



Data Model Classification for Interoperability in the Industry

Yuhan Chen, D Annebicque, Véronique Carré-Ménétrier, Philippot Alexandre,
T Daneau

► To cite this version:

Yuhan Chen, D Annebicque, Véronique Carré-Ménétrier, Philippot Alexandre, T Daneau. Data Model Classification for Interoperability in the Industry. IFAC Conference on Manufacturing Modelling, Management and Control (MIM), 2022, Nantes, France. pp.2354-2359, 10.1016/j.ifacol.2022.10.060 . hal-03716066

HAL Id: hal-03716066

<https://hal.science/hal-03716066>

Submitted on 7 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Data Model Classification for Interoperability in the Industry

Y. CHEN*, D. ANNEBICQUE**, V. CARRE-MENETRIER***, A. PHILIPPOT****, T. DANEAU *****

* *CRéSTIC, University of Reims Champagne Ardenne, France*
(e-mail: yuhan.chen@renault.com).

** *CRéSTIC, University of Reims Champagne Ardenne, France*
(e-mail: david.annebicque@univ-reims.fr)

*** *CRéSTIC, University of Reims Champagne Ardenne, France*
(e-mail: veronique.carre@univ-reims.fr)

**** *CRéSTIC, University of Reims Champagne Ardenne, France*
(e-mail: alexandre.philippot@univ-reims.fr)

***** *Renault Group, Renault Tecnocentre, Guyancourt, France*
(e-mail: thierry.daneau@renault.com)

Abstract: In the context of industry 4.0, interoperability is a major challenge for the manufacturing world. With new use cases heavily depending on industrial data analysis, data structuration through model has become essential for system and process description. However, diversities in data model types due to silo working between domains, represent a challenge for interoperability. Models don't have a clear definition of different domains and there is a need for alignment. This study proposes a classification of different types of data models based on capability criteria to help model definition alignment.

Keywords: Industry 4.0, data model classification, manufacturing, interoperability

1. INTRODUCTION

The fourth Industrial Revolution opens the door for new opportunities and production means to the manufacturing world. One major evolution comes in the form of global digitalization of the entire organization from factory floor to higher business domains. New use cases such as production monitoring or predictive maintenance are nowadays core components of Industry 4.0 thanks to the arrival of Information Technology's (IT) to the shop floor (Sajid, Haleem, Javaid, Goyal, & Mittal, 2021). These new perspectives all share a common characteristic which is the use of data from operational technology (OT) like robots, PLC, or Computerized Numerical Control units. Although the data collection subject has existed for decades in the IT domain, it remains relatively new for OT. With the progress of cloud computing and the use of big data, new information technologies now coexist alongside with traditional manufacturing means (robots, Programmable Logic controllers ...) for data collection.

However, data collection on the OT level faces major heterogeneity challenges with the large diversity of protocols and technologies coming from equipment suppliers. This is a serious problem for data exploitation as data needs to be understood by data engineers from the upper business levels.

Furthermore, the industrial ecosystem differs from the IT ecosystems because of the differences of constraints that exist between the two worlds. Consequently, the challenge of data heterogeneity must be addressed and adapted for existing equipment of the industrial ecosystem. Several approaches exist to tackle this problem such as data standardization (OPC UA Foundation, 2021) or data mediation (Paviot, Cheutet, & Lamouri, 2011), but all agree on the necessity of data structuration using data models. A Model is an important support for data stakeholders to understand and structure the data, but it is also impacted by diversity. In fact, among the same organization, different types of data models are used, and this often creates incompatibilities in an ecosystem where interoperability is the key to success. There is a real need for data stakeholders to agree on which types of models they need to use for their business process. The study is based on Renault Group industry data models and Renault's Industry 4.0 project called Industry Data Management 4.0. In this project, industrial data from the plant level is captured via standardized data model for business use cases such as maintenance or traceability.

Although there is a consequent state of art of data modeling for the IT domain, the main contribution of this paper is to assess commonly used models from an industrial point of view and to propose a simple and practical model classification based on different criteria relevant to business needs. Section 2 focuses on the definition and role of data models in the digital

manufacturing ecosystem. Section 3 details the methodology for model classification and section 4 compares data model types from an industrial perspective.

2. DATA MODEL OVERVIEW

2.1 Data model definition

Data modeling is a process to manage, organize and store data. It is a main requirement for data structuration and data exploitation, thanks to the work of formalization (Patel, 2019). The data model is the end product of data modeling process and serves as a link artifact for interoperability between other processes. In a world of Big Data and cloud computing, data models constitute the backbone of information exchanges and storage. It enables and illustrates the type of data types used and stored within the system, the relationships among these data types and their whole organization.

A data model is created according to industrial or business use cases (IBM, 2022). Rules and requirements are defined in a formal and a standardized way that every stakeholder can understand. It is the main requirement for data collection and exploitation.

Models come at various levels of abstraction and granularity. A high-level model such as a meta model can, for instance, represent a whole ecosystem whereas a more technical model could illustrate a specific or granular domain in a technical aspect. In this regard, semantics and data formats can greatly differ between different models.

As mentioned before, data modeling requires formal techniques and standardized schemas. Consistency and predictability among heterogeneous domains are vital for interoperability and understandability.

Historically, data modeling is not a new concept. In fact, information modeling has been conducted from the early 1958 by Young and Kent, who wanted to create a formal notation to organize any pieces of hardware at an abstract level. With time, modeling techniques have evolved with the increase in complexity and size of the modeled environment. The birth of concepts on information management, such as the management information system in 1960 has confirmed the necessity of usage of a paradigm revolving around data structuration and data relationship in software engineering.

In the continuity of data modeling, Model Driven Engineering (MDE) has emerged in 2000 to solve automation challenges for software production, maintenance and to ease software system accessibility (Silva, Valentin, Carvalho, & Barreto, 2021). Based on data modeling concepts, MDE englobes a new modeling technique that uses domain specific models and metamodels which represent the global architecture of a system.

For a long time, data models were specific for IT domains such as software engineering. Industry 4.0 pushes industrial and

manufacture domains to collect and exploit data from plant machines. There are many use cases possible and consequential benefits for companies. However, an industrial environment greatly differs with a traditional IT one with new types of challenges and barriers (Kamal, Mubarak, Scodova, & Naik, 2016). Equipment life cycles become a much more important issue for plants than software life cycle for instance. Industrial Equipment heterogeneity is also a big challenge and creates a great amount of data with different meanings and formats. To make collected data understandable and exploitable, data scientists need to perform data cleansing which impacts hugely on business performance. This can be overcome by using a formal and standardized way to represent a system therefore, using data models.

2.2 Model granularity

Data models come with different levels of abstraction. With a high level of abstraction, models are more conceptual and mainly aim to illustrate ideas and relationships. With less abstraction, it becomes more concrete and specific.

Data models are generally divided into three categories according to their abstraction level. The model with the highest abstraction is the conceptual model. Then come the logical model and the physical with the lowest degree of abstraction.

- **Conceptual data model (Fig.1).** Also referred to as domain model, they capture the main concepts and organization of a system. Conceptual model can also contain various business rules, requirements, and dependencies with other models. They are usually composed of entity classes that correspond to elements important for the business to represent in the data model. Relationship between classes must be clearly expressed as well as its characteristics and potential constraints. Notation remains simple and clear. Figure 1 exposes a conceptual representation of a plant organization.

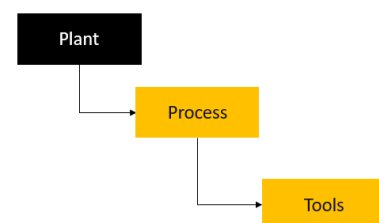


Figure 1: Conceptual Model

- **Logical models (Fig.2).** Compared to a conceptual model, a logical model is less abstract, and provides greater details about relationships and concept of a particular domain or system. It must follow a formal notation that describes data attributes such as types, data lengths and relationships. Logical models, however, don't express any technical system requirements as they are not meant to be directly implemented. Figure 2 adds variables related to every plant entity (ID, Name ...)

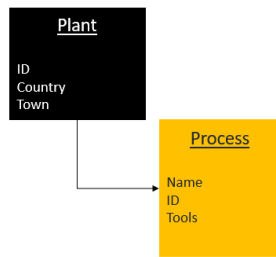


Figure 2: Logical Model

- **Physical data model (Fig.3).** Physical data models provide a schema for how the data will be physically stored within a database with the lowest level of abstraction possible. It is the final design that can be physically implemented in an information system such as a relational database. A same conceptual model can have different physical models depending on the system requirement of its implementation. Figure 3 specifies the type of variables from the logical model.

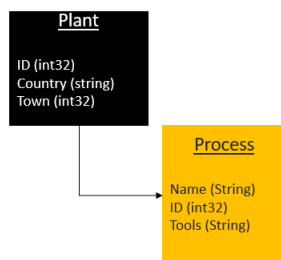


Figure 3: Physical Model

All these different data models exist in the data model creation process of Renault where conceptual ideas are modeled in the conceptual model and logical and physical models are for implementation purposes (Renault data model process).

3. DATA MODEL CLASSIFICATION

Models can be evaluated by different criteria and a common one is complexity. In the state of art, complexity in models and ontologies exist especially around object-oriented models. Complexity metrics can be applied to two main parts of a model: Relational complexity and Class complexity (Elish & Rine, 2003).

Though these metrics can be easily applied for the software engineering domain or for web ontologies, they remain unpractical for an industrial context. In a practical business environment, decision making need to be efficient and mobilize minimum human and time resources. With countless different models being used, a thorough complexity study of each model is not sustainable for the company. Simpler metrics need to be used to rapidly answer compatibility between different models. With a more functional approach,

models must be classified with the ability to satisfy complex uses cases.

Before deciding on the metrics, it is essential to know the whole digital course of data in a cloud manufacturing environment. The study is based on Renault digital ecosystem called Industrial data management 4.0 (IDM 4.0) which is one of the company's main data projects for Industry 4.0.

The goal of IDM 4.0 is to collect data from the plant level and store and exploit it on the business level. Industrial data models representing the functional system of a manufacturing process are created and exploited by data scientists. With the analyze of Renault's data models, we identified 3 main criteria, based on model capability, that can be applied to data models:

- **Structural capability:** It is the capability of a model to establish structure such as hierarchical or non-hierarchical links between entities. The structure is fundamental to represent a complex system and helps data engineers to better visualize the data context.
- **Semantical capability:** It is the capability of a model to incorporate "meanings" within the links and relations between concept and entities. Semantics can also include modeling rules and grammar to support more complex data models. From a data scientist's viewpoint, strong semantic capability is essential to prevent data exploitation ambiguities that lead to analyze errors.
- **Functional capability:** It is the capability of a model to integrate advanced functionalities to support model creation and model maintenance. For instance, an object-oriented model includes methods, interface, and inheritance to prevent redundancy in model creation. This is especially useful for highly complex models like process manufacturing models. Renault data models use the OPC Unified Architecture (OPC UA) information model that rely on inheritance of industrial assets such as industrial tools.

With the 3 criteria, data stakeholders can rapidly assess data model according to model capability and eliminate incompatible models. In the case of Renault, since OPC UA is an object-oriented model that have capabilities in structure, semantics and functionalities, it will have combabilities with other model meeting the same characteristics such as UML. The following data model scale can help to better visualize model capabilities (Fig.4).

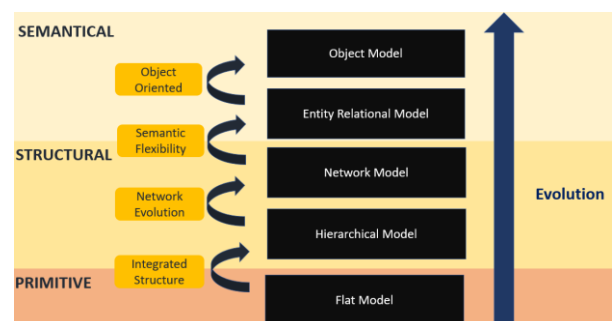


Figure 4: Data Model Scale

4 DATA MODEL ASSESSMENT

With the increase usage of data and its opportunity perspectives (artificial intelligence, business analysis, predictive maintenance ...), modeling has become a vital part of an organization (zur Muehlen & Indulska, 2010). However, model types still differ greatly from a domain to a another according to business and technical references. Throughout time, different models have emerged because of the evolution of business need. Each one has its own specification, degree of complexity and richness of information.

In the industrial domain where modeling is new, the concept of a “model” is not yet well defined and often creates confusion about its specification. For instance, there’s a huge gap of richness of information between a primitive model like a file-based model and a more evolved model (object-oriented model). For final uses cases as complexes as predictive maintenance or artificial intelligence, the definition of a model must be standardized for the whole organization at the start. To fill this gap of normalization, our approach aims to assess the main types of models used in the industry and create a classification according to a degree of complexity and richness of information.

3.1 Flat model

One of the most rudimentary and early data models are flat file models (Table 5). The flat model consists of a file where data are stored in row or columns in a uniform format. (Yusof & Man, 2018) There is however no structure for indexation and no relationships between the data. Implementation of flat model consists of a plain file in an information system. The file can be any text-based document such as csv, txt or binary. Although there are no technical means for relationship between data, it can still be indirectly implied by adding further information but the model. In table 5, you can see the data representation of an industrial robot axis.

	Actual X Position	Actual Z position	Actual Y Positon
Axis 1	12.3	25.1	-32.1
Axis 2	26.3	-69.3	26.5
Axis 3	-29.3	12.9	56.1

Table 5: Flat Model

Despite the lack of structure and semantic capability, flat models are still widely used in many information systems, especially in the industrial area where simplicity is key. This raises several concerns for data exploitation with the rise of industry 4.0. With data flowing into the hands of heterogenous domains, semantics are crucial to preserve data quality. Flat file models are for the most part unable to fill the requirement for data exploitation and requires adding substantial information for any implementation perspectives.

3.2 Hierarchical Model

Coming after flat file models are the hierarchical model. Designed like a tree model, hierarchical model introduces the

concept of hierarchy among model entities (Fig.6). Hierarchical modes enable the one-to-many relationship type and provides an easier data access (Educba, 2020). This means that every child only has one parent. The implementation of basic relationship information and knowledge adds a new level of complexity and widen its field of usage compared to flat models. It, however, suffers of several weaknesses and limitation due to its rigidity of design. For instance, hierarchy implies that relationship is limited to parent and child. Moreover, this kind of methodology does not fit all kinds of systems where parent/child logic is not prominent and leads to difficulty in model maintenance. Lack of structural independence and flexibility will limit hierarchical model to specific use cases, but their simplicity makes them still being used today. Figure 5 represents hierarchically domains of a digital ecosystem.

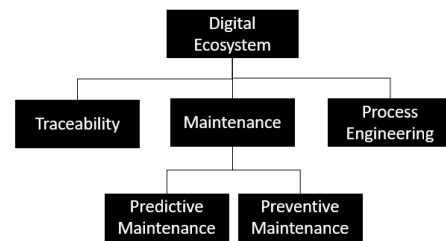


Figure 6: Hierarchical Model

Limitation of the first hierarchical models has conducted to its evolution. In a similar fashion, Network models are based in a hierarchical logic (Fig.7). The difference with the previous model resides in the possibility of many-to-many relationship type. This offers the possibility of a child to have multiple parents. Network models are more flexible than hierarchical models and can be used to model a larger variety of systems. However, relationships are still parent-to-child which make this a model with an incomplete flexibility.

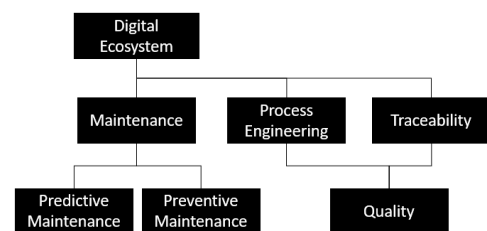


Figure 7: Network Model

The industrial domain is structured in a very hierarchized manner. For instance, according to Renault vision, a plant is composed of different operations. In each operation there can be different station with different processes. With this approach, the concept of hierarchy is essential to portray an industrial ecosystem. However, hierarchical based models are not enough to fulfill all the data requirements. The lack of ability to support properties under model entities can rise semantic concerns and affect information completeness.

3.3 Entity Relational Model (ER)

Previous model introduced structured relationships but lacked semantic information for data to be properly understood and exploited (Fig.8). Naturally, a new and more complex design was created with the possibility to add valuable semantic information. Entity relationship model add the new ability to create properties for model classes (Han, Xu, & Yao, 2010). Relationship types have expanded from a unique hierarchical type to multiple type of relationships by adding semantic information. These evolutions from previous model allows ER models to be far more flexible and suitable to represent any types of business environment and complex systems. ER models can come in different forms and variant with its own modeling rules and grammar and can be used as a domain specific ontology. Furthermore, its design permits a quick understanding from an outer perspective of the modelled system and is a favorable for any form of collaboration. ER logic is commonly used in software engineering, but it is also applicable for other domains. Figure 8 represents an example of generic information that can be collected from industrial machine tools.

Robot ID	Axis ID	Model ID
123	235	4566
124	236	4567
125	237	4568

Axis ID	Actual X Position	Actual Z position	Actual Y Position
235	12.3	25.1	-32.1
236	26.3	-69.3	26.5
237	-29.3	12.9	56.1

Model ID	Model Name
4566	Kuka
4567	ABB
4568	Fanuc

Figure 8: Entity Relationship Model

From an industrial environment perspective, ER model allows better depiction of industrial and manufacturing assets. For instance, modeling a tool can be done by adding properties such as the tool's name, localization, or physical measure. The tool's relationship can be expressed by linking it with another entity and by adding semantic information to inform the nature the relationship: the tool can be a part of a process and can be operated by an operator. Here the process and the operator are two different entities.

ER models are ideal for representing any kind of environment and provides semantic semantics and a formal grammar to be understood by all stakeholders. Consequently, they can be applied for different types of models (conceptual, logical and physical) and different granularities. In terms of modeling techniques, ER remains limited and do not integrate any advanced methodologies for certain modeling issues such as redundancy.

3.4 Object Oriented Model (OO)

With the growth of software development in the mid-90s, the new type of model, object-oriented model has gained popularity due to not only its richness of structure and

semantic but also the introduction of new methods within the model that facilitate the modeling process (Fig.9). In the object-oriented vision, a real-world element is represented by an object with different properties and methods (Bonnard, Hascoët, Mognol, Eduardo, & Alvares, 2019). For example, if we view a human as an object, it will have proprieties, called attributes in OO, such as the hair color or the height. The human object also has methods, which is basically actions, such as speaking or walking. Combined, attributes and methods can represent all kinds of environments and systems.

One major advantage of object-oriented model over ER models is the existence of advanced modeling techniques that open the door for more perspectives and use cases. These techniques are specific to object-oriented model and aims to solve different issues encountered in the modeling phase. Inheritance, for instance, makes it possible for a child class to inherit methods or attributes from a parent class. The human class can inherit from the mammal class which can itself inherit from the animal class.

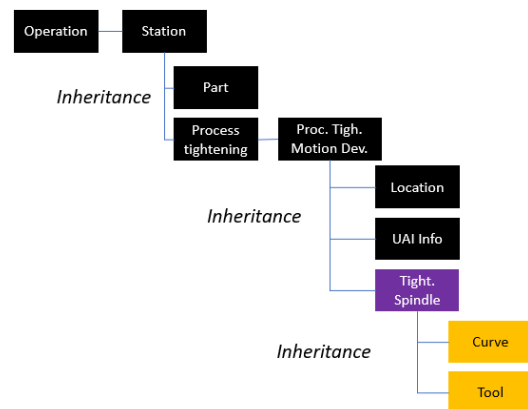


Figure 9: Object Oriented Model

There are many forms of object-oriented model that exists. One of the most known is the UML that consist of a standardized object-oriented modeling language. UML represents two different views of a system model with different diagrams (Muller & Gaertner, 2000):

- Static or (structural): It consists of the static structure of a system represented by attributes, operations, and relationships. It includes class diagrams and composite structure diagrams
- Dynamic or (behavioral): It is focused on the dynamic behavior of a system showing collaboration among objects and change to the internal state objects of objects. It includes sequence, activity diagrams and state machine diagrams.

Different types of diagrams and objects call for a need of centralization. For this purpose, object oriented has also developed the meta-modeling approach. The meta model is composed of different layers of granularity of an entire system. Object Oriented modeling approach is being used in the industry domain as well. Strong semantic and structural means combined with advance techniques suite modeling complex industrial system perfectly.

Modeling standards like OPC UA information model are based on object-oriented approach with inheritance as one the main pillar. OPC UA have base/generic classes, called nodes, that derives to more specific nodes which can be customized. Attributes follow the same design with generic ones (Variable type) that can be turned to specific. In figure 9 for instance, base objects that define generic industrial assets such as stations, processes or tools are defined as object types and can inherit specific objects.

5. DISCUSSION AND CONCLUSION

Interoperability is a major challenge for Industry 4.0 (Panetto, Lung, Ivanov, Weichhart, & Wang, 2019) and data model alignment is crucial for a data driven ecosystem. This study proposes a preliminary definition alignment for data models in an industrial environment based on 3 main criteria.

Although the data model scale helps to clear the path for model interoperability perspectives, the question remains complex and needs further studies. Industrial models must meet requirements like standards and organizational rules. Two models with the same nature (OO model for instance) might not be compatible if the internal structure or modeling rules are too apart from another. Moreover, the data model life cycle is also an important topic as they are vulnerable to internal and external changes. Impact assessment for model changes must be conducted to elaborate modeling rules if we wish to build a sustainable and scalable solution. Integrating external standards to existing data models for the purpose of interoperability could induce consequences to data conception and data exploitation.

In the thesis of “conception and propagation of industrial data model” in collaboration with Renault Group and CReSTIC lab of the university of Reims, we aim to find further answers for data model interoperability and sustainability with a manufacturing point of view. The study is based on internal Renault methodologies such as the creation of common variable dictionary for a global interoperability among domains. External factors will be taken into account as well like the German Mechanical Engineering Association, Verband Deutscher Maschinen- und Anlagenbau (VDMA) which create data models as well. Comparisons between Renault’s models and external models will allow us to establish a methodology for model conception and propagation rules.

ACKNOWLEDGMENT

This research and development undertaking is conducted in a context of an industrial and academic collaboration (CIFRE convention) between Renault Group and CReSTIC laboratory of University of Reims Champagne Ardennes, financed by Renault Group and the National Association of Research and Technology.

REFERENCES

- Bonnard, Hascoët, J.-Y., Mognol, P., E. d., & Alvares, A. J. (2019). Hierarchical object-oriented model (HOOM) for additive manufacturing digital thread. *Journal of Manufacturing Systems*.
- Educba. (2020). *Hierarchical Database Model*. Retrieved from <https://www.educba.com/hierarchical-database-model/>
- Elish, M., & Rine, D. C. (2003). Investigation of metrics for object-oriented design logical stability. *Seventh European Conference on Software Maintenance and Reengineering*.
- Han, L., Xu, J., & Yao, Q. (2010). Entity-Relationship semantic meta-model based on ontology. *2010 International Conference on Computer Application and System Modeling*.
- IBM. (2022). *What is data modeling*. Retrieved from <https://www.ibm.com/cloud/learn/data-modeling>
- Kamal, S., Mubarak, M., Scodova, B., & Naik, P. (2016). IT and OT Convergence - Opportunities and Challenges. *PE Intelligent Energy International Conference and Exhibition*.
- Muller, P.-A., & Gaertner, N. (2000). *Modélisation objet avec UML*.
- OPC UA Foundation. (2021). *OPC UA Information Model*. Retrieved from <https://opcfoundation.org/>
- Panetto, H., Lung, B., Ivanov, D., Weichhart, G., & Wang, X. (2019). Challenges for the cyber-physical manufacturing enterprises of the future. *Annual Reviews in Control*.
- Patel, J. (2019). An Effective and Scalable Data Modeling for Enterprise Big Data Platform. *International Conference on Big Data (Big Data)*.
- Paviot, T., Cheutet, V., & Lamouri, S. (2011). A PLCS framework for PDM/ERP interoperability. *International Journal of Product Lifecycle Management*.
- Sajid, S., Haleem, A., Javaid, M., Goyal, T., & Mittal, M. (2021). Data science applications for predictive maintenance and materials science in context to Industry 4.0. *Materials Today: Proceedings*.
- Silva, E., Valentin, E., Carvalho, J. R., & Barreto, R. (2021). A survey of Model Driven Engineering in robotics. *Journal of Computer Languages*.
- Yusof, K., & Man, M. (2018). Efficiency of flat file database approach in data storage and data extraction for big data. *Indonesian Journal of Electrical Engineering and Computer Science*.
- zur Muehlen, M., & Indulska, M. (2010). Modeling languages for business processes and business rules: A representational analysis. *Information systems*.