

Towards EPC standardization

From EPC Standard

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

The process of standard-making is fundamentally based on agreement and consensus of a domain community. This platform provides a web-based, collaborative system specifically designed for the purpose of EPC standardization. It is openly accessible for everyone interested. The platform provides a community where domain experts and practitioners can come together and take an active part in developing a standard for the EPC language. For further information regarding the standardization process and additional details, please visit [Collaborative Specification Engineering - How To](#).



Contents

[1 The Event-Driven Process Chain](#)

[2 EPC Process Elements](#)

[2.1 Event](#)

[2.2 Function](#)

[2.3 Operators](#)

[3 EPC Quality](#)

[3.1 Syntactical criteria](#)

[3.2 Semantic criteria](#)

[3.3 Pragmatic criteria](#)

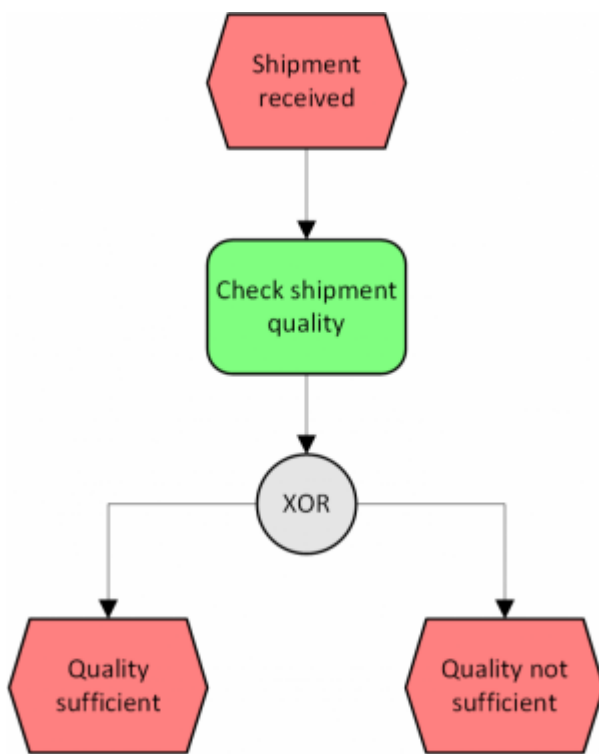
[4 Exchanging and storing EPC models](#)

[5 Exchange formats in literature](#)

[6 EPC extensions](#)

[7 References](#)

The Event-Driven Process Chain



The event-driven process chain (EPC) has been one of the most dominant languages for business process modelling over the last decades and is well established in both research and practice.^{[1][2][3]} The maturity of EPCs manifests itself in numerous scientific publications covering a wide range of language aspects. In addition, the EPC has proven its relevance in practice by being implemented in most common business process modelling tools.^[4] However, in contrast to languages such as Business Process Model and Notation (BPMN), whose popularity has been significantly boosted by the existence of a defined standard^[5] no systematic standardization effort for the EPC language has been made yet. Consequently, the absence of a standard hampers EPC usage and diffusion, especially due to difficulties in terms of interoperability, further development and overall acceptance.^{[6][7]} Nowadays, most business process modelling languages have been standardized by respective institutions, for example the Object Management Group (OMG) or the International Organization for Standardization (ISO). The publication of those standards ensures international adherence to specified language components such as syntax, notation or exchange format. Although there have been attempts to provide detailed specifications for certain aspects of the EPCs language (e.g. ^{[8][9]}), an official standardization process guided by a standardization development organization (SDO) has not been initiated to date. In addition, due to the widespread nature of EPCs and extensive previous research in the field, there exists a multitude of contributions ranging from various syntactical or semantical propositions to multiple language extensions^{[10][11]}, ultimately resulting in a mosaic-like EPC landscape. Naturally, this situation renders standard-making a difficult challenge.

EPC Process Elements

This section gives an overview over the EPCs' elements. Thereby already published literature about the EPC is recited. Therefore, the state of this chapter does not necessarily have to be coherent to the collaborative acquired state of this platform. The general elements of the EPC (speaking about the basis EPC, not the extended EPC) are the event, the function, and the three logical connectors AND, XOR and OR. Furthermore, the control flow arcs can be considered as a process element, too.

Event



The [Event](#) represents a current state in an EPC model and is both used as the starting and ending activity of an EPC model. Hence, every EPC model has at least one start and one end event. [\[12\]\[13\]\[14\]](#)

Events are able to trigger functions (and are triggered by them). Therefore, the event and the function form an alternating entity, whose only exception is the abovementioned start and end event. [\[8\]\[13\]\[15\]](#)

Furthermore, it is relevant to emphasize the non-existing decision-making power of an event. As it 'only' represents a current state, it is not able to decide about the further path of the model. Therefore, OR- and XOR-operators must not follow after an event. [\[13\]\[14\]\[16\]](#)

Function



The [Function](#) represents an action or a task in an EPC model. They can be therefore described as 'current state changes' or transformations from an initial state to a resulting state. Unlike events, they have decision-making power and are able to trigger all kind of operators. [\[8\]\[13\]\[15\]](#)

Operators



In order to represent possible splits and joins in the logical control flow, three types of [connectors](#) are used. These are the [AND operator](#), the [OR operator](#), and the [XOR operator](#). Following the AND operator, every ensuing path has to be treaded. After an OR operator, either one, two, three... or all ensuing paths "'can"' be treaded, while the XOR represents an exclusive decision (just one of the following paths can be treaded). Some literature demand that for the sake of semantics and pragmatics splitting paths need to be merged again by the same type of operator. [\[8\]\[13\]\[16\]](#)

EPC Quality

Aspects regarding the quality of EPC models can be distinguished in different categories. While there are general applicable quality recommendations like the seven process modelling guidelines [\[17\]](#), some quality frameworks like the 3QM or the SIQ Framework are concerning the categories "syntax", "semantic" and "pragmatic". Following this structure, EPC quality criteria will be assigned to these categories on this platform. As FELLMANN ET AL. already did such an assignment of EPC criteria [\[6\]](#), in the following their results will be presented in aggregate form.

Syntactical criteria

Representation

The representation of different EPC elements can be accessed via the respective pages on this platform.

Usage

Currently, there are 8 different quality criteria regarding the syntactical quality of EPC processes.

1. There has to be at least one start-event and one end-event.
2. The EPC is a bipartite graph with a set of events and functions. These elements are alternating, which means functions are followed by events and events are followed by functions. The only exception to this rule is the prior mentioned start- and end-event.
3. Information objects and organizational units can only be connected with functions. In this connection, organizational units have to be connected to functions involving human interactions.
4. An operator connects several events and functions. Joining operators have to be of the same type as splitting ones.
5. In conclusion, logical operators need at least one incoming ""or"" one outgoing control flow arc.
6. Events and Functions only possess one incoming and one outgoing control flow arc.
7. As an event lacks of decision making power, it must not be followed by an OR- or XOR-operator.
8. Process interfaces are linked to events only.

Semantic criteria

Linguistic Correctness

Please refer to the respective section of the [Function](#) and [Event](#) page.

Formal semantics

1. If two identical events occur before a join operator, the model can often be simplified.
 1. If the join operator is a XOR, there might be a modelling mistake.
2. AND- and OR-operators might not be followed by mutually exclusive Events.
3. If possible: Two mutually exclusive Events after a XOR operator are normally sufficient.
4. If operators are used for comparing values, often the case that the value remains constant is forgotten.
5. If there is a polar question, use the XOR operator.
6. A control flow arc should not be initiated more than once, so events and functions are triggered just once.
7. XOR-joins do not block if multiple tokens can arrive.
8. Each split-operator should have a corresponding join-operator.
9. It has to be possible to reach an end-event from every start-event.
10. If there are no further elements on a specific position in a process, this position has to be an end-event.
11. There are no elements in a process not integrated in a control flow from a start- to an end-event.

Compliance

1. The behaviour of a model should comply with rules specified with reference to the semantics of individual model elements.
2. The occurrence of model elements and the connections between model elements in an EPC model should comply with rules specified with reference to the semantics of individual model elements.
3. The model should comply with rules covering both the model elements and their attributes.

Pragmatic criteria

1. There should be no overlapping arcs.
2. There should be no change of flow direction.

3. The naming conventions should be observed.
4. If two or more consecutive functions (needless to say with events among them) are performed by the same organizational unit, not every function has to be connected to the respective organizational unit. Instead, it is sufficient to correlate functions to organizational units when there is a change of responsibility.
5. If the process is considerable large, it should be split with the aid of process interfaces.
6. There should be no unnecessary process elements.
7. Minimize the amount of control flow arcs.
8. Preferably, there should be just one start- and one end-event.
9. The process should be as structured as possible (cf. rule no. 8 of formal semantics)
10. OR-operators should be avoided as their semantics are vague.
11. The naming conventions should be applied.

Exchanging and storing EPC models

Exchange formats in literature

Exchange format	Desc.	LanguageType		EPC specific	Meta data	Layout
AML	ARIS Markup Language, proprietary file format for the ARIS Toolset	EPC	XML (proprietary)	×	✓	✓
XML for EPC	First step towards an EPC-specific exchange format	EPC	XML	✓	×	×
EPML	Event-driven Process Chain Markup Language, specifically designed to exchange EPC models	EPC	XML	✓	×	×
Fuzzy-EPML	EPML extension	EPC	XML	✓	✓	✓
oEPML	EPML extension	oEPC	XML (EPML)	✓	✓	✓
GXL	Graph Exchange Language, stores process models as graphs	EPC	XML (GXL)	×	×	×
sEPC	Exchanges semantically annotated EPC models	S-EPC	Ontology	×	✓	×

EPC extensions

EPC extension	Reference	Desc.	Specification	Type
EPC	[16]	EPC prototype, consisting only of functions, events and connectors	enumerative	-
eEPC	[18][19][20]	Enriched EPC, including organizational units, information objects, IT systems and process refinements	enumerative	element
rEPC	[21]	Includes a state machine, which is modelled in parallel to the EPC	formal semantics	execution
EPC*	[22]	Information objects can be connected to model data flow and conditions can be added to the control flow	-	workflow

oEPC	[23] [24]	Uses an object-oriented process definition	enumerative	element
Risk-EPC	[25]	Adds a risk element to the EPC, which can be linked to functions	enumerative	element
Fuzzy EPC	[26] [27] [28]	A fuzzy connector allows modelling fuzzy decision-making in business processes	formal semantics; meta model	element
yEPC	[29]	Extends the EPC to support state-based workflow patterns, multi-instantiation and cancellation	enumerative	workflow; element
Risk EPC extended	[30] [31]	Adds risk events, which are triggered when an exception occurs	enumerative; meta-model	element
C-EPC	[32] [33]	Functions, events and connectors are configurable	formal semantics; enumerative	reference modelling
Semantic EPC	[34] [35]	EPC elements are linked to ontologies	enumerative	semantic annotation
N-EPC	[36]	Allows modelling of several events between two functions, trivial events can be followed by XOR or OR	formal semantics	simplification
Service EPC	[37]	Functions are replaced by services	enumerative	element
C-iEPC	[38]	Extends the configuration introduced in the C-EPC to information objects, IT systems and organizational objects	formal semantics	extended configuration

References

2. T. Knuppertz and S. Schnägelberger, "Status Quo Prozessmanagement 2007/2008 - Ergebniszusammenfassung," Komptenzzentrum für Prozessmanagement, 2008
4. P. Fettke, "Ansätze der Informationsmodellierung und ihre betriebswirtschaftliche Bedeutung: Eine Untersuchung der Modellierungspraxis in Deutschland," Schmalenbachs Zeitschrift für betriebswirtschaftliche Forsch., vol. 61, pp. 550-580, 2009
6. C. Houy, P. Fettke, and P. Loos, "Stilisierte Fakten der Ereignisgesteuerten Prozesskette - Anwendung einer Methode zur Theoriebildung in der Wirtschaftsinformatik," in EPK 2009. 8. Workshop der Gesellschaft für Informatik e.V. (GI) und Treffen ihres Arbeitskreises "Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten (WI-EPK). Gesellschaft für Informatik, 2009, pp. 22-41
8. J. Drawehn, M. Kochanowski, and F. Kötter, Business Process Management Tools 2014. Stuttgart, Germany: Fraunhofer, 2014
10. J. Recker, M. Indulska, M. Rosemann, and P. Green, "How Good is BPMN Really? Insights from Theory and Practice.," in Proceedings of the 14th European Conference on Information Systems, 2006
12. M. Fellmann, S. Bittmann, A. Karhof, C. Stolze, and O. Thomas, "Do We Need a Standard for EPC Modelling? The State of Syntactic, Semantic and Pragmatic Quality," in 5th International Workshop on Enterprise Modelling and Information Systems Architectures (EMISA), 2013, pp. 103-116
14. R. K. L. Ko, S. S. G. Lee, and E. W. Lee, Business process management (BPM) standards: a survey, vol. 15, no. 5. 2009
16. M. Nüttgens and F. J. Rump, "Syntax und Semantik Ereignisgesteuerter Prozessketten (EPK)," Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen, vol. P-21, no. 6, pp. 64-77, 2002

18. J. Mendling, "Detection and prediction of errors in EPC business process models.," Institute of Information Systems and New Media, University of Vienna, Vienna, Austria, 2007
20. P. Rittgen, "Quo vadis EPK in ARIS? Ansätze zu syntaktischen Erweiterungen und einer formalen Semantik," *Wirtschaftsinformatik*, vol. 42, no. 1, pp. 27-35, 2000
22. P. Fettke, C. Houy, and P. Loos, "Zur Bedeutung von Gestaltungswissen für die gestaltungsorientierte Wirtschaftsinformatik – Ergänzende Überlegungen," *Universität des Saarlandes*, 191, 2010
24. R. Laue and J. Mendling, "Structuredness and its significance for correctness of process models," *Inf. Syst. Ebus. Manag.*, vol. 8, no. 3, pp. 287-307, Jun. 2009
26. M. Nüttgens and F. J. Rump, "Syntax und Semantik Ereignisgesteuerter Prozessketten (EPK)," *Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen*, vol. P-21, no. 6, pp. 64-77, 2002
28. J. L. Staud, *Geschäftsprozessanalyse*. 2006
30. R. Klein, F. Kupsch, and A. Scheer, "Modellierung inter-organisationaler Prozesse mit Ereignisgesteuerten Prozessketten," *Inst. für Wirtschaftsinformatik*, no. November, 2004
32. G. Keller, M. Nüttgens, and A.-W. Scheer, "Semantische Prozeßmodellierung auf der Grundlage 'Ereignisgesteuerter Prozeßketten (EPK),'", *Universität des Saarlandes*, 89, 1992
34. J. Mendling, H. Reijers, and W. van der Aalst, "Seven process modeling guidelines (7PMG)," *Inf. Softw. ...*, 2010.
36. W. Hoffmann, J. Kirsch, and A.-W. Scheer, "Modellierung mit Ereignisgesteuerten Prozeßketten," *Universität des Saarlandes*, 101, 1993
38. J. Galler and A.-W. Scheer, "Workflow-Management - Die ARIS-Architektur als Basis eines multimedialen Workflow-Systems," *Universität des Saarlandes*, 108, 1994
40. G. Keller and T. Teufel, *SAP R/3 prozeßorientiert anwenden*. Bonn: Addison-Wesley, 1997
42. W. Hoffmann, R. Wein, and A.-W. Scheer, "Konzeption eines Steuerungsmodells für Informationssysteme-Basis für die Real-Time-Erweiterung der EPK (rEPK)," *Universität des Saarlandes*, 106, 1993
44. O. Zukunft and F. Rump, "From Business Process Modelling to Workflow Management - An Integrated Approach," in *Business Process Modelling*, B. Scholz-Reiter and E. Stickel, Eds. Berlin: Springer, 1996, pp. 3-22
46. A.-W. Scheer, M. Nüttgens, and V. Zimmermann, "Objektorientierte Ereignisgesteuerte Prozeßkette (oEPK) - Methode und Anwendung," *Universität des Saarlandes*, 141, 1997
48. M. Nüttgens and V. Zimmermann, "Geschäftsprozeßmodellierung mit der objektorientierten Ereignisgesteuerten Prozeßkette (oEPK)," in *Informationsmodellierung - Referenzmodelle und Werkzeuge*, M. Maicher and H.-J. Scheruhn, Eds. Wiesbaden, Germany: Gabler, 1998, pp. 23-35
50. E. Brabänder and H. Ochs, "Analyse und Gestaltung prozessorientierter Risikomanagementsysteme mit Ereignisgesteuerten Prozessketten," in *EPK 2002 - Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten*, *Proceedings des GI-Workshops und Arbeitskreistreffens*, 2002, pp. 1-5
52. O. Thomas, C. Hüsselmann, and O. Adam, "Fuzzy-Ereignisgesteuerte Prozessketten - Geschäftsprozessmodellierung unter Berücksichtigung unscharfer Daten," *EPK 2002 - Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten*, *Proc. des GI-Workshops und Arbeitskreistreffens*, pp. 7-16, 2002
54. O. Thomas and T. Dollmann, "Attributierung und Regelintegration," *Proc. 5th GI Work. Event-Driven Process Chain. (EPK 2006)*, pp. 49-68, 2006
56. O. Thomas, *Fuzzy Process Engineering*, 1st editio. Wiesbaden: Gabler Verlag, 2009

58. J. Mendling, G. Neumann, and M. Nüttgens, "Yet another event-driven process chain," in Lecture Notes in Computer Science, 2005, vol. 3649, p. 428
60. M. Rosemann and M. zur Muehlen, "Integrating Risks in Business Process Models," Australas. Conf. Inf. Syst., no. Figure 1, pp. 62–72, 2005
62. T. Rieke and A. Winkelmann, "Modellierung und Management von Risiken. Ein prozessorientierter Risikomanagement-Ansatz zur Identifikation und Behandlung von Risiken in Geschäftsprozessen," Wirtschaftsinformatik, vol. 50, no. 5, pp. 346–356, 2008
64. M. Rosemann and W. M. P. van der Aalst, "A configurable reference modelling language," Inf. Syst., vol. 32, no. 1, pp. 1–23, 2007
66. J. C. Recker, M. Rosemann, W. M. P. van der Aalst, and J. Mendling, "On the Syntax of Reference Model Configuration - Transforming the C-EPC into Lawful EPC Models," in Business Process Management Workshops: BPM 2005 International Workshops, BPI, BPD, ENEI, BPRM, WSCOBPM, BPS, 2005, pp. 60–75
68. O. Thomas and M. Fellmann, "Semantische Ereignisgesteuerte Prozessketten.," Data Warehous., pp. 205–224, 2006
70. A. Filipowska, M. Kaczmarek, and S. Stein, "Semantically annotated EPC within semantic business process management," in Lecture Notes in Business Information Processing, 2009, vol. 17, pp. 486–497
72. O. Kopp, T. Unger, and F. Leymann, "Nautilus Event-driven Process Chains: Syntax, Semantics, and their mapping to BPEL," in Proceedings of the 5th GI Workshop on Event-Driven Process Chains (EPK 2006), 2006, pp. 85–104
74. S. Huth and T. Wieland, "Geschäftsprozessmodellierung mittels Software-Services auf Basis der EPK," in Service-orientierte Architekturen, 1st ed., V. Nissen, M. Petsch, and H. Schorcht, Eds. Wiesbaden: Deutscher Universitäts-Verlag, 2007, pp. 61–76
76. M. La Rosa, M. Dumas, A. H. M. ter Hofstede, J. Mendling, and F. Gottschalk, "Beyond Control-Flow: Extending Business Process Configuration to Roles and Objects," vol. 5231, pp. 199–215, 2008

[Category:](#)

This page was last edited on 25 January 2021, at 15:42.

Content is available under [Creative Commons „Zero“ \(Gemeinfreiheit\)](#) unless otherwise noted.

[Privacy policy](#)

[About EPC Standard](#)

[Disclaimers](#)

[Powered by MediaWiki](#)