Introduction

Nowadays, business process management techniques develop quickly in both academia and industry. To increase the flexibility and controllability of the management of organizations, business processes are used to describe the services that an organization provides and the internal processes that implement those services. As a result, it is common to see collections of hundreds or even thousands of business process models. For example, the collection of SAP reference models consists of more than 600 business process models (Curran & Keller, 1999), and the collection of the reference models for Dutch Local Government contains a similar number of models (Documentair Structuurplan). As business process model collections increase in size, tools and techniques are required to manage them. This includes tools and techniques for identifying and structuring the business processes in an organization.

A business process architecture is a means to structure a collection of business process models. It is defined as a (graphical) representation of the business processes that exist in an organization and the relations that they have with each other. A challenge in this field is then to design a business process architecture. This involves identifying the processes that exist in an organization and structuring them.

In previous work, Dijkman, Vanderfeesten and Reijers (2011) surveyed existing approaches to design a business process architecture, and determined how usable each of these approaches are in practice. Here, we present one concrete approach that is based their findings and the work by Obers and Achterberg (2009).

First, we briefly characterize the process architecture that is produced by the approach. Second, we present the approach itself.

Process Architecture Design in two Dimensions

The approach to process architecture design that we propose, leads to a process architecture in two dimensions: case type and business function.

The case type dimension classifies the types of cases that are handled by the organization. A case is something that an organization or a part of an organization handles. Typically a case is a product or service that is delivered by an organization to its customers, such as an insurance, a logistics service or a toy.
Note that, depending on the part of the organization for which the process architecture is designed, the cases can represent products or services that are delivered to the customers of the organization, but also products or services that are delivered by one department of the organization to another department. Cases can be classified as desired, using any number of properties. For example, an insurance company handles insurances, which can be classified according to product type (home-insurance, car-insurance and life-insurance), but also according to the channel that the company uses to interact with its customers (telephone, office and internet). A combination of these properties can also be used to classify cases. In the insurance example, cases would then be classified using both product type and channel (home-insurance via telephone, home-insurance via office, car-insurance via telephone, ...).

The function dimension classifies the functions of an organization. A function is something that an organization does. Typically, a hierarchical decomposition of functions can be made: a function consists of sub-functions, which, in turn, consist of other sub-sub-functions, etceteras. For example, a production company performs purchasing, production and sales functions. The purchasing function, in turn, can be decomposed into vendor selection and operational procurement functions.

Figure 1 shows an example of a business process architecture that uses the case type and function dimensions to structure its processes. It shows an architecture for a harbor authority. The figure shows an organization of processes by modality in the horizontal – case type – dimension and by business function in the vertical – function – dimension. The function dimension shows what the organization does: handling pre-arrival of sea ships, which involves notifying the relevant parties about the estimated time of arrival of the ship and what the ship is carrying; handling the actual arrival of the ship, which involves guiding the ship to its dock; etceteras. The case type dimension shows the types of cases that the organization handles: sea ships, trucks, trains and inland transportation by barge. There are three processes that are created to handle these types of cases, using the different functions. These three are shown as covering the various functions and case types. The inbound planning process is used for handling pre-arrival of sea ships. The inbound handling process is used for handling arrival and trans-shipment of sea ships and the outbound handling process is used for handling trans-shipment and departure of trucks, trains and barges.

![Figure 1. Example Business Process Architecture.](image-url)
An Approach to Process Architecture Design

This section proposes an approach to design a business process architecture as it is explained in the previous section. Thus, it can be used to identify and structure the business processes that exist in an organization. The approach consists of the following four steps, each of which we explain in detail below.

1. Identify case types
2. Identify functions for case types
3. Construct one or more case/function matrices
4. Identify processes

Step 1: Identify Case Types

In the first step, a classification of case types is developed for the organization. This is done by selecting the case properties that will be used for the classification. The main purpose for identifying different classes in this dimension of the process architecture, is to determine the different ways in which (similar) processes are handled in the organization. It is important to keep this in mind, because the only properties that should be included in the classification are the ones that lead to different behavior. Properties that may distinguish cases, but do not lead to different behavior, should not be included. For example, a retail store sells many different types of products. However, it sells all these types of product in the same manner. Therefore, ‘product type’ is not a useful dimension when classifying the cases that are handled by a retail store. An insurance company also sells different types of products (insurances) and, in contrast to the retail store, the products that it sells are handled differently. For example, for a life insurance a declaration of health must be filled out, but for a car insurance that is not necessary. Therefore, the ‘product type’ is a useful property to classify the types of cases that are handled by an insurance company, but not to classify the types of cases that are handled by a retail store.

A classification of the types of cases that an organization handles can be developed using any number of properties. However, some of the more commonly used properties are:

- Product Type;
- Service Type;
- Channel; and
- Customer Type.

The product type identifies the type of products that are handled by an organization. These can be hierarchically decomposed. For example, an insurance company handles ‘damages’ and ‘life’ insurance products. In the class of ‘damage insurances’, a further decomposition is possible into ‘car insurance’ and ‘home insurance’ and in the class of ‘life insurance’ a further decomposition is possible into ‘healthcare insurance’ and ‘accident insurance’. When (a part of) an organization handles services rather than products, the service type identifies the types of services that the organization handles, similar to the way in which ‘product type’ identifies the types of products that the organization handles. The channel represents the channel through which the organization contacts its customers. We can, for example, distinguish: face-to-face (over the counter) contact, telephone or internet contact. The customer type represents the types of customer that the organization deals with. An airline, for example, may distinguish frequent flyers from regular travelers.
Note again that, although these are the most commonly used properties to distinguish different case types, these are certainly not the only properties that can be used. Any property that distinguishes types of cases that are handled differently can be used. For example, if an organization does things differently in North America than in Europe, cases may be classified according to location. As another example, if cases are handled differently, depending on the expertise that is required to handle them, they may be classified according to expertise.

Also, note again that the classification can be developed using any number and combination of properties. If a company sells insurances in both North America and Europe and handling of insurances differs on those continents, because of local regulations, then a classification of cases according to both product type and location can be used.

**Step 2: Identify Functions for Case Types**

In the second step, a classification is developed of the business functions that are performed on the different case types. This step requires that each of the case types is examined in detail and for each case type, the functions that can be performed on it, are identified.

Potentially, the functions that are performed in an organization, can be related to existing classifications that are proposed by reference models. An example of such a reference model is the APQC reference model (APQC), of which a part is shown in Figure 2. Such reference models can serve as a starting point to develop a classification of business functions and be adapted to the specific needs of the organization.

<table>
<thead>
<tr>
<th>10002</th>
<th>1.0 Develop Vision and Strategy (10002)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10017</td>
<td>1.1 Define the business concept and long-term vision (10014)</td>
</tr>
<tr>
<td>10021</td>
<td>1.1.1 Analyze and evaluate competition (10021)</td>
</tr>
<tr>
<td>10022</td>
<td>1.1.2 Identify economic trends (10022)</td>
</tr>
<tr>
<td>10023</td>
<td>1.1.3 Identify political and regulatory issues (10023)</td>
</tr>
<tr>
<td>10024</td>
<td>1.1.4 Assess new technology innovations (10024)</td>
</tr>
<tr>
<td>10025</td>
<td>1.1.5 Analyze demographics (10025)</td>
</tr>
<tr>
<td>10026</td>
<td>1.1.6 Identify social and cultural changes (10026)</td>
</tr>
<tr>
<td>10027</td>
<td>1.1.7 Identify ecological concerns (10027)</td>
</tr>
<tr>
<td>10028</td>
<td>1.1.8 Survey markets and determine customer needs and wants (10028)</td>
</tr>
<tr>
<td>10029</td>
<td>1.1.9 Conduct qualitative/quantitative assessments (10029)</td>
</tr>
<tr>
<td>10030</td>
<td>1.1.10 Capture and assess customer needs (10030)</td>
</tr>
<tr>
<td>10031</td>
<td>1.1.11 Perform internal analysis (10031)</td>
</tr>
<tr>
<td>10032</td>
<td>1.1.12 Analyze organizational characteristics (10032)</td>
</tr>
<tr>
<td>10033</td>
<td>1.1.13 Create baselines for current processes (10033)</td>
</tr>
<tr>
<td>10034</td>
<td>1.1.14 Analyze systems and technology (10034)</td>
</tr>
<tr>
<td>10035</td>
<td>1.1.15 Analyze financial positions (10035)</td>
</tr>
<tr>
<td>10036</td>
<td>1.1.16 Identify enterprise core competencies (10036)</td>
</tr>
<tr>
<td>10037</td>
<td>1.1.17 Establish strategic vision (10037)</td>
</tr>
<tr>
<td>10038</td>
<td>1.1.18 Align stakeholders around strategic vision (10038)</td>
</tr>
<tr>
<td>10039</td>
<td>1.1.19 Communicate strategic vision to stakeholders (10039)</td>
</tr>
<tr>
<td>10040</td>
<td>1.2 Develop business strategy (10040)</td>
</tr>
<tr>
<td>10041</td>
<td>1.2.1 Develop overall mission statement (10041)</td>
</tr>
<tr>
<td>10042</td>
<td>1.2.1.1 Define current business (10042)</td>
</tr>
<tr>
<td>10043</td>
<td>1.2.1.2 Formulate mission (10043)</td>
</tr>
<tr>
<td>10044</td>
<td>1.2.1.3 Communicate mission (10044)</td>
</tr>
</tbody>
</table>

*Figure 2. Part of the APQC Process Classification Framework (APQC).*

Whether this identification of functions starts with a reference model or not, it involves many interviews with different people in the organization. These interviews serve to either identify the functions directly, or to check to which extent the functions from a reference model apply to the organization. The interviews must both be held with employees that are involved in the different cases that the organization handles and with product (and service) managers of the different products and services that the organization handles. It is therefore
important to observe that the different people involved, may use different terms for similar business functions. Especially, because at this stage in the design of the process architecture, we are interested in explicitly identifying the similarities of and differences between the functions that are performed for the different case types. Determining what is similar and what is different for different case types is often hard to do, based on terminology alone. It requires an intricate understanding of the operations of an organization. For example, what is called ‘acquisition’ in one part of the organization may be called ‘market survey’ in another. At the same time, two functions called ‘implementation’ may represent different activities: one may represent the implementation of software, while the other represents the implementation of new regulations in the organization.

In addition, functions may be organized differently. Consider, for example Figure 3, which is taken from a real-world case and shows parts of the functional decompositions of two departments from the same organization, one in Europe and one in North America. The European department distinguishes between purchasing and sales, where both purchasing and sales are split up into tactical and operational functions: sourcing and order-to-pay for purchasing and marketing and sales operations for sales. The North American department distinguishes between sourcing, marketing and order handling. Where order handling involves both order-to-pay and operational sales activities (but is not decomposed any further).

![Figure 3. Two Functional Decompositions.](image)

Clearly, in the latter case of differences between case types, a negotiation step may be required between the different people involved to unify the functional decompositions. Especially, because the functional decomposition may be more than just a modeling exercise. It may also represent real organizational properties. For example, in the case that is illustrated in Figure 3, there were managers for the different functions at the different levels of decomposition. In Europe there was a manager for sales and a manager for procurement and lower-level managers for sourcing, order-to-pay, marketing and operational sales, while in North America, there were managers for sourcing, marketing and order management. Therefore, when the functional decompositions of the departments had to be harmonized, the management structure also had to be harmonized. Note, however, that also when it comes to simple naming differences between functions, harmonization may be necessary to resolve naming differences.

A functional decomposition should not be confused with a decomposition according to case type. It is possible that an organization is structured according to both business function and other properties. It is then tempting to develop the functional decomposition further according to these other properties. However, these other properties should be reflected in the case type dimension rather than the function dimension. For example, an organization can be structured according to business function into a sales and a procurement department with managers leading both departments. It can be further structured according to location,
having both a sales and a procurement department in Europe and in North America. In this situation, the functional decomposition ends with the decomposition into sales and procurement. Should a further decomposition according to location be relevant, then this decomposition should be reflected in the case type dimension, not in the function dimension.

An important decision that must be made, when developing the functional decomposition, is determining the level of decomposition at which the functional decomposition ends. In theory, the functional decomposition can be performed up to a level that represents the tasks that are performed by the individual employee (fill-out form, check correctness of information on form, have colleague check correctness of information on form, ...). However, for a process architecture a more coarse level of decomposition is usually chosen. Two rules of thumb that can be used to choose the level of decomposition at which the functional decomposition ends, are the following.

1. The functional decomposition should at least be performed down to a level at which functions correspond to different organizational units (with corresponding managers). For example, if an organization has both a sourcing and an order-to-pay department and both have their own managers, this is a strong indication that the functional decomposition should contain the functions that are performed by these departments.

2. The functional decomposition should include different functions for the different roles in each department. For example, if the sourcing department has buyers, who do requirements analysis and vendor selection, as well as senior buyers, who do vendor relationship management and contract management, this may lead to a decision to include requirements analysis, vendor selection, vendor relationship management and contract management as functions.

Note, however, that these are just rules of thumb. They do not have to be followed strictly, but merely provide a means to help determine the lowest level of decomposition that should be used.

**Step 3: Construct one or more case/function matrices**

The first two steps lead to a matrix that has the different case types as columns and the different functions as rows. A cell in the matrix contains an 'X', if the corresponding function can be performed for the corresponding case type.

Figure 4 shows an example of a case/function matrix. The matrix shows a decomposition of case types by customer type, resulting in three case types: one for private customers, one for corporate customers and one for internal customers. The figure also shows a functional decomposition into three main functions and a subsequent decomposition of those main functions into ten sub-functions. Management and support functions are only performed for internal customers, while operational functions are performed for private and corporate customers.
A case/function matrix can be split-up into multiple matrices, when that improves the readability. We typically split-up a case/function matrix, in case a partition of the matrix’ functions and case types is possible, such that all X’s are preserved. For example, the matrix from Figure 4 can be partitioned into a matrix that contains the management and support functions and the internal customers and a matrix that contains the operational functions and the private and corporate customers.

**Step 4: Identify Processes**

In the fourth and final step, we determine which combinations of business functions and case types form a business process. To determine this, we need to find a trade-off between two extremes, one in which the entire matrix forms one big process and one in which each single cross in the matrix forms a process.

We find this trade-off, using the rule that, in principle, the entire matrix forms one big process, which we only split-up in case one of the following rules applies. We explain these rules below in more detail.

1. If a process has different flow-objects, it can be split-up vertically.
2. If the flow-object of a process changes multiplicity, the process can be split up vertically.
3. If a process changes transactional state (in the action-theoretic sense of the word), it can be split-up vertically.
4. If a process contains a logical separation in time, it can be split-up vertically.
5. If a process contains a logical separation in space, it can be split-up horizontally.
6. If a process contains a logical separation in another relevant dimension, it can be split-up horizontally.
7. If a process is split-up in a reference model, it can be split-up.
8. If a process covers (many) more functions in one case type than in another, it can be split-up horizontally.
These rules can be used as desired. However, note that they are just guidelines. Therefore, they may or may not apply to a particular organization and they are not the only rules that should be applied. Other rules can be applied as well.

Figure 5 shows the running example that we will use to explain the guidelines. The figure shows a case/function matrix for a mortgage broker, which brokers mortgages both in the Netherlands and in Belgium and distinguishes between simplex and composite mortgages. A composite mortgage can be adapted to the specific requirements of a customer, by composing it from different types of loans, savings accounts, life insurances and investment accounts. A simplex mortgage consists of a predefined package of a loan, a savings account and a life insurance. On these different types of mortgages, various business functions can be performed. Risk assessment involves assessment of risk of both individual clients, who are in the process of applying for a mortgage, and mortgage products as a whole. Mortgage brokering involves the selection of a particular mortgage package based on the requirements of a particular customer and subsequently offering that package to the customer and closing the contract. The financial functions involve paying out the mortgage and subsequently collecting the monthly payments. Finally, product development is the periodic review of the mortgage products and their components.

**Rule 1.** A flow object is an object in the organization that flows through a business process. It is the object on which business process activities are being carried out. Typically, each business process has a single flow object, such that flow objects can be used to identify business processes. Consequently, if multiple flow objects can be identified in a business process, this is a strong indication that the process should be split-up. Figure 5 illustrates this case. One flow object for the mortgage brokering process is a mortgage application on which activities are carried out during a mortgage application by a client. These activities include a risk assessment and paying out the mortgage to the client. Another flow object in the mortgage brokering process is a mortgage product on which activities are carried out periodically to assess the risk of the product as a whole and to evaluate and develop the product. Consequently, we can split up the mortgage brokering
process into two processes, one that has a mortgage application as a flow object and one that has a mortgage product as a flow object. We call the former the mortgage application process and the latter the product development and assessment process.

**Rule 2.** It is possible that in a business process a single flow object is sometimes used, while at other times multiple flow objects of the same type are used. This is typical for batch processing, in which certain activities are performed for multiple customer cases in batch at the same time. If, in the same process, the number of flow objects that is processed per activity differs, this can be a reason for splitting up the process. For example, in Figure 5 the mortgage application process is performed for a single mortgage application. However, the collection of payments happens for all mortgages in batch at the end of the month. This is a reason for splitting the process and having ‘mortgage collection’ as a separate process.

**Rule 3.** According to the action-workflow theory, a business process goes through a number of phases. In particular, we distinguish: the initiation, the negotiation, the execution and the acceptance phase (Medina-Mora, Winograd, Flores, & Flores, 1992). During the initiation phase, contact between a customer and a provider is initiated. During the negotiation phase, the customer and the provider negotiate about the terms of service or delivery of a product. During the execution phase, the provider delivers the product or service to the customer and during the acceptance phase, the customer and the provider negotiate about the acceptance and payment of the delivery. A transition in a process from one phase to another is an indication that the process can be split up. For example, in Figure 5, during the negotiation phase the mortgage broker and the customer negotiate about the selection of mortgage products, ultimately leading to a contract being signed by both parties. During the execution phase the mortgage is paid out to the customer and the monthly payments are collected. Therefore, we split up the process into a ‘mortgage application’ process and a ‘mortgage payment’ process.

**Rule 4.** A process may contain a logical separation in time. This is the case, if its parts are performed at different time intervals. Intervals that can typically be distinguished include: once per customer request, once per day, once per month and once per year. For example, in Figure 5 mortgage selection, offering and contracting is performed once per mortgage application, while mortgage payment collection is performed once per month. This is another indication that we should split up mortgage selection, offering and contracting from mortgage payment collection.

**Rule 5.** A process may also contain a logical separation in space. A process contains a logical separation in space, if it is performed at multiple locations and is performed differently at those locations. It is important to note that, while rules 1-4 lead to a separation of processes between rows (a vertical split), this rule leads to a separation of processes between columns (a horizontal split). It is also important to note that it is not sufficient for processes to just be separated in space, because many transitions within a single process from one task to another will also be associated with a transition in space. The separation must be such that there is no choice but to perform the processes differently for the different logical units. For example, in case a process is performed at different locations within the same country, there is not necessarily a reason to perform it differently at those locations. Consequently, there is no reason to split it up. In fact, organizations should strive to make their processes as uniform as possible, to benefit from economies of scale. Indeed many organizations nowadays started projects in
which they aim to make their processes more uniform across different locations, where processes became different purely for historic reasons or because the different locations did not share information about their process flow. As another example, the processes from Figure 5 are performed at two different locations in different countries. However, still not all of these processes should differ at these two locations. For example, mortgage payment and collection may be the same in Belgium and the Netherlands. However, risk assessment, mortgage brokering and product development may differ between the Netherlands and Belgium, due to country-specific rules and regulations.

**Rule 6.** A process may finally contain a logical separation in any other relevant dimension between different columns (leading to a horizontal split). Like with the separation in space, it is not sufficient for processes to just be separated. The separation must be such that there is no choice but to perform the processes differently for the different logical units.

**Rule 7.** A reference process architecture may be used to further decompose the processes that are distinguished. A reference process architecture structures a collection of processes and relations. For example, if a reference financial services exists, its structure can be used as an example or starting point to structure your own process architecture.

Figure 6 shows the results of applying guidelines 2 through to 7 to the case/function matrix from Figure 5. It shows that after applying these guidelines as illustrated above, there are 6 processes: ‘product development and assessment Netherlands (PD NL)’, ‘product development and assessment Belgium (PD BE)’, ‘mortgage application Netherlands’, ‘mortgage application Belgium’, ‘mortgage payment’ and ‘mortgage collection’.

**Rule 8.** The last guideline to apply should be guideline 8, because its result depends on the current decomposition of processes. When applying this guideline, it is necessary to look at the current decomposition of processes and check if, within a process, (many) more functions are performed for one case type than for another, i.e.: whether a process has many more crosses in one column
than in another. If so, this is a strong indication that the process should be split up
for these two case types. For example, when looking at Figure 6, we see that the
‘mortgage application Netherlands’ process, has many more function for
composite mortgages than for simplex mortgages. Therefore, we split-up this
process. This results in Figure 7, which is our final process architecture.

Figure 7. A Case/Function Matrix Evolving into a Process Architecture (Applying Guideline 6).

Bibliography


mit den R/3-Referenzprozessen*. Bonn: Addison-Wesley.

Architecture: An Overview of Approaches and their Use*. Eindhoven: BETA.

Documentair Structuurplan. (n.d.). Retrieved August 26, 2011, from
http://www.model-dsp.nl

approach to workflow management technology. *Conference on Computer
Supported Cooperative Work*, (pp. 281-288).

Van Haren Publishing B.V.