factor for efficient software development in high performing technology organisations. The generated elements in the business process model reflect the ideal separation of processes. The analyst should assess whether this separation is feasible considering the actual context of the problem.

Example: In the business strategy model in Fig. 1.C, "Order Management Area" and the new "Order Delivery Cell" units originate the "Order Manage-

ment" and "Delivery Management" processes depicted as green start nodes in Fig. 3.A. The start nodes are named following the names of their respective organisational units. The guideline proposes designing an independent business process for the delivery service, otherwise, the new team would still be coupled to the Order Management Area process. Although the example specifically regards the split of an existing unit, the guideline is also helpful in analysing the creation, fusion, or hiring of external teams for tackling new business opportunities.

Guideline 2: *For each dependency among organisation units in the business strategy, add events to the organisation units' processes to handle the dependency.*

Algorithm: Let $LS.IntInfl_i$ be all the dependencies between the organisation units (OUs) $LS.IOU_i$ in the strategy model. Let $LS.IntInflSrcOU_i$ the source OU of the dependency $Ls.IntInfl_i$ and $CA.SrcProcess_i$ the process created for the source unit $LS.IntInflSrcOU_i$ in Guideline 1. Let $LS.IntInflTrgtOU_i$ the target OU of $Ls.IntInfl_i$ and $CA.TrgtProcess_i$ the process created for the target organisation unit $LS.IntInflSrcOU_i$ in Guideline 1. Map each dependency $IntInfl_i$ as a new event $CA.SrcProcessEvent_i$ in the **source process** $CA.SrcProcess_i$. Create a receiver actor $CA.SrcProcessReceiverActor_i$, following the name of the target organisation unit $LS.IntInflTrgtOU$. Create an outgoing communicative interaction $CA.SrcProcessOutgoingCI_i$ from the event $CA.SrcProcessEvent_i$ to the receiver actor $CA.SrcProcessReceiverActor_i$. Also, map the same dependency $IntInfl_i$ as a new event $CA.TrgtProcessEvent_i$ in the **target process** $CA.TrgtProcess_i$. Create a primary actor $CA.TrgtProcessPrimaryActor_i$ following the name of the target organisation unit $LS.IntInflTrgtOU$. Create an ingoing communicative interaction $CA.TrgtProcessIngoingCI_i$ from the primary actor $CA.TrgtProcessPrimaryActor_i$ to the created event in the target process $CA.TrgtProcessEvent_i$.

Analysis Guideline: This guideline is based on the need to manage and reduce the dependencies among organisation units to foster their autonomy, which is a practice followed by operational models such as the Spotify Model [2] and EDGE [16]. The guideline drives the analyst to specify how the organisation units' dependencies are operationalised at the process level. This could reveal the need for interoperability or refactoring of software components that could hinder the agile implementation of the strategy. The transformation introduces an actor (named after the name of the target organisation unit), which the analyst can rename according to the domain. The transformation does not specify a primary actor for providing the information for the interaction between processes; if the system already has all the information, there is no need for a primary actor. In another case, the analyst can add a primary actor that provides the information needed for the processes' interaction.

Example: The influence relationship "16.Requests Delivery" from the organisation unit "Order Management Area" to the "Order Delivery Cell" in Fig. 1.C is mapped as the events depicted in orange in Fig. 3.A: an event to perform the influencing behaviour (16.Requests Delivery), and an event to address the influence (DEL01-Handle Delivery Request). A new actor is introduced to han-

dle the dependency, representing the target organisation unit of the dependency (Order Delivery Cell). The name of the events and actors follow the strategy diagram, but the analyst can change them according to the domain information.

Guideline 3: *For each business strategy objective, add an event to their respective organisation unit' process to collect information about the objective's status*

Algorithm: Let $LS.IOU_i$ the organisation unit to which the role $LS.Obj_k$ belongs and $CA.Process_i$ the process created for the organisation unit $LS.IOU_i$ in Guideline 1. Let $LS.Obj_k$ be all the objectives of the organisation unit $LS.IOU_i$ and $LS.Role_k$ the role assigned with the objective $LS.Obj_k$. For each organisation unit $LS.IOU_i$, and for each their objectives $LS.Obj_k$, create a new event $CA.ObjectiveEvent_k$ int the process of the organisation unit $CA.Process_i$. Create new receiver actor $CA.ObjectiveEventEventReceiver_k$ named after the role assigned with the objective $LS.Role_k$. Create an outgoing communicative interaction $CA.ObjectiveOutgoing_k$ from the event $CA.ObjectiveEvent_k$ to the receiver actor $CA.ObjectiveEventEventReceiver_k$.

Analysis: This guideline is based on the practice of a shared measurement of the success of strategic initiatives, which is enforced by frameworks for digital transformation such as EDGE [16] and Objectives and Key Results (OKR) [7], and on the proposal by Sousa et al. [27] to collect goal achievement data in business process activities. The generated elements guide the analyst in collecting and delivering the information needed to monitor the objectives continuously. Similarly to guideline 2, the transformation does not generate a primary actor to provide the information. It will not be needed if the information is already in the system; otherwise, the analyst can add a primary actor according to the problem domain.

Example: In the strategy diagram in Fig. 1.C, the objectives "10.Consumer growth greater than 20%" of the organisation unit "Order Management Area" is mapped to the event "ORD06.Report Consumer Growth" in Fig. 3.A, depicted in yellow. Similarly, the objective "12.Increase consumer satisfaction with delivery by 80%" is mapped to the event "DEL06-Report Delivery Satisfaction". In both cases, the receiver actors are the roles assigned to the objectives in the strategy diagram (Order Manager and Product Owner).

3.4 Effects on the PIM level in an MDA context.

Stra2Bis guidelines are expected to affect the information system model at the PIM level. Although the mapping of business processes to the information system model is not part of this work (but has already been proposed in [10]), we exemplify in Fig. 3.B. the effects of the guidelines on the initial information system model was presented in Fig. 1.B.

Regarding Guideline 1, since the two organisation units *Order Management Area* and *Order Delivery Cell* had their separated business processes *Order Management* and *Order Delivery Management*, the `Delivery` domain class and services must be disentangled in a different component. Fig. 3.B shows in green

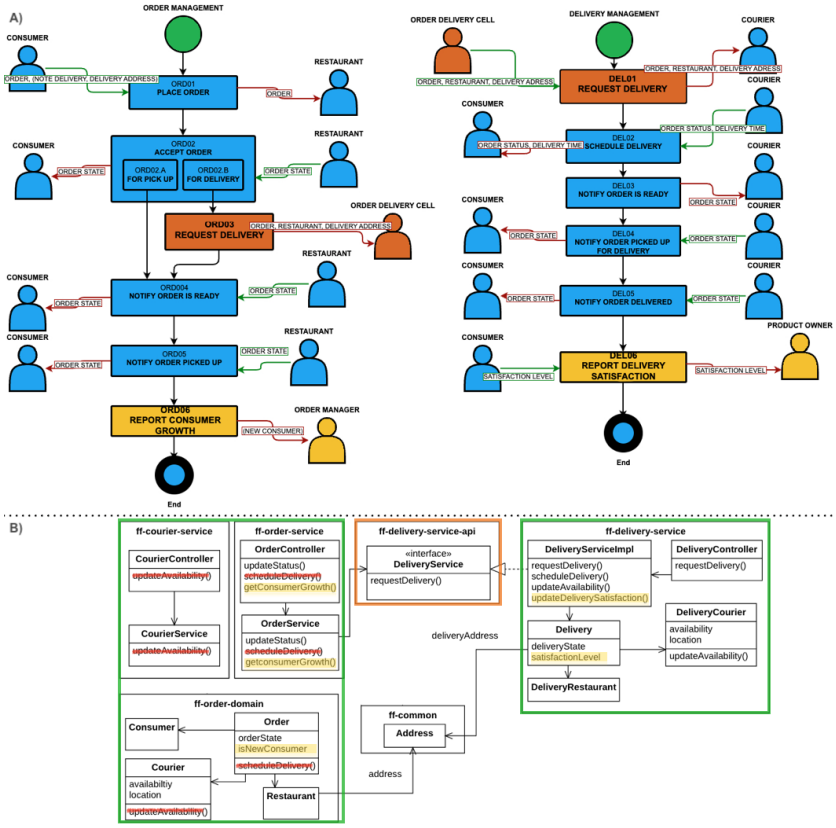

Fig. 3. A) Re-designed business process model. B Re-designed class diagram for the information system model.

the components for both processes. The new component `ff-deliver-service` supports the Order Delivery Process. Some services are removed from the order management components previously introduced in Fig. 1.B. The changes mainly consist of removing the delivery-related services that were initially located in the `ff-courier-service`, `ff-order-service` and `ff-order-domain` components and moving them to the new `ff-deliver-service` component.

Regarding Guideline 2, the interaction between the processes is mapped as an interface `ff-deliver-service-api` depicted in orange in Fig. 3.B. The interface is implemented by the component supporting the delivery process `ff-delivery-service`. It allows the initial order management system to request the services that were moved to the new `ff-delivery-service`.

Finally, the effects of Guideline 3 are mapped into services and attributes to update the values for the strategic objectives collected through the processes. As highlighted in yellow in Fig. 3.Bs, the Order class has a new attribute `isNewConsumer` to identify whether the order is from a new consumer. This helps

to track the objective "10.Consumer growth greater than 20%" objective initially defined in the strategy model in Fig. 1.C. In a similar way, the `Delivery` class has the attribute `satisfactionLevel` of the objective "12.Increase consumer satisfaction with delivery by 80%".

4 Initial Evaluation and Discussion

We conducted an exploratory evaluation through a focus group, since this technique is suitable for the *"initial evaluation of potential solutions, based on the practitioner or user feedback"* [18]. The research question was, "what information from the business strategy model is valuable for designing business processes?". The goal is to find whether practitioners' insights and experience match the Stra2Bis guidelines in terms of the information traceable from business strategy to business process and to information system model. We wanted to contrast opinions from practitioners working in traditional consultancy services companies (CSC) and in Software-as-a-Service companies (SaaS), which main value offer is based on software. The participants were five volunteers having technical leader or scrum master roles, with between four and nine years of experience. Participants S1, S3 work in CSC, and participants S2, S4, S5 work in a SaaS. The activity had two parts of 30 minutes each. First, we showed the working example from Figs. 1 and 1.C and asked "what information would be useful for redesigning business processes and why"?. The participants shared and agreed on a set of statements that were written down publicly by the moderator. In the second part, we presented the Stra2Bis guidelines and the models from Fig. 3, and asked the participants to comment on their usefulness and drawbacks. The analysis method was based on pattern-matching [18] the participant's ideas from the first part of the focus group with the guidelines and then looking for explanations from the discussion of the second part.

Insights for Guideline 1: In the first part, the respondents did not identify the organisation units as an important source of information for the business process design. After seeing the redesigned process and explaining Guideline 1, all the participants agreed that independent units must have independent processes. All of the respondents recalled difficulties when business processes of different units were entangled, including the software code. Respondent S2, who works in a SaaS, stated that *"it is important for us to have an independent business flow because each cell can take the challenges and opportunities of their own process"*.

Insights for Guideline 2: In the first part, all the respondents identified as relevant the dependency among the organisation units. S1 and S2 agreed that *"the dependency must be clear in the business process flow"*. All the participants agreed on the value of the guideline for defining the dependency at the process level. It is worth noting that respondents S1 and S3, from CSCs, claimed that sometimes the flow interactions were not well defined by "business people", requiring *"several meeting between teams to define the flow"* (S1). On the other hand, S2, who works on a SaaS, declared that his unit was designed with a well-

defined contract with other organisation units, so they never had problems of coupling problems at the business level.

Insights for Guideline 3: In the first part, just S1 identified as valuable the objectives and lnked them with OKR, one of the frameworks on which the guideline is based on [7]. In the second part, all the respondents valued measuring strategic objectives in the business process. Participants S4 and S5 commented *we have code written to measure the NPS*[5]. However, for the rest of the participants, the effect on the software product was different to what we presented in Section 3.4, who stated that objectives measurement are solved using external tools such as Hotjar[6] (for measuring customer satisfaction) or Google Analytics.

Two topics continuously appeared in the participants' responses that were not addressed by the guidelines. The first is how the actions assigned to an organisation unit in the strategy model (tactics in Fig. 1.C) are realised in the business process model. The second is participants associated the dependencies between units with software development process management concerns.

Considering the above results, we discuss three topics about: 1. When should business strategy models be used to redesign business processes?; 2. To which extent the guidelines could automatically produce part of the business process model?; and 3. How Stra2Bis could be integrated into model-driven methods?.

For the first topic, we believe that digital companies and traditional companies taking incremental steps for digital transformation can benefit from modelling business strategy and from following the guidelines to redesign their business processes since they are continuously responding to change and need this kind of alignment. This is a vision shared with other recent works [3, 31]. Modelling the strategic scenario that drives the creation (or hiring) of a development team could be the starting point to analysing whether the business process addressed by the new team is independent enough to work in an agile way, preventing the problems stated by S1 and S3 when analysing Guideline 2. For the second topic, we believe that the automatic, top-down transformation can be done and is useful since it provides an initial structure for further designing processes. Analysts can perform further modelling according to the problem domain, and constraints, which is common to other initiatives at the CIM level [14, 22, 17, 6]. Finally, Stra2Bis could be integrated into different MDA-based development methods jointly with the alignment approaches presented in Section 2 having as target a business process language. However, it would require mapping the Communication Analysis concepts to the corresponding language. We believe that, as shown in the focus group activities, Stra2Bis guidelines could also help reason about alignment in non-model-driven contexts.

5 Conclusions and Future Work

This article presented Stra2Bis, a framework for designing strategically aligned business processes in an MDA context. Stra2Bis proposes to align business pro-

[5] Net Promoter Score, https://hbr.org/2003/12/the-one-number-you-need-to-grow
[6] https://www.hotjar.com/

cesses to the organisational units' structure, dependency and goals. The framework proposes adding a strategy modelling step to represent the organisational elements that drive the business process re-design and three guidelines to generate an initial version of the new business process model. We conducted an initial evaluation through a focus group with eight software development practitioners, who supported all the proposals. However, the effects on software design could be different for adding strategic objectives measurement features. Although the respondents' profile, experience, and non-model-driven context set threats to validity to the focus group, the activity showed that the proposed guidelines were helpful for reasoning about the strategic alignment of business processes. Future work focuses on applying the proposal in an industrial case study and other focus groups and interviews with practitioners to foster the proposal's adoption.

References

1. Amyot, D., Akhigbe, O., Baslyman, M., Ghanavati, S., Ghasemi, M., Hassine, J., Lessard, L., Mussbacher, G., Shen, K., Yu, E.: Combining goal modelling with business process modelling. Enterprise Modelling and Information Systems Architectures (EMISAJ) **17**, 2–1 (2022)
2. Atlassian: The spotify model. https://www.atlassian.com/agile/agile-at-scale/spotify, (Accessed on 05/31/2021)
3. Babar, Z., Yu, E.: Digital transformation–implications for enterprise modeling and analysis. In: 2019 IEEE 23rd International Enterprise Distributed Object Computing Workshop (EDOCW). pp. 1–8. IEEE (2019)
4. Bērziša, S., Bravos, G., Gonzalez, T., et al.: Capability driven development: an approach to designing digital enterprises. Business & Information Systems Engineering **57**(1), 15–25 (2015)
5. Conway, M.E.: How do committees invent. Datamation **14**(4), 28–31 (1968)
6. De Castro, V., Marcos, E., Vara, J.M.: Applying cim-to-pim model transformations for the service-oriented development of information systems. Information and Software Technology **53**(1), 87–105 (2011)
7. Doerr, J.: Measure what matters: How Google, Bono, and the Gates Foundation rock the world with OKRs. Penguin (2018)
8. Engelsman, W., Gordijn, J., Haaker, T., Sinderen, M.v., Wieringa, R.: Quantitative alignment of enterprise architectures with the business model. In: International Conference on Conceptual Modeling. pp. 189–198. Springer (2021)
9. España, S., González, A., Pastor, Ó.: Communication analysis: a requirements engineering method for information systems. In: International Conference on Advanced Information Systems Engineering. pp. 530–545. Springer (2009)
10. España, S.: Methodological Integration of Communication Analysis into a Model-Driven Software Development Framework. Ph.D. thesis, Valencia (Spain) (2011)
11. Evans, E., Evans, E.J.: Domain-driven design: tackling complexity in the heart of software. Addison-Wesley Professional (2004)
12. Forsgren, N., Humbpotifle, J., Kim, G.: Accelerate: The science of lean software and devops building and scaling high performing technology organizations (2018)
13. Gröner, G., Asadi, M., Mohabbati, B., Gašević, D., Bošković, M., Parreiras, F.S.: Validation of user intentions in process orchestration and choreography. Information Systems **43**, 83–99 (2014)

14. Guizzardi, R., Reis, A.N.: A method to align goals and business processes. In: International Conference on Conceptual Modeling. pp. 79–93. Springer (2015)
15. Habba, M., Fredj, M., Benabdellah Chaouni, S.: Alignment between business requirement, business process, and software system: a systematic literature review. Journal of Engineering (2019)
16. Highsmith, J., Luu, L., Robinson, D.: EDGE: Value-Driven Digital Transformation. Addison-Wesley Professional (2019)
17. Insfrán, E., Abrahão, S., de Oliveira, R.P., González-Ladrón-de Guevara, F., Fernández-Diego, M., Cano-Genoves, C.: Specifying value in grl for guiding bpmn activities prioritization. In: Procedings of the International Conference on Information Systems Development (2017)
18. Kontio, J., Lehtola, L., Bragge, J.: Using the focus group method in software engineering: obtaining practitioner and user experiences. In: Proceedings of the 2004 International Symposium on Empirical Software Engineering (2004)
19. Kraiem, N., Kaffela, H., Dimassi, J., Al Khanjari, Z.: Mapping from map models to bpmn processes. Journal of Software Engineering **8**(4), 252–264 (2014)
20. Larman, C., Vodde, B.: Large-scale scrum: More with LeSS. Addison-Wesley Professional (2016)
21. Mintzberg, H.: The strategy concept i: Five ps for strategy. California management review **30**(1), 11–24 (1987)
22. Nagel, B., Gerth, C., Engels, G., Post, J.: Ensuring consistency among business goals and business process models. In: 2013 17th IEEE International Enterprise Distributed Object Computing Conference. pp. 17–26. IEEE (2013)
23. Noel, R., Panach Navarrete, J.I., Ruiz, M., Pastor Lopez, O.: The litestrat method: Towards strategic model-driven development. In: Procedings of the International Conference on Information Systems Development (2021)
24. von Rosing, M., White, S., Cummins, F., de Man, H.: Business process model and notation-bpmn (2015)
25. Ruiz, M., Costal, D., España, S., Franch, X., Pastor, O.: Gobis: An integrated framework to analyse the goal and business process perspectives in information systems. Information Systems **53**, 330–345 (10 2015)
26. Scaled Agile, INC: Safe 5 for lean enterprises. https://www.scaledagileframework.com/, (Accessed on 04/10/2021)
27. Sousa, H.P., Prado Leite, J.C.S.d.: Modeling organizational alignment. In: International Conference on Conceptual Modeling. pp. 407–414. Springer (2014)
28. The Object Management Group: Model driven architecture (mda). https://www.omg.org/mda/, (Accessed on 04/14/2021)
29. The Object Management Group: Business Motivation Model Specification Version 1.3 (2015), https://www.omg.org/spec/BMM/About-BMM/
30. The Open Group: The archimate® enterprise architecture modeling language. https://www.opengroup.org/archimate-home, (Accessed on 04/20/2022)
31. Tsilionis, K., Wautelet, Y.: A model-driven framework to support strategic agility: Value-added perspective. Information and Software Technology **141**, 106734 (2022)
32. Vara, J.L.d.l., Sánchez, J., Pastor, Ó.: Business process modelling and purpose analysis for requirements analysis of information systems. In: International Conference on Advanced Information Systems Engineering. pp. 213–227. Springer (2008)
33. Yu, E., Strohmaier, M., Deng, X.: Exploring intentional modeling and analysis for enterprise architecture. In: 2006 10th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW'06). pp. 32–32. IEEE (2006)
34. Zimmermann, O.: Microservices tenets. Computer Science-Research and Development **32**(3), 301–310 (2017)