A Pattern for Global Payment Optimization via REA Ontology

Paper submitted for the REA-25 Special Track II – REA Business Patterns

REA-25 Enterprise Model Conference, June 13-15 2007 in Newark, Delaware

Dr. P.E.A. Vandenbossche (*)

University of Groningen

Faculty of Systems, Organization and Management

P.O. Box 800

9700 AV Groningen

The Netherlands

(*) Dr. Piet Vandenbossche is Assistant Professor at the University of Groningen, The Netherlands. p.e.a.vandenbossche@rug.nl, Tel.: +31 6 53.65.72.79

Abstract

Business patterns are used by information system designers to model recurring functionality in a transparent and reusable way. Recent developments in the REA ontology design have focused on reusing REA concepts to model business patterns. The REA ontology was originally described to support reusable accounting frameworks for accounting data model design. Using REA concepts also to describe business patterns is promising in order to guarantee on transparency, accuracy and functional completeness across patterns. This paper proposes a global payment optimization pattern which is designed via concepts of the REA ontology. In this pattern, transaction costs related to payments are optimized for two variables: 1. minimizing transaction costs of cross-country payments; and 2. minimizing transaction costs of currency conversion. An extension to the REA ontology is proposed to model a decision process as implementation of the REA Policy Pattern.

Keywords: REA, design pattern, cash management, payment pattern, decision process modeling

1. Introduction

Designers of business information systems use business patterns to model recurring functionality in a transparent way. Examples of domain analysis and business pattern designs are proposed by Fowler (1996) and Evans (2003) amongst others. These business patterns are often expressed in a general information modeling language such as UML, so that they are ready to be adopted in technical designs after some contextual modification. However, UML does not incorporate any specific feature related to the business domain. This hampers re-use of business patterns. This is one of the reasons why domain-specific language extensions raise currently much interest in computer science research (amongst others: Budinsky et al., 2004; Ledeczi et al., 2001; Meijler, 2004). Most business patterns thus far have been designed in isolation of each other. Little attention has been paid to coherence between patterns and transparency of concepts used between patterns. Transparency across business patterns is currently being enforced via choosing the same modeling language. It is not pursued via reusing the same business concepts over different business patterns.

Hruby (2006) has especially focused on the latter approach for modeling business patterns. He has chosen the REA ontology to redesign known structural and behavioral business patterns using one transparent business concept of Resources-Events- and Agents (REA), and business relationships between them. The REA ontology has a long heritage and was used in multiple different contexts. It was originally defined as generalized accounting framework to define historic accounting data in a shared data environment for a single company (McCarthy, 1982). Afterwards, it was elaborated towards the extended REA model and the REA ontology, providing a series of new conceptual enhancements such as e.g. definition of future data, resource conversions, commitments and schedules, etc. (Geerts and McCarthy, 2000, 2002). The REA concept was also extended to be used as taxonomy for data exchange between companies in a supply chain B2B context (Haugen and McCarthy, 2000). Additional patterns could be designed using the REA ontology, thus extending the body of knowledge on REA-based patterns. New patterns could lead to an extension of the REA ontology when certain design aspects cannot be expressed via current REA concepts.

This paper focuses on the design of a pattern for global payment optimization using concepts of the REA ontology. This pattern is currently not yet described using the REA ontology. It focuses specifically on the problem of payments in an international context where suppliers and customers reside in many different countries. The organization effectuating the payment uses different currencies and operates via network of available bank accounts in different countries. Payment traffic incurs bank transaction costs when it is effectuated between business partners residing in different countries, and when currency conversions are involved. The optimization of payments will focus on these two aspects. In smaller organizations, the decision which bank account to use for payment is often made manually. For modern ERP systems, it is expected to support this trade-off automatically via the system. Especially for the design of these systems, the pattern for global payment optimization is useful.

The following methodology has been applied to design the pattern. The design takes into consideration all REA concepts which are currently available and defined in the REA ontology. Other business patterns which are designed using concepts of REA have been evaluated and used for the design of the global payment optimization pattern when relevant. When certain required design features are not yet provided for as REA-concept, extensions to the REA ontology are proposed. Unlike features of other available business patterns designed via concepts of the REA ontology, the payment optimization pattern is characterized by a decision process where trade-offs have to be decided upon at various decision points dependent on the value of a previous activity. Decision policies can already be defined via the REA Policy Pattern as static diagram, but this pattern does not express the dynamic behavior which characterized the decision process. An extension of the REA ontology to design a dynamic decision process will be proposed, and will be illustrated afterwards for the global payment optimization process.

The paper is structured as followed. First, Section 2 proposes the payment optimization process as scope of the pattern to be designed. Next, in Section 3, an extension to the REA ontology will be presented to model the dynamic behavior of a decision process via REA concepts. Afterwards, in Section 4, the pattern for global payment optimization pattern will be described. Finally, some conclusions will be presented in Section 5.

2. A global optimization pattern for payments in a company operating in a multi-currency and multi-country environment

This section outlines the context and the scope definition of the global optimization pattern for payments. Organizations pursue optimization of their business processes to achieve efficiency goals. Efficiency gains can be achieved via optimization of two types of processes: the primary processes and the support processes (of which financial processes are an example) (Porter, 1980). Most gains can achieved via optimization of the primary processes – the processes which relate to the creation, inventory, physical distribution and delivery of finished goods and services (i.e. the operational resources). Financial business processes focus on the efficient use of various types of financial resources, such as e.g. the available cash on a bank account, the portfolio of loans and securities used, etc. Optimization of financial business processes is often forgotten or only partially pursued. However, further optimization is also possible here. All possible financial resource flows (inflow, outflow, surplus investment, deficit financing) can be optimized in an integral context or independently of each other (Vandenbossche, 2005). The global payment optimization pattern which is subject of this paper is a specific example of the cash outflow process, investigated in isolation of other financial resource flows. Examples include: payments effectuated to suppliers to settle purchase invoices, periodic payment of interest, loan redemption, etc. Global organizations often have business divisions as well as business partners in many *different countries* which can be operating in *different currencies*. The impact of both variables, specific to the optimization of payments, will be explained next.

First, the impact of money flows in different currencies is being discussed. Financial resource flows such as payments and collections are effectuated in multiple different currencies as a result of requirements and/or preferences of local business partners (often customers and/or suppliers). Global organizations try to minimize the costs related to operating in different currencies by setting up a network of bank accounts by currency. It is beneficial to minimize the number of currency conversions as they incur currency conversions costs, to be paid into the bank. They therefore lead all cash inflows and outflows in a certain currency to the same bank account (which has been specifically created to handle transactions in a certain currency). This way, the number of currency conversions (and related costs) is minimized.

Second, the effect of money flows which are effectuated between two different countries is subject of further investigation. Another source of banking costs relates to the transactions costs when paying to a bank of a business partner, which is located abroad. Smaller local business partners (such as customers) sometimes do not accept the cost of paying abroad. A well thought-out network of globally located banks is the practical solution to this problem. Global organizations avoid the bank costs of cross-country payments by opening up a bank account in a certain country (and in a certain currency) when the volume of transactions (to be collected and/or to be paid) is expected to be high.

A special situation is the so-called ‘common currency zone’, which is a territory consisting of multiple countries which are using the same currency and for which no cross-country payment costs exist. The ‘Single Euro Payment Area (SEPA)’ (i.e. one ‘space for payments’ in Europe, in which payments, credit transfers, direct debits and card payments in Euro can all be processes in the same way, both for domestic payments as well as cross country payments) is an example of a common currency zone. Common currency zones drastically reduced the need for currency conversions and cross-country payments.

For a global organization which is operating in a multi-site environment where several operational units make use of the central network of bank accounts to handle all possible cash flow transactions (inflows, outflows, deficit financing, surplus investment etc.), this is a complex optimization problem. In Figure 1, a simplified situation has been adopted where all possible payment transactions in a global network of bank accounts are depicted.

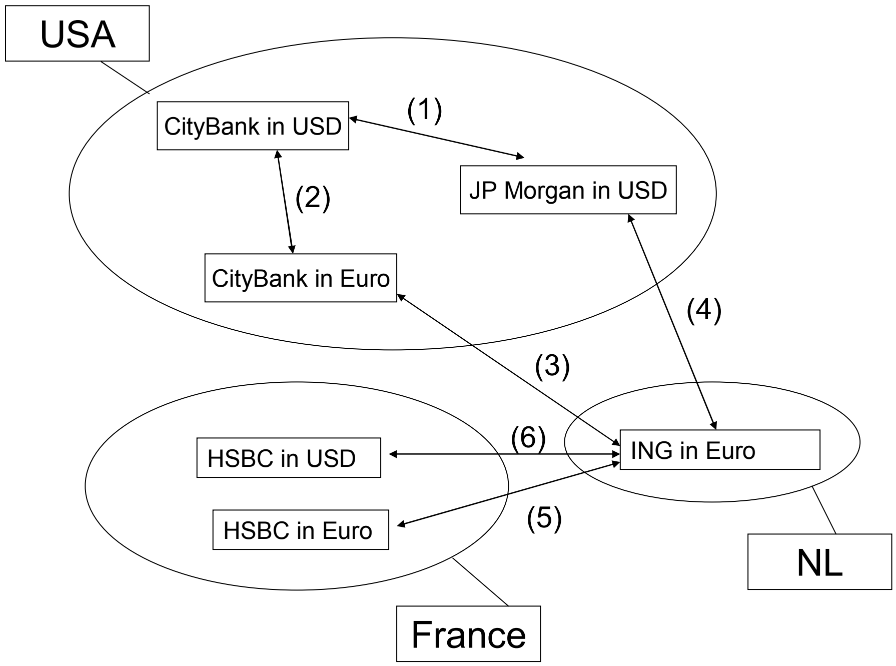


Figure 1: Possible payment transaction flows inside and between countries

In Figure 1, the SEPA Zone (as explained above, here represented by countries 'France' and 'The Netherlands') is an example of a common currency zone. The transaction costs involved in each of the payment transaction types is outlined in Table 1.

Table 1: Transaction costs by type of payment transaction

Flow #	Same Country	Same Currency	Common Currency Zone	Currency Conversion Cost	Cross-country Transaction Cost
(1)	X	X	N	N	N
(2)	X		N	Y	N
(3)		X	N	N	Y
(4)			N	Y	Y
(5)		X	Y	N	N
(6)			Y	Y	N

Table 1 illustrates that transaction costs involved for payment between countries in a common currency zone (payment types (5) and (6)) are the same as transaction costs within one single country (payment types (1) and (2)).

Next, it will be discussed how the pattern of a dynamic decision process can be added to the REA ontology. This is a required additional REA concept to be able to model the global payment optimization pattern via the REA ontology as will be discussed in Section 4.

3. The need for modeling a dynamic decision process in the REA ontology

The objective of the global payment optimization pattern is to execute any payment transaction at the lowest possible transaction cost. In order to achieve this goal, each payment transaction goes through a decision algorithm which determines the choice from which bank account to pay. A bank account is characterized by a certain location (most significant for this purpose is the country where the bank account is maintained) and a certain currency.

From a generic perspective, a dynamic decision process consists of a flow of activities where the outcome of trade-offs of a previous activity (based on certain decision criteria) determine which next activity has to be executed (out of a range of one or more alternative possibilities). The sequence of activities to be executed in a decision process is not fixed, but depends on the outcome of the trade-offs which are being made during the execution of the decision algorithm. In case of the global payment optimization pattern, these trade-offs relate to the question: 'which bank account has sufficient available (unallocated) balance so that it can be used for the payment'. This evaluation has to be made in a dynamic situation where available amounts on each bank account change on a daily basis. Hence, payments can be effectuated making use of a different set of bank accounts each time again. The scope of the global payment optimization pattern is visualized in Figure 2.

= = = = = = = = = = =

Insert Figure 2 here

= = = = = = = = = = =

A fundamental requirement to model the global payment optimization pattern lies in the need for additional REA concepts to model the dynamics of a decision process. The REA ontology currently only focuses on static diagrams for modeling REA Structures at Operational Level (i.e. REA Exchange process, REA Conversion Process and REA Value Chain) and REA Structures at Policy Level (e.g. REA Commitments, REA Schedule, REA Contracts, amongst others) (Hruby, 2006, also based on McCarthy, 1982; Geerts and McCarthy, 2000,2002). The REA ontology does not contain concepts to model the dynamic aspect of a decision process yet. Various complex business patterns have already been expressed via concepts of the REA ontology. Examples include: 'Sales process with return of products', 'sales process with shipment', 'tax with purchase and sales', etc. (Hruby, 2006). The common characteristic between all these patterns is the fact that two or more activities take place in a predefined sequential order. No trade-offs or decision moments are modeled in each of these patterns. These patterns are expressed as static diagrams.

Hruby (2006) proposes a 'REA Policy Pattern'. He describes a policy as '*A rule or practice or procedure to guide decisions and actions'* . Rules in this context have the functional meaning of a 'company policy' on what norms to achieve and how to make certain decisions. The REA Policy Pattern leads to a static diagram, not meant to express the dynamic decision behavior. The relationship between the REA Policy Pattern and the dynamic REA Decision pattern which will be proposed here after, can be explained as follows. The REA Policy Pattern could set the boundaries of any decision process whereas the REA Decision Pattern holds the dynamic behavior of the actual decision flow. This means the following, when applied to the dynamic decision process in the global optimization pattern for payments as explained later in this paper, the REA Policy Pattern could set the boundaries on which transaction costs are accepted and which bank account to use in a certain payment scenario. As such, the REA Policy Pattern can be the input of the payment decision process to be executed by the Global Payment Optimization Pattern. The *implementation* of the REA Policy Pattern (applied to the execution of global payments) is further detailed in the Global Payment Optimization Pattern.

Some modeling languages such as UML support the modeling of dynamic decision behavior, e.g. via the UML Activity Diagram and the UML Sequence Diagram (Fowler, 1996). The dynamic decision process of the global payment optimization pattern is modeled via an UML Activity Diagram and is visualized in Appendix A. The REA ontology needs a conceptual enhancement which allows to model a dynamic change of the activity flow in a decision process, similar to the approach offered in the UML Activity diagram.

The following different patterns of a decision process can be added to the REA ontology:

A. Basic decision process: REA Decision 'OR' Pattern

The basic decision process is the pattern which allows to model a conditional clause with only two possible outcomes: either the condition is valid or the condition is invalid. The pattern of the proposed basic REA Decision 'OR' Pattern is visualized in Figures 3a and 3b. In the Figure 3a, the 'or' relationship is expressed explicitly. Since most of the basic decision processes between two conditions have an 'or' relationship, it is proposed to assume this as default and leave out the expression as visualized in Figure 3b.

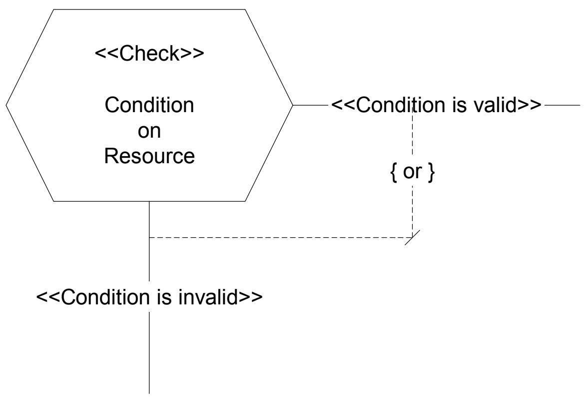


Figure 3a: REA Decision 'OR' Pattern (explicit expression)

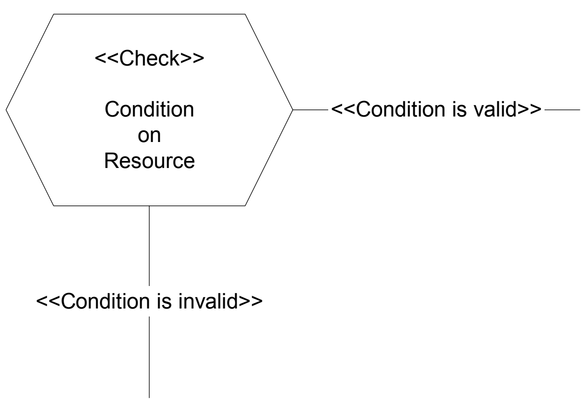


Figure 3a: REA Decision 'Or' Pattern (default expression)

B. Basic decision process: REA Decision ‘AND’ Pattern

The second type of a basic decision process is the pattern which allows to model the situation where two conditional clauses are both applicable at the same time. The proposed REA Decion ‘AND’ pattern is visualized in Figure 4.

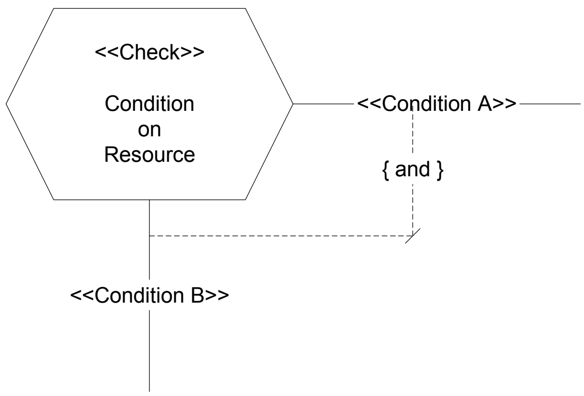


Figure 4: REA Decision ‘AND’ Pattern

C. Basic Decision Process: REA Decision ‘AND/OR’ Pattern

The third variant of the basic decision process relates to the situation when the decision consist of two expressions where either only one of both expressions; or alternatively; both expressions together need to occur at the same time to meet the decision criterion. This variant is called the REA Decision ‘AND/OR’ Pattern, and is visualized in Figure 5.

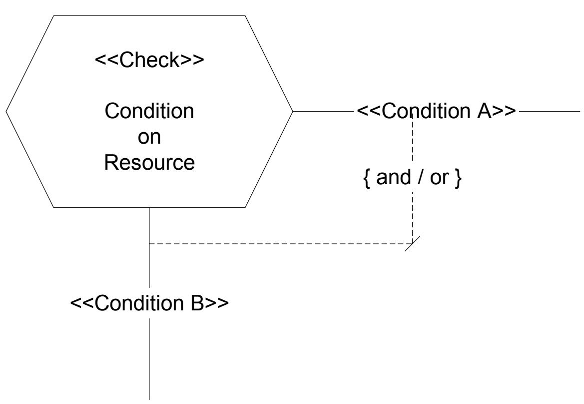


Figure 5: REA Decision ‘AND/OR’ Pattern

D. Decision process with multiple alternative outcomes: REA Decision 'CASE' Pattern

The more complex pattern of a decision process is the decision pattern which processes a condition which has more than two possible alternative outcomes. Dependent of the value of the outcome of the condition, one of the alternative options will be further executed. This proposed pattern is called the REA Decision 'CASE' Pattern, and is visualized in Figure 6.

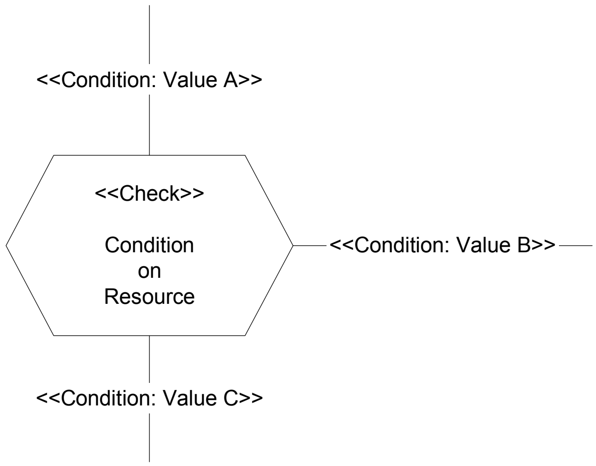


Figure 6: The REA Decision 'CASE' Pattern

Designing the flow of dynamic behavior of a decision process has the objective to enforce the system to proceed in a certain predefined way depending on decision criteria. This is something different compared to human choices which are normally made during routine deployment. In recent REA ontology literature, it is illustrated and explained how REA concepts could work in actual implementations (Hruby, 2006). It is e.g. illustrated that at run time, different resources could be available whereas only one is supposed to be chosen; or at run time, many agents exist, where each could have different economic interests whereas only one agent with one particular interest is relevant in a certain application context, etc. The fact that actual deployments (i.e. instances of REA concepts) might consist of several instances of REA components with an 'assumed selection of one of the instances' is logical and in line with expectations for any implementation. In these latter examples, taken from Hruby, 2006, one cannot speak of predefined dynamic decision process. However, when more complex decision logic comes into play, additional constructs are needed, such as proposed in Figures 3 to 6.

4. A REA pattern for Global Payment Optimizations

This section describes how to express the global payment optimization pattern via proposed extension of REA concepts. The scope of the global payment optimization pattern is visualized in Figure 2 and Appendix A. The starting situation for using the global payment pattern is a series of one or more payment lines which are ready to be effectuated. It is assumed that previous activities such as e.g.: approved purchase invoice submitted for payment, approval of the payment line, etc. are already successfully finished, and are as such out of scope for this pattern. The entire application (of which the dynamic decision process on payment optimization is only a part) can be described as a series of integrated proposed REA concepts as visualized in Figure 7.

= = = = = = = = = =

Insert Figure 7 here

= = = = = = = = = =

The global payment optimization pattern as proposed REA business pattern is presented in Figure 7. Appendix A describes the individual steps of the payment decision flow. The payment decision flow can be modeled as a sequence of consecutive basic Decision 'OR' Patterns as proposed REA extension in the previous section. A different course of action takes place in the situation where sufficient balance is available on a bank account (in a given country and a given currency) compared to the situation where the required balance is not available. The following consecutive decisions are evaluated, each via using a basic Decision Pattern as REA extension:

1. Is cash available on a bank account in the same country and in the same currency ?
2. Is cash available on a bank account in the same country but in a different currency ?
3. Is cash available on a bank account in a common currency zone ?
    a. If yes – Is cash available in the same currency ?
4. Is cash available on a bank account in a country which is not within a common currency zone ?
    a. If yes – Is cash available in the same currency ?

In the situation where cash resources are available but in a different currency, the REA Conversion pattern (see Hruby, 2006) is being applied to convert cash resources from one currency into another currency.

As explained earlier in this paper, there is a difference in scope and objective between the REA Policy Pattern (which is a static, structural diagram, outlining the boundaries of a decision process) and any example of a Decision Pattern as proposed REA extension, here the Global Payment Optimization Pattern (which is a diagram specifying a dynamic decision flow). Because of the difference in nature of the diagram (static versus dynamic), the REA Global Payment Policy Pattern is not part of the REA Global Payment Optimization Pattern.

5. Conclusion

Business patterns are more transparent if they are not designed in isolation of each other, but reuse the same set of business concepts. Examples are e.g. the business patterns which are designed based upon REA concepts of the REA ontology. In this paper, we have introduced a dynamic decision pattern as addition to the REA Ontology, and have illustrated the use via a global payment optimization pattern which can be considered as an addition to the body of knowledge of REA-based business patterns. The recent REA development towards use for business pattern modeling can be seen as an evolution towards a definition of a domain-specific modeling language from a computer science perspective.

REA concepts have a long heritage and have been used to achieve multiple varying goals. The REA ontology was only recently also used to design business patterns. REA-based business patterns are currently only designed as static diagrams. In principle, it not possible to express the flow of activities which dynamically varies dependent on specific situations (e.g. as an outcome of: context variables, decision outcome, et cetera.). Some business patterns do not focus on dynamic behavior. For this type of patterns, the static diagram is sufficient, and the gap is irrelevant. But some other business patterns focus primarily on the expression of dynamic behavior whereas the equivalent static diagram is far less important. This is the case for modeling e.g.: complex heuristics, dynamic decision processes where decision flows are based upon context variables, sorting algorithms, etc. In line with other modeling languages such as UML, besides static domain and pattern analysis design diagrams, there is a whole additional range of supporting diagrams – each serving a specific purposes. Examples include: the UML activity diagram, the UML Sequencing diagram, the UML Use Case Diagram, etc. In order for the REA ontology to grow towards a mature business pattern modeling language, additional research could focus on building out the required modeling environment, which is required to achieve these goals.

Literature

Budinsky F., Steinberg D., Merks E., Ellersick R. and T.J. Grose (2004). *Eclipse Modeling Framework.* Addison Wesley.

Evans, E. (2003). *Domain driven design: Tackling complexity in the heart of the software*, Addison Wesley-Professional.

Fowler, M. (1996). *Analysis Patterns: Reusable Object Models*, Addison-Wesley Professional.

Geerts, G.L. and W.E. McCarthy (2000). *The Ontological Foundation of REA Enterprise Information Systems.* Paper presented at the American Accounting Association Conference, Florida, 2000.

Geerts, G.L. and W.E. McCarthy (2002). An ontological analysis of the economic primitive of the extended REA enterprise architecture, International Journal of Accounting Information Systems, 3, pp. 1-16.

Haugen, R. and W.E. McCarthy (2000). REA: A Semantic Model for Supply Chain Collaboration. Paper presented at the OOPSLA conference in 2000, Minneapolis.

Hruby, P. (2006). *Model driven design using Business Patterns*. Springer-Verlag, Berlin, pp. 357.

Ledeczi A., Nordstrom G., Karsai, G., Volgysei P. and M. Marioti (2001). *The Generic Modeling Environment.* In Proceedings of WISP, 2001.

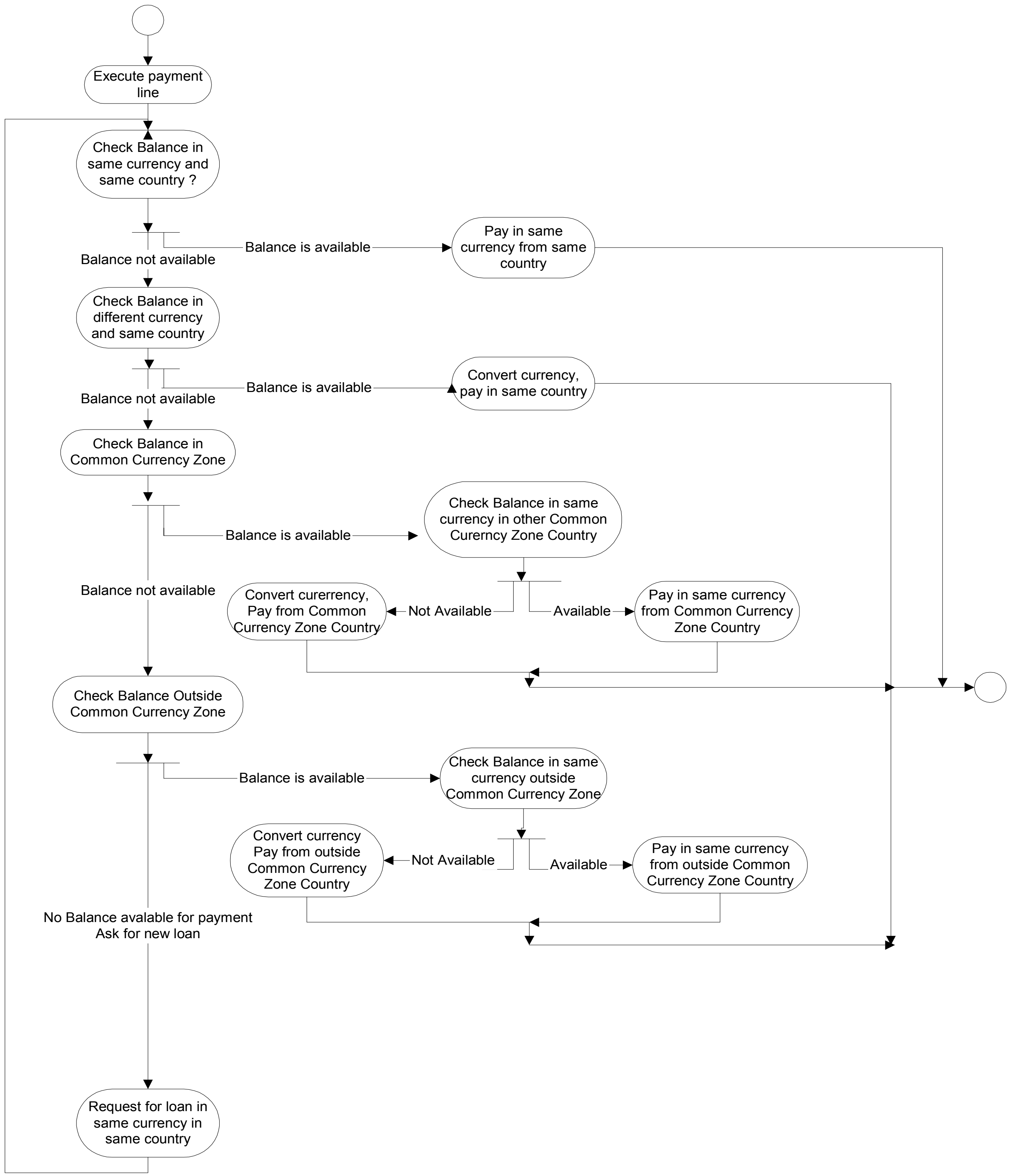
McCarthy, W.E. (1982). The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment. The Accounting Review (July 1982), pp. 554-78.

Meijler, T.D. (2004). *Incremental MDA through Causal Connectedness.* UML Modeling languages and Applications: UML 2004 Satellite Activities Lisbon, Portugal, October 11-15, 2005. Revised Selected Papers LNCS 3297 Eds. Nuno Jardim Nunes, Bran Selic, Alberto Siva, Ambrosio Toval

Porter, M.(1980). *Competitive Strategy: Techniques for analyzing Industries and Competitors.* Free Press / Simon & Schuster, Inc. NY.

Vandenbossche, P.E.A. (2005). Accounting information for changing business needs. Concepts of business logistics applied to Treasury Management. Ph.D. Thesis University of Groningen.

Appendix 1: UML Activity Diagram of the Global Payment Optimization Pattern



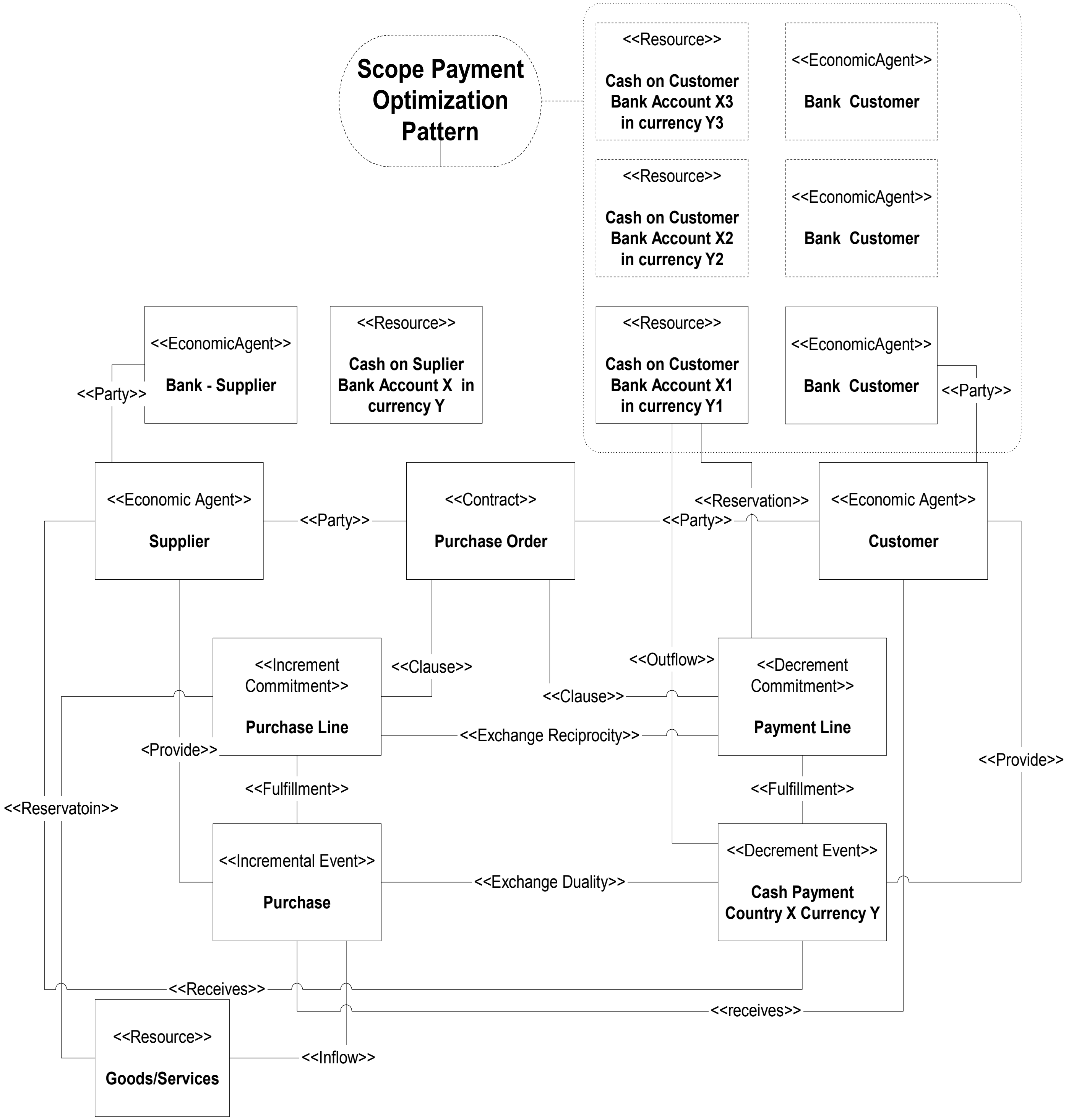
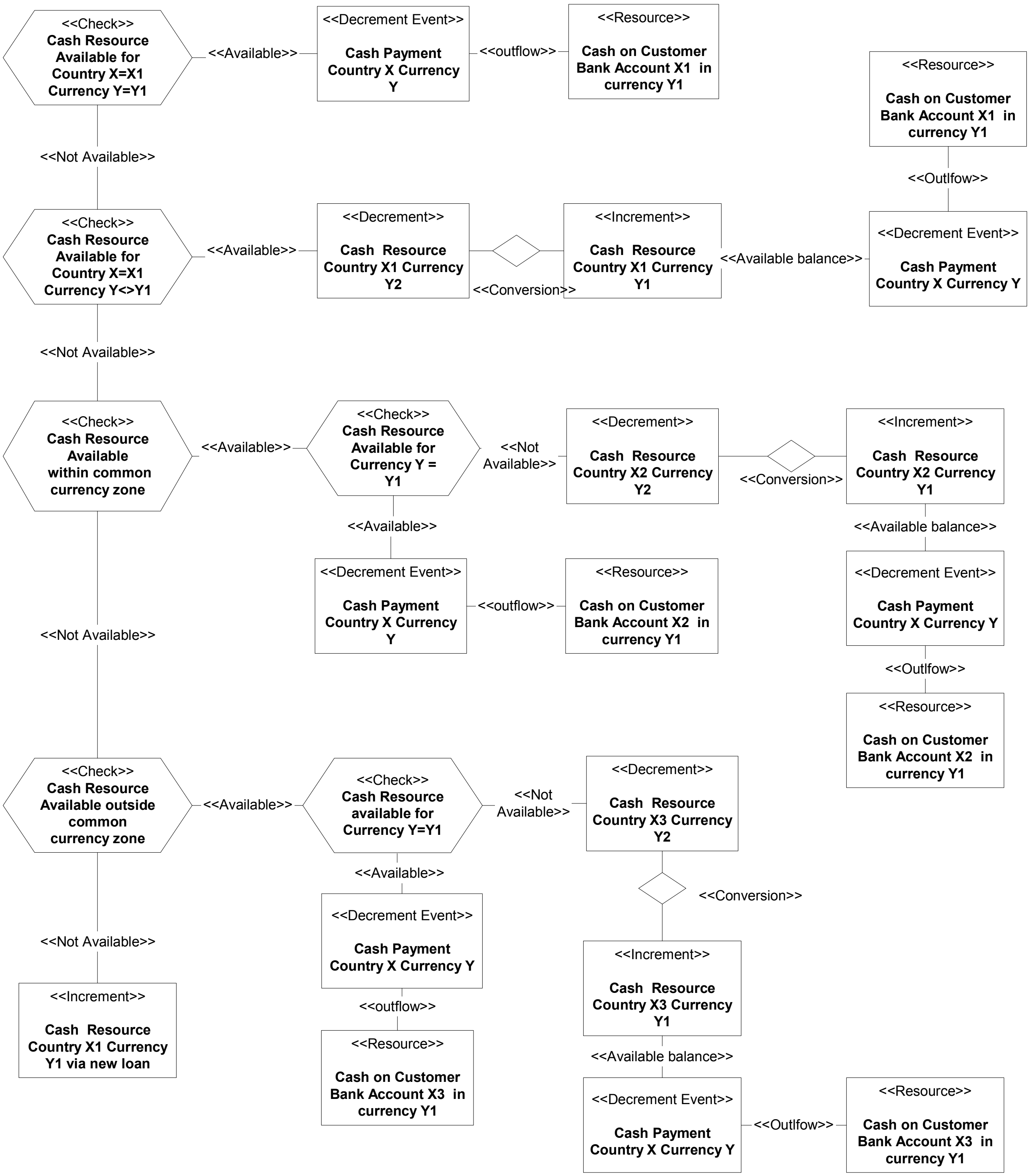


Figure 2: Scope Global Payment Optimization Pattern in REA (base pattern of a Purchase Process extended from McCarthy, 1982, Hruby, 2006)



X1 = Payment-from country same as payment-to country
X2 = Payment-from country and payment-to country in common currency zone
X3 = Payment-from and payment-to country not both in common currency zone
Y1 = currency to be paid
Y2 = any currency different than Y1

Figure 7: Global Payment Optimization Pattern