Supplementary material to:

An integrative modelling technique bridging the gap between business and information systems development

P. Joubert[a], C. de Villiers[a], J. H. Kroeze[b]
[a] Department of Informatics, University of Pretoria, South Africa
[b] School of Computing, University of South Africa, South Africa

Table SM1 An overview of modelling techniques studied

Techniques	Source
Holistic techniques: The soft system methodology (SSM) was used to represent the holistic approach, emphasising techniques such as rich pictures, root definitions and conceptual models.	The article by Checkland where he takes a thirty-year retrospective view of SSM [1].
Data techniques: Entity modelling and structured query language (SQL) were used to represent the data approach to ISD.	• The seminal article by Chen explaining entity relationship modelling for the first time [2]–[4]. • *Microsoft SQL Server Books Online* (notes on SQL) (https://msdn.microsoft.com/en-us/library/ms369860.aspx).
Business process techniques: The techniques to represent processes are many and can be categorised as follows: • **Functional modelling:** IDEF0 and IDEF3 • **Information modelling:** DFD, IDEF1 and IDEF1x • **Dynamic modelling:** IDEF2, Petri-Nets, role activity diagram (RAD), agent relationship modelling (ARM) and agent/object life cycles (ALCs/OLCs) • **Integrated modelling:** BPMN	• Certain techniques that fall under the process approach, like action diagrams and entity life cycles, were studied via their object-oriented counterparts. • For functional modelling, the IDEF0 standard was used [5], as well as other sources [6]. • Information modelling was handled under data modelling [2]. • Dynamic modelling was represented by role activity diagrams (RAD), as in Bal [7], agent relationship modelling (ARM), as in [8], agent/object life cycles (ALCs/OLCs) as in [8] , and agent-object relationship (AOR) modelling as in [9]. • Integrated modelling was represented by BPMN as in OMG [10].
Object-oriented techniques: The following techniques, as specified in UML, and grouped per view, were considered:	The UML 2 standard and other sources [11]–[13].

Techniques	Source
• The use case view of use case diagrams, and use case narratives • The static view of class diagrams and object diagrams • The dynamic view of sequence diagrams, collaboration diagrams and activity diagrams • The implementation view of component diagrams and deployment diagrams	
Project management techniques: Various project estimation techniques, PERT chart, Gantt chart and critical path method (CPM) techniques representing the project management approach.	PMBOK notes on PERT, Gantt and CPM techniques [14].[1]
Organisational techniques: Lateral thinking, critical success factors, scenario planning, future analysis, SWOT analysis, case-based reasoning, risk analysis.	Not studied because these techniques do not describe specific ontological objects but rather processes.
People techniques: Stakeholder analysis, joint application design (JAD), joint requirements planning (JRP).	Not studied because these techniques do not describe specific ontological objects but rather processes.
Enterprise architecture techniques: The main approach is the Zachman framework. The open distributed processing (ODP) standards were also used.	• The analysis of the Zachman framework for enterprise architecture from the GERAM perspective by [15]. • The open distributed processing (ODP) standards [16].
Process logic description techniques: Techniques describing process steps at a lower level of detail than processes including decision trees, decision tables, structured English, structure diagrams, Warnier-	Process logic description techniques were represented by action diagrams under OO techniques (see row 4 above) because action diagrams encompass all the other diagrams

[1] Various versions of PMBOK appeared after 1996 (see https://www.4pmti.com/PMBOK-6th-ed-changes.aspx). Since PMBOK was not used during the grounded approach to identify the base elements and their relationships (see Table SM2 and SM3), these were not discussed in this article.

Techniques	Source
Orr diagrams, Jackson diagrams, action diagrams, entity life cycles, state-dependency diagrams and various matrices like the create, read, update and delete (CRUD) matrix.	plus concurrency not covered by any of them.
Linguistic techniques: Language action perspective (LAP).	The overview of language action perspective (LAP) in [17].

Table SM2 A comparison of the base entity "actor" in the proposed integrative modelling technique and of various related constructs in existing business and ISD modelling techniques

Technique	Technique main construct	Corresponding integrative modelling technique construct
ALC/OLC	Agent	Actor
AOR	Internal/external agent	Actor
BPMN	Participant, pool, swim lane	Actor
DFD	External entity	Actor
Gant and PERT	Mechanism	Actor
IDEF0	Entity	Actor
LAP	Actors	Actor
	Agent	Actor
SSM	Actor	Actor
Zachman	Who	Actor

Table SM3 Comparison between the constructs in existing ISD modelling techniques and the proposed integrative modelling technique

(Compare Table SM1 for references to the various existing techniques)

Technique	Technique main construct	Corresponding integrative modelling technique construct
ALC/OLC	Agent	Actor
	Organisational agent	Institutional actor
	Agent life cycle	Action
	Responsibility	Relationship (between agents)
	Operations	Action
	Object	Base entity
	Object state	Base entity (state)
	Process and behavioural perspectives	Model view
AOR	Biological/human agent	Human actor
	Institutional agent	Institutional actor
	Actor actions	Action (actor)
	Actor relationships	Relationship (actor)
	Commitments, duties and rights	Relationship (between actors)
	Generalisation, composition	Relationship (type)
	Claims	Event (future)
	Internal/external agent	Actor
	Objects/entities	Object
	Entity properties/attributes	Object (properties)
	Entity type	Object (types)
	Social interaction process	Relationships (between actors)
	Non-social interaction process	Relationships (between actors and objects/artificial actors)
	Reaction rule	Action
ARM	Agency	Institutional actor
	Complex (macro) agent	Institutional actor
	Primitive agent	Human actor
	Responsibility	Relationship (between actors)
	Physical objects	Physical object, informational object, artificial actor, human actor
	Logical objects	Conceptual object
	Structural perspective	Model view
	Contractual relationships	Relationships (between actors)
	Functional and ownership relationships	Relationships (between actors and objects)
	Object	Object

Technique	Technique main construct	Corresponding integrative modelling technique construct
	Subject	Subject
BPMN	Business entities	Institutional actor
	Participant, pool, swim lane	Actor
	Data object	Informational object
	Text or graphical information	Conceptual object
	Business process, activity, subprocess, tasks	Action, action step
	Event	Event
	Event type	Event (type)
	Human level and machine level views	Model view
	Flow, gateways	Action relationships
DFD	External entity	Actor
	Data store	Informational object
	Information	Conceptual object
	Material resources	Physical object, artificial actor
	Process	Action
	Input, output	Conceptual object
	Flow	Action relationships
	Types of models: current physical, current logical, required logical, required physical	Model view
	Context	Model view
Gantt and PERT	Resource	Base entity
	Project	Action
	Perspective, view	Model view
	Work breakdown structure	Action relationships
	Predecessors	Action relationships
	Mechanism	Actor
IDEF0	Nouns or noun phrases	Base entity
	Software	Artificial actor
	Equipment, machines	Physical object, artificial actor
	Product	Object
	Raw material	Physical object
	Systems	Artificial object
	Function	Action
	Input, output	Object, model block
	Control	Action, object or actor
	Functional and context view	Model view
	Entity	Actor
IDEF1	Dictionary	Informational object

Technique	Technique main construct	Corresponding integrative modelling technique construct
	Physical entity	Physical object, informational object, artificial actor, human actor
	Abstract entity	Conceptual object
	Entity class	Base entity
	Key	Object (property)
	Information view	Model view
	As-is, to-be	Model view (type)
	Relationship	Relationship
	Things	Base entity
IDEF1X	Relationships	Relationships
	Synonyms, aliases, non-standard names.	Base entity (property)
	Primary key, foreign key, candidate key, alternate key	Base entity (property)
	Domain	Base entity (property)
	Verbs	Actions, relationships
	Semantic view	Model view
	Relationship	Relationship
	Object	Base entity
IDEF3	Noun or noun phrases	Base entity
	State condition types	Base entity state (types)
	Facts, constraints	Relationships, action steps
	Process, units of behaviour	Action
	Scenario	Action
	Objective view	Model view
	Link, junctions	Action relationship
	Kind and term	Any entity
IDEF5	Essential, accidental or defining properties	Any entity (property)
	Ontology, taxonomy	Conceptual object
	Vocabulary, terminology	Informational object
	Process	Action
	Relations	Relationship
	Organisation	Institutional actor
LAP	Actors	Actor
	Actor cycle	Action
	Agenda	Action
	Actor role	Action
	Authorisation/delegation/ propagation	Relationship (between actors)
	Production, coordination and communication acts	Action

Technique	Technique main construct	Corresponding integrative modelling technique construct
	Transaction	Action
	Initiator/customer	Actor (type)
	Fact	Base entity, action step, relationship
	Event	Event
	Atomic, fibre and molecular layers	Model view
	Semiotic layers	Model view
	Action rules	Action
	Agent	Actor
ODP	Agent role	Action
	Structuring rules (obligation, permission, prohibition)	Actions step or relationship
	Artefact	Object, artificial actor
	Artefact role	Action
	Service	Action
	Node	Place, artificial actor
	Enterprise, information, computational, engineering and technology perspectives/ viewpoints	Model view
	Role	Action
RAD	State	Base entity (property)
	Activities	Actions
	Trigger	Event
	Dynamics view	Model view
	Sequence	Action relationship
	Permission	Relationship
SQL	Database action	Action
	Database object, schema, table, virtual table,	Informational object
	Constraints	Action step, relationship
	Data type	Informational object (property)
	Action, transaction	Action
	Trigger	Action
	Database view	Model view
	Control-of-flow	Action relationships

Technique	Technique main construct	Corresponding integrative modelling technique construct
	Customer/client/beneficiary/ victim/owner	Actor (type)
SSM	Actor	Actor
	Transformation	Action
	System/subsystem	Agent
	Soft systems view	Model view
UML	Object/class	Base entity
	Attribute	Base entity (property)
	Interface	Informational object + action
	Package	Informational object
	Software component	Artificial actor (component)
	Node	Artificial actor, place
	Use case	Action
	Activity, operation	Action
	Static, dynamic, functional and implementation view	Model view
	Association	Relationship
	Decision	Action
Zachman	Who	Actor
	What	Object
	How	Action
	Where	Place
	Primitives vs. composites	Relationships
	Planner, owner, designer, builder, subcontractor perspectives	Model view

Table SM4 EU-RENT CASE STUDY

EU-RENT CASE STUDY

The case study below is quoted mostly verbatim from Hay and Healy [18, pD.1-D.8], but numbering has been added. All business rules are numbered consistently with two decimals, while all headings and subheadings are numbered with no or one decimal.

1. EU-RENT car rentals

1.1.1 EU-Rent is a car rental company owned by EU-Corporation. It is one of three businesses – the other two being hotels and an airline – that each has its own business and IT systems, but with a shared customer base.

1.1.2 Many of the car rental customers also fly with EU-Fly and stay at EU-Stay hotels.

2. EU-RENT business

2.1.1 EU-Rent has 1 000 branches in towns in several countries. At each branch, cars, classified by car group, are available for rental. Each branch has a manager and booking clerks who handle rentals.

2.2 Rentals

2.2.1 Most rentals are by advance reservation; the rental period and the car group are specified at the time of reservation. EU-Rent will also accept immediate ("walk-in") rentals if cars are available.

2.2.2 At the end of each day, cars are assigned to reservations for the following day. If more cars have been requested than are available in a group at a branch, the branch manager may ask other branches if they have cars they can transfer to him/her.

2.3 Returns

2.3.1 Cars rented from one branch of EU-Rent may be returned to a different branch. The renting branch must ensure that the car has been returned to some branch at the end of the rental period. If a car is returned to a branch other than the one that rented it, ownership of the car is assigned to the new branch.

2.4 Customers

2.4.1 A customer can have several reservations, but only one car rented at a time. EU-Rent keeps records of customers, their rentals and bad experiences, such as late returns, problems with payment and damage to cars. This information is used to decide whether to approve a rental.

3. EU-RENT BUSINESS RULES

3.1 External constraints

3.1.1 Each driver authorised to drive the car during a rental must have a valid driver's licence.

3.1.2 Each driver authorised to drive the car during a rental must be insured to the level required by the law of each country that may be visited during the period of rental.

3.1.3 Rented cars must meet local legal requirements for mechanical conditions and emissions for each country that may be visited during the period of rental.

3.1.4 Local tax must be collected (at the drop-off location) on the rental charge.

3.2 Rental reservation acceptance

3.2.1 If a rental request does not specify a particular car group or model, the default is group A (the lowest-cost group).

3.2.2 Reservations may be accepted only up to the capacity of the pickup branch on the pickup day.

3.2.3 If the customer requesting the rental has been blacklisted, the rental must be refused.

3.2.4 A customer may have multiple future reservations, but may have only one car at any time.

3.3 Car allocation for advance reservations

3.3.1 At the end of each working day, cars are allocated to rental requests due for pickup the following working day. The basic rules are applied by a branch:

3.3.1.1 Only cars that are physically present in EU-Rent branches may be assigned.

3.3.1.2 If a specific model has been requested, a car of that model should be assigned if one is available. Otherwise, a car in the same group as the requested model should be assigned.

3.3.1.3 If no specific model has been requested, any car in the requested group may be assigned.

3.3.1.4 The end date of the rental must be before any scheduled booking of the assigned car for maintenance or transfer.

3.3.1.5 After all assignments in a group have been made, 10% of the group quota for the branch (or all the remaining cars in the group, whichever number is lower) must be reserved for the next day's walk-in rentals. Surplus capacity may be used for upgrades.

3.3.1.6 If there are not sufficient cars in a group to meet demand, a free one-group upgrade may be given (i.e., a car of the next

higher group may be assigned at the same rental rate) if there is capacity.

3.3.1.7 Customers in the loyalty incentive scheme have priority for free upgrades.

3.4 Walk-in rentals

3.4.1 The end date of the rental must be before any scheduled booking of the assigned car for maintenance or transfer.

3.4.2 If there are several available cars of the model or group requested, the one with the lowest mileage should be allocated.

3.5 Handover

3.5.1 Each driver authorised to drive the car during a rental must be over 25 and must have held a driver's licence for at least one year.

3.5.2 The credit card used to guarantee a rental must belong to one of the authorised drivers, and this driver must sign the rental contract. Other drivers must sign an "additional drivers' authorisation" form.

3.5.3 The driver who signs the rental agreement must not currently have a EU-Rent car on rental.

3.5.4 Before releasing the car, a credit reservation equivalent to the estimated rental cost must be made against the guaranteeing credit card.

3.5.5 The car must not be handed over to a driver who appears to be under the influence of alcohol or drugs.

3.5.6 The driver must be physically able to drive the car safely – must not be too tall, too short or too fat; if disabled, must be able to operate the controls.

3.5.7 The car must have been prepared — cleaned, full tank of fuel, oil and water topped up, tires properly inflated.

3.5.8 The car must have been checked for roadworthiness — tire tread depth, brake pedal and handbrake lever, lights, exhaust leaks, windscreen wipers.

3.6 No-shows

3.6.1 If an assigned car has not been picked up 90 minutes after the scheduled pickup time, it may be released for walk-in rental, unless the rental has been guaranteed by credit card.

3.6.2 If a rental has been guaranteed by credit card and the car has not been picked up by the end of the scheduled pickup day, one day's rental is charged to the credit card and the car is released for use the following day.

3.7 Return from rental

3.7.1 At the end of a rental, the customer may pay by cash, or by a credit card other than the one used to guarantee the rental.

3.7.2 If a car is returned to a location other than the agreed drop-off branch, a drop-off penalty is charged.

3.7.3 The car must be checked for wear (brakes, lights, tires, exhaust, wipers etc.) and damage, and repairs scheduled if necessary.

3.7.4 If the car has been damaged during the rental and the customer is liable, the customer's credit card company must be notified of a pending charge.

3.8 Early returns

3.8.1 If a car is returned early, the rental charge is calculated at the rate appropriate to the actual period of rental (e.g., daily rate rather than weekly).

3.9 Late returns

3.9.1 If the car is returned late, an hourly charge is made up to six hours' delay; after 6 hours a whole day is charged.

3.9.2 A customer may request a rental extension by phone – the extension should be granted unless the car is scheduled for maintenance.

3.9.3 If a car is not returned from rental by the end of the scheduled drop-off day and the customer has not arranged an extension, the customer should be contacted.

3.9.4 If a car is three days overdue and the customer has not arranged an extension, insurance cover lapses and the police must be informed.

3.10 Car maintenance and repairs

3.10.1 Each car must be serviced every three months or 10 000 kilometres, whichever occurs first.

3.10.2 If there is a shortage of cars for rental, routine maintenance may be delayed by up to 10% of the time or distance interval (whichever was the basis for scheduling maintenance) to meet rental demand.

3.10.3 Cars needing repairs (other than minor body scratches and dents) must not be used for rentals.

3.11 Car purchase and sale

3.11.1 Only cars on the authorised list can be purchased.

3.11.2 Cars are to be sold when they reach one year old or 40 000 kilometers, whichever occurs first.

3.12 Car ownership

3.12.1 A branch cannot refuse to accept a drop-off of a EU-Rent car, even if a one-way rental has not been authorised.

3.12.2 When a car is dropped off at a branch other than the pick-up branch, the car's ownership (and, hence, responsibility for it) switches to the drop-off branch when the car is dropped off.

3.12.3 When a transfer of a car is arranged between branches, the car's ownership switches to the "receiving" branch when the car is picked up.

3.12.4 In each car group, if a branch accumulates cars to take it more than 10% over its quota, it must reduce the number back to within 10% of quota by transferring cars to other branches or selling some cars.

3.12.4 In each car group, if a branch loses cars to take it more than 10% below its quota, it must increase the number back to within 10% of quota by transferring cars from other branches or buying some cars.

Figure SM5 Object view

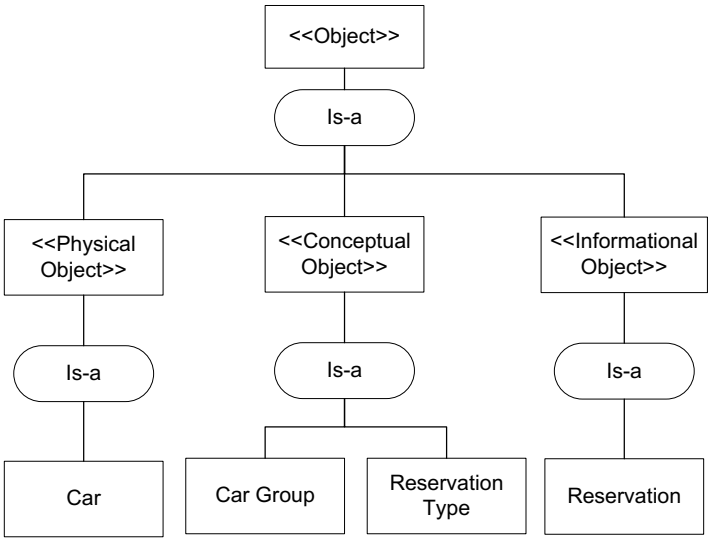


Figure SM6 Actor view

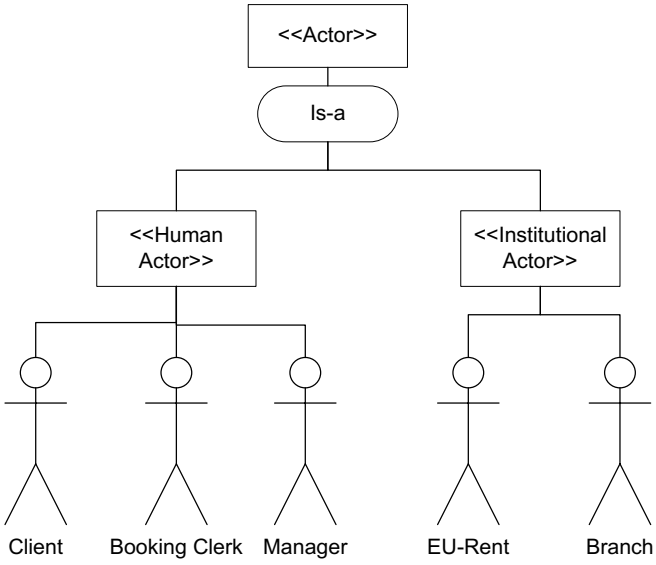


Figure SM7 Place view

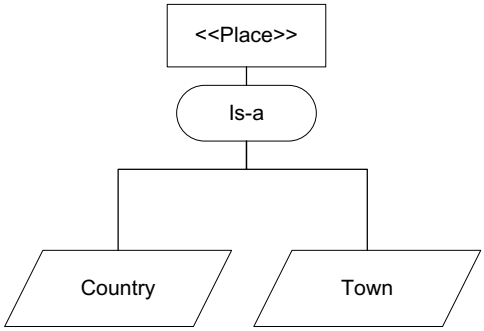


Figure SM8 Graphical representation of the static relationships

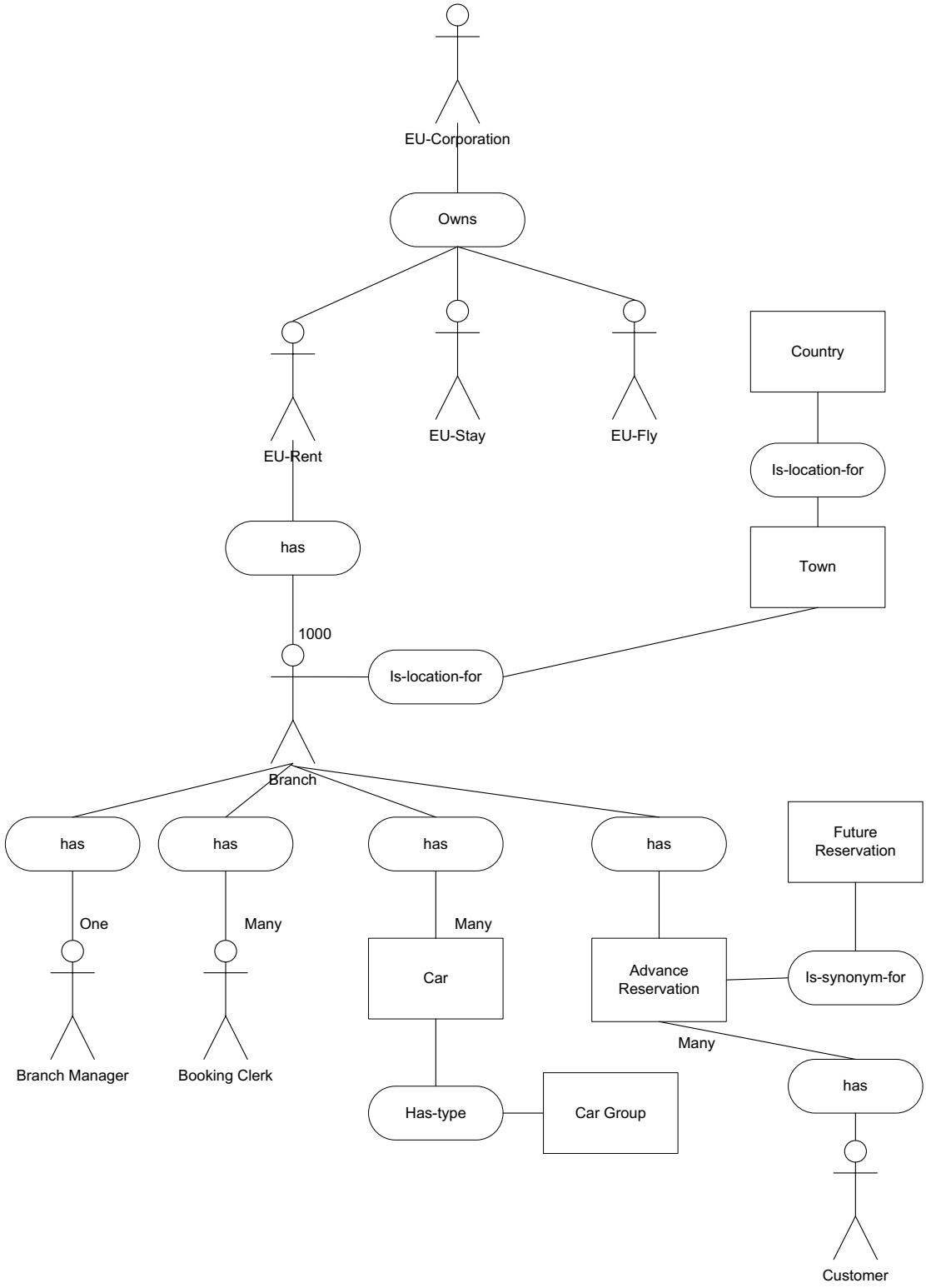


Figure SM9 Graphical representation of the dynamic relationships

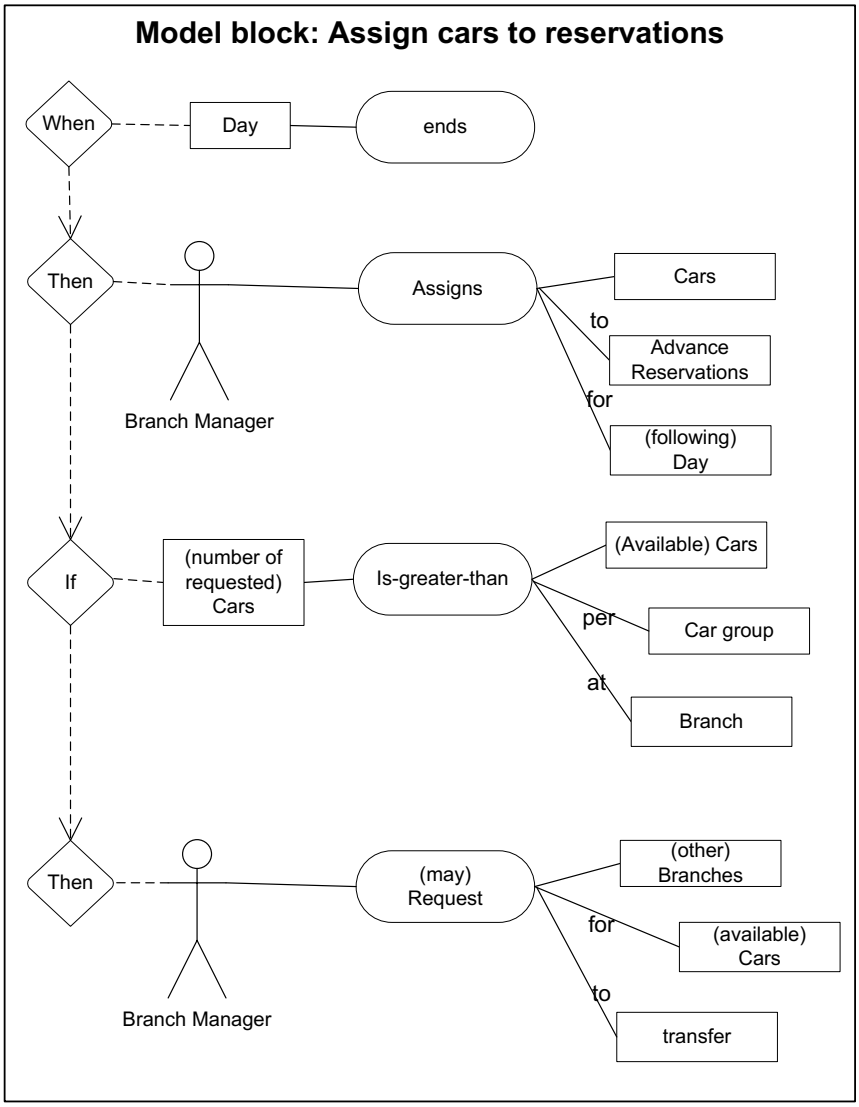


Figure SM10 Graphical representation of logical and causal relationships

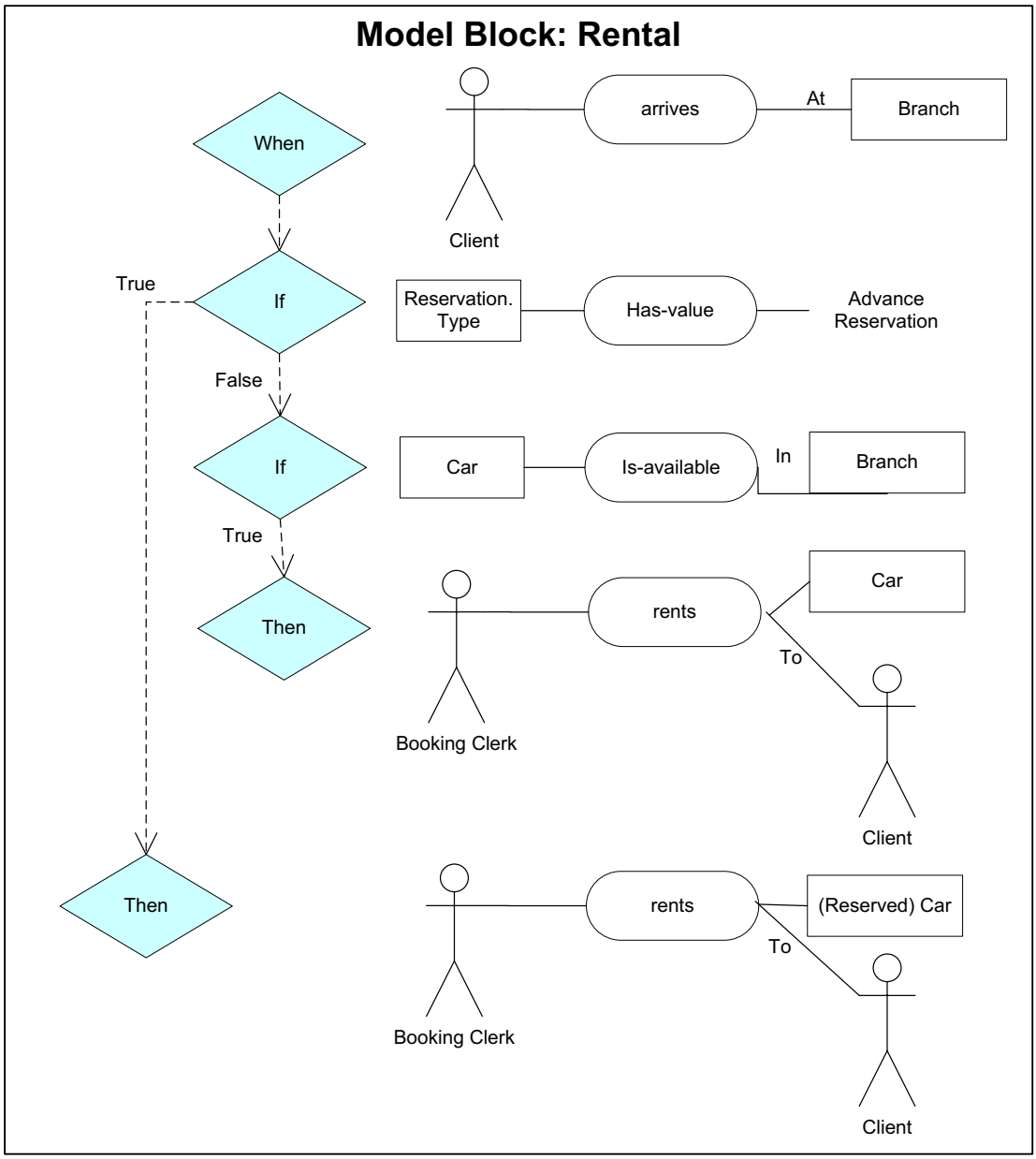
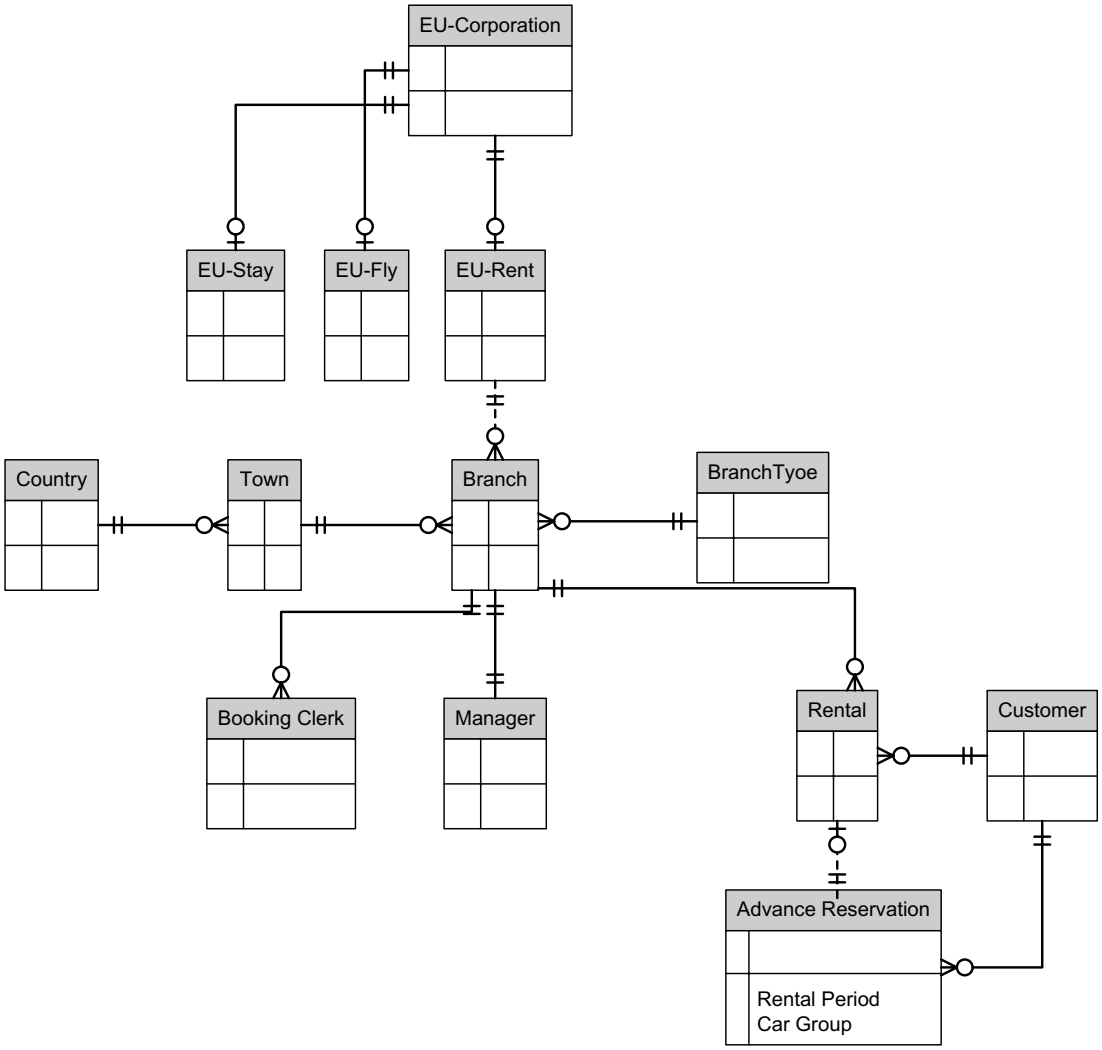


Figure SM11 Example of class diagram developed using the outcomes of the integrated modelling technique



Reference list to supplementary material

[1] P. Checkland, "Soft systems methodology: A thirty year retrospective," *Syst. Res. Behav. Sci.*, vol. 17, pp. 11–58, 2000.

[2] P. P.-S. Chen, "The entity-relationship model - Toward a unified view of data," *ACM Trans. Database Syst.*, vol. 1, no. 1, pp. 9–36, 1976.

[3] P. Prescott, *SQL: Learn the structured query language for the most popular databases including Microsoft SQL Server, MySQL, MariaDB, PostgreSQL, and Oracle*. CreateSpace Independent Publishing Platform, 2015.

[4] N. S. Umanath and R. W. Scamell, *Data modeling and database design*, 2nd ed. Course Technology, 2014.

[5] IDEF0, "Integration Definition for Function Modeling (IDEF0)," *Draft Fed. Inf. Process. Stand. Publ. 183 - Comput. Syst. Lab. Natl. Inst. Stand. Technol.*, vol. IV, no. June 1981, pp. 1–128, 1993.

[6] S. Kappes, "Putting your IDEF0 model to work," *Bus. Process Manag. J.*, vol. 3, no. 2, pp. 151–161, 1997.

[7] J. Bal, "Process analysis tools for process improvement," *TQM Mag.*, vol. 10, no. 5, pp. 342–354, 1998.

[8] G. Valiris and M. Glykas, "Business analysis metrics for business process redesign," *Bus. Process Manag. J.*, vol. 10, no. 4, pp. 445–480, 2004.

[9] G. Wagner, "The agent-object-relationship metamodel: Towards a unified view of state and behavior," *Inf. Syst.*, vol. 28, no. 5, pp. 475–504, 2003.

[10] OMG, "Business Process Modeling Notation (BPMN). Version 1.2," *OMG*, 2009. [Online]. Available: http://www.omg.org/spec/BPMN/1.2. [Accessed: 26-Mar-2018].

[11] OMG, "Unified modeling language: Infrastructure. Version 2.1.1," 2007. [Online]. Available: https://www.omg.org/spec/UML/2.1.1/About-UML/. [Accessed: 26-Mar-2018].

[12] OMG, "Unified modeling language: Superstructure. Version 2.1.1," 2007. [Online]. Available: https://www.omg.org/spec/UML/2.1.1/About-UML/. [Accessed: 26-Mar-2018].

[13] A. Evans, R. France, K. Lano, and B. Rumpe, "The UML as a formal modeling notation," *LNCS*, vol. 1618, pp. 336–349, 1999.

[14] PMBOK, "A guide to the project management body of knowledge," 1996. [Online]. Available: http://www2.fiit.stuba.sk/~bielik/courses/msi-slov/reporty/pmbok.pdf. [Accessed: 22-May-2018].

[15] O. Noran, "An analysis of the Zachman framework for enterprise

architecture from the GERAM perspective," *Annu. Rev. Control*, vol. 27, no. 2, pp. 163–183, 2003.

[16] P. J. Toussaint, A. R. Bakker, and L. P. J. Groenewegen, "Integration of information systems: Assessing its quality," *Comput. Methods Programs Biomed.*, vol. 64, no. 1, pp. 9–35, 2001.

[17] J. L. G. Dietz, "The atoms, molecules and fibers of organizations," *Data Knowl. Eng.*, vol. 47, no. 3, pp. 301–325, 2003.

[18] D. Hay and K. A. Healy, "Defining business rules: What are they really?," Final Report, 2000. [Online]. Available: www.businessrulesgroup.org. [Accessed: 26-Mar-2018].