Online Resource 2 Supplement data table for Figures 8 and 9 in main article. This table contains all the references with their corresponding identified complexity drivers, structured in two-dimensions: by complexity viewpoint (keyword) and by common attribute group.

	Complexity viewpoint							
	Social		***Process***		***System***			***Tooling***
Relevant Viewpoint keyword / **Common Attribute group**	*Organization*	*Designers / Human developers*	*Process / Activities / Methods*	*Project*	*Domain / Discipline*	*Product/ System under development*	*Design & analysis knowledge, information and data*	*Tools / IT systems / Databases*
Quantification (Number / Amount / Degree / Ratio / Size / Scale Of)	- Involved Organizations [1] [2] [3] [4] [5] [6] [7][8] [9] [10] [11][12] [13] [14] [15] [16] [17] [18] [19] size [20] [21] [22] [23] *(more like larger parties → aero-industry, small individual firms → construction)* [24] *partners* [25] [26] [27] - Involved Departments / teams / functional units/ programs [28] [29] [30] [31] [32] [2] [4] [33] [34] [8][9] [10] [11] [12] [13] [15] [35] [36] [37] [38] [39] [40] [20] [41] [42] [43] [44] [45] - Dependencies, interfaces, interactions among organizational units / departments /teams or with other organizations [46] [28] [5] [47] [15] [39] [40] [20] [41] [22] - Dependencies, interfaces, interactions among organizational skill sets [46] [48], need for expertise - External relationships and dependencies with suppliers and customers, tool vendors [35] [38] [48] [49] [50] [51] [52] [20] [18] [19] [53] [22] [25] [26] [43] [54] - *Size of portfolio of products* [55]	- Designers (and other stakeholders) [29] [30] [31] [32] [56][3] [4][57] [5] [6] [34][58] [7] [8] [9][59][60] [10] [61] [11] [12] [47] [13] [62] [15] [35] [36] [63] [64] [38] [17] [65] [48] [39] [66] [40] [50] [67] [51] [68] [69][70] [20] [18] [19] [71] [21] [72] [73] [22] [74] [75] [54] [76] [77] [78] [45] - Dependencies, interfaces, interactions and exchanges among designers / information flows (and other stakeholders) [46] [28] [31] [3] [4] [57] [5] [7] [79] [8] [59] [60] [61] [11] [80] [15] [35] [36] [65] [48] [81] [39] [49] [66] [40] [67] [68] [20] [41] [22] suppliers [52]*, information flows* [18] [19] [72] [53] [22] [74] across-companies [25]*, asynchronous exchanges* [82] - Work per person (work distribution) [28] [3] [7] [17] [39] [72] (large workload due to many tools and a lot of manual inputting to tools) [77] - Flexibility of human resources (which resource is capable to work on what task) [72]	- (Sub-)Processes, methods, ways of working / activities/tasks [53] [31] [4] [33] [57] [79] [10] [83] [47] [14] [84] [35] [63] [85] [38] scale [16] [48] [81] [66] [67] [86] [71] [53] [26] [23] steps/activities [18] [19] [53] [22] [55] [25] [78], *tasks* [77] - Process models [69] [71] - Processes' concurrency/ parallelism [53] [87] [1] [3] [6] [9] [59] [60] [64] [65] [88] [81] [67] [20][18] [72] [53] [22] [75] *software and hardware* [55] - *Processes' iterations and process duration / temporal extent [product lifecycle have multiplicity* [26]*]* [53] [89] [57] [59] [11] [35] [65] [90] [66] [51] [53] [22] (to progress the design but also to correct errors [rework]) [72] [22] *size of the iterations (small change a component, large change the whole design based on market feedback)* [91]*, duration* [25] - *Process phases or stages* [3] [4] [33] [60] [10] [61] [47] [14] [35] [17] [49] [50] [90] [67] [68] [69] [20] [19] [22] [54] [44] [24] [77] [92] *(stages are orthogonal to each other)* [26] referred to as technical domains (requirements, structure, behavior, V&V) [38] - Dependencies, interfaces, interactions / connections [46] [89] [87] [31] [47] [65] [48] [88] [93] [66] [40] [68] [20] [41][18] [72] (information flows) [53] [22]*, asynchronous exchanges* [82]*, inputs and outputs* [77] - Process loops/ jumps [87] [93] - Feedback processes [66] *feedback delays* [72] - Process directions / alternatives [87] [33] - Process approval/ release management/ archiving [3] [33] [65], bureaucracy [76] - Of Process standards [86] [77] - Degree of formalization of the process (for example in aerospace is highly formal) [77] - number of involved engineering decisions [53]	- Projects [57] [79] [8] [9] [12] [62] [15] [35] size /scale [38] [39] [20] [55] [75] [94] [45] - Sub-projects [3] [55] - Project phases [3] - Work and how it is distributed [28] [57] [75] - Project needs [40]	(Sub-) Disciplines involved [28] [31] [1] [56] [3] [4] [33] [5][95] [6] [58] [7][79] [9] [59][83] [61] [11] [12] [13] [14] [96] [84] [80] [97] [35] [36] [63] [64] [85] [38] [98] [16] [17][65] [48][88] [81] [39][49] [93] [90] [66][40] [50] [67] [51] [86] [70] [20] [99] [19] [72] [53] [73] [22] [74] [55][42] [25] [26] [75] [24][100] [101] [102][94] [103] [45][104] *even within software engineering* [92] - *Technologies* [105] [4] [33] [47] [38] [106] [17] [65] [66] [50][19] *Interactions* [81] [40] [67] [51] [68] [20] [99] [41] [72], asynchronous data exchanges [82] [100]	- Systems (e.g. SoS) [53] [107] [30] [56] [3] [5] [8] [59] [10] [108] [61] [11] [47] [14] [96] [84] [15] [35][63] [64][98] [16] [67] [86] [19] [72] [53] [73] - Subsystems [53] [28] [56] [3] [109] [33] [5] [58] [79] [59] [10] [108] [61] [11] [47] [14] [84] [15] [35] [110] [63] [64] [106] [17] [65] [48] [81] [49] [93] [68] [86] [19] [21] [72] [53] [22] [74] [43] - *Components* [28] [111] [56] [3] [109] [33] [79] [59] [10] [83] [108] [61] [11] [47] [14][84] [15] [35] [110] [63] [64] [37] [98] [16] [106] [17] [81] [39] [93] [68] [68] [86] [99] [19] [21] [72] [22] [74] [43] [102] product structure breadth (number of components) [53] - *Physical phenomena involved (electrical, mechanical, thermal, structural, etc.)* [101] [103] - Functions, their interactions and constraints [38] [53] [107] [105] [109] [58] [10][14] [35] [38] [17] [81] [67] [68] [99] [22] behavior or functionality [19] *constraints* [40][73] [43] [112] function density [49] [86], functionalities and other characteristics (the "ilities") [71] - System variants / Product Families / Product Lines [105] [28] [29] [113] [46] [89] [3] [109] [33] [59] [47] [13] [35] [110][37] [38] [98] [17] [40] [53] [114] [24] [78] *product diversification* [19][115] [26] - Dependencies, interfaces, interactions, connections (both direct and indirect), relationships, i.e. density/strength thereof [46] [111] [116] [117] [107] [28] [109] [33] [5] [6] [59] [10] [83] [108] [61] [11] [47] [14] [96] [15] [36] [63] [64] [98] [17] [118] [65] [49] [93] [40] [68] [86] [99] [53] [74] [43] degree of interconnectedness [72], interactions between physical and computational components [22]*, interactions of sub-functions* [22]*,* Number or degree of interactions with environment [53] - Parameters, variables [111] [29] [109] [61] [47] [14] [35] [49] [40] [51] [68] [71] - Areas / Environment of application [33] [14] [96] [15] [35] [49] [51] [68] physical or sociological and organizational [21], interdependence with the environment [26] - Features [108] [96] [15] [37] [74], States [61] - Software code lines / Software intensity (computational resources) [47] [15] [16] [17] [68] [21][55] [23] [102] - Serialization (production in mass or custom) / Production concepts (MTS, ATO, MTO, ETO) [47] [35] [37] [39] [115] [26] - Requirements [47] [14] [97] [110] [39] [51] [68] [19] [115] functional requirements [21], Capabilities (autonomy, etc.) [96] - Of data that the system has to process [96] - Expected lifetime / lifespan / lifecycle of the system [15][110] [17] [49] [51] [20] [53] [22] [42] [114] - System scale CPS is large scale [68], Size [72] [23] - Materials used [21] - Degree of symmetry [72] - Depth of the product (number of levels in the hierarchy) [53] For a production system (the number of required process steps and their interlinking in a process network) [53] - Product/System type [26]	- Design information, knowledge and data resources / engineering artifacts (models, drawings, diagrams, documents, collections of information (i.e. building blocks), and other sources of data (like big data) etc.) [53] [31] [1] [119] [56] [120] [3] [109] [4] [57] [5] [6] [34] [58] [79] [60] [10] [83] [108] [61] [11] [12] [47] [13] [14] [96] [84] [80] [97] [36] [110] [63] [64] [38] [16] [106] [118] [48] [39] [40] [68] [52] [70] [19] [71] [72] [74] [115] [26] [43][112] [114][121] [23] [94] [45] [104] size of the models (tens of thousands of equations with algebraic loops and non-linearities) [101] amount of information received at the same time (information flow) [53] [22]*, variety of engineering information (design models, analysis results, inspection data)* [82]*, information about requirements, test cases, source code, etc.* [122] - Data/ Information (product) models and structures [120] [3] [3] [4] [95] [6] [59] [10] [83] [108] [61] [13] [80] [35] [36] [38] [65] [39] [49] [19] [100] - (Entity) Relationships in the data models and/or the resources (between systems, functions, requirements, analysis, risk, costs, schedule, and discipline-specific etc.) [120] [3] [95] [6] [79] [60] [10] [108] [61] [11] [80] [97] [35] [36] [63] [64] [38] [65] [66] [40] [50] [67] [68] [52] [72] [122] - Data standards / formats / types [56] [3] [4] [95] [6] [79] [60] [10] [47] [35] [16] [106] [65] [40] - Modelling languages (CAD, Simulink) / approaches [10] [13] [35] [36] [110] [65] - Data and information concurrency [3] [80] - Number of views / Model size [38] [49] - Speed of incoming information [48] - Degree of conflict due to contradictory information [48] -	- Tools, databases, servers, infrastructure, services (i.e. data sources) [30] [31] [1] [56] [3] [4] [123] [6] [34] [79] [9] [59] [83] [11] [47] [13] [14] [96] [84] [62] [15] [35] [64] [38] [17] [65] (heavyweight) [16] [48] [39] [49] [40] [68] [86] [70] [99] [19] [53] [22] [42] [26] [43] [75] [112] [23] [78] [101] [102] [94] [45] *different tools of HW and SW* [55], number of databases [78], sets of specialized development tools which are specific for the particular development phase [92] - Interactions of tools, databases, servers, infrastructure and services [1] [5] [6] [83] [11] [12] [13] [96] [80] [35] [38] [17] [48] [19] [22] [26] [122] [45] - Data / Information (exchange) standards formats [4] [6] [79] [35] [64] [98] [16] [65] [40] [26] [44] - Needs the tool needs to cover [38] [17] [49] - Size of the files/order of system models the tools/solvers have to manage (may be hard to process) [40] [114] [23] *modelling large systems is neglected by many UML tools, difficult to create modular and thus understandable models* [124]*, simulation and verification tools do not scale to the complexity of extremely large models (tens of thousands of equations with algebraic loops and non-linearities)* [101]
Diversity (Homogeneity & Heterogeneity)	- Of Organizational structure / Structure of workforce [9] [40]*, across- collaborating companies* [25] - *Diversity, in case of companies collaborating or a set of companies as part of a group/consortium* [25] [26] - Cultural [3] [123] [5] [7] [8] [10] [96] [63] [106] [17] [118] [90] [67] [26][125] (per region of the world and culture) [52] [20], globalization [78], cross-company collaboration [25] - Strategical (in management, vision, training, etc.) [46] [123] [33] [7] [10] [80] [35] [63] [17] - Geographical [46] [2] [8] [15] [71] [53] [55] [25] [26] [75] [78] - Temporal - Variety of customer needs / type of industry (large series, small series, one-of-a-kind) [29] [109] [33] [5] [7] [126][35] [55] [26] (affected by trends like regionalization, fragmentation and saturation) [53] - Of spoken Languages [8] [53] - Differences between management and practitioners (designers, engineers, architects) [52]	- Contractual relationships / responsibilities and roles (including at different levels system vs SoS) [3] [33] [57] [9] [14] [15] [64] [16] [49] [40] [67] [86] [74] [76] in SoS is not clear whose responsibility it is to integrate the SoS [73] - Personality / Mentality / Proactivity / Perspective / Motivation (incl. personal biases) [29] [32] [128] [123] [14] [96] [80] [16] [118] [118] [81] [90],[25] - Attitudes people have and convey to others [69] - Workforce age [47] - Educational backgrounds and semantic gaps [32] [6] [9] [60] [80] [35] [63] [65] [39] [49] [66] [40] [68] [67] [22] [25] [26] [94] master social as well as technical sciences [50], - Personal values, motivations, preferences [119] [128] [123] [6] - Jargon / Vocabulary / Terminology (impreciseness of language) [32] [119] [6] [9] [36] [63][38] [17] [118] [49] [70] [19] [74] About tooling concepts [114] - Knowledge level and skills / expertise [30] [123] [126] [10] [11][13] [14] [96] [80] [97] [35] [63] [38] [16] [106] [17] [39]	- In process, methods, way of working (per person, team, department, per discipline, and in research) [86] [6] [79] [10] [84] [97] [39] [86] [26] for simulation [64], per discipline [98] [67] [19], between data science and manufacturing /design [48], different ways of working software and physical systems [81] [88] [71] [74] , between architects and engineers [51], application area/ discipline [51] [22] [24], per regions and countries [52]*, diverged management paths of HW and SW* [55] - *In process models (a plethora of different and often overlapping models used in the same process)* [69] - *In activity-related information / aspects (information related to one activity expressed in different ways in multiple models)* [69] - In product lines [97] - In process model types (procedural, analytical, abstract, management science/ operations research) [66] - *Processes distribution* [110] [16] - *In Hardware and software lifecycle business processes and how to manage them* [55] - *Distribution of Information exchanges of activities, tasks (if activities are dependent or independent)* [89] [29] [6] [60] [47] [98] [40] - *Purpose of the process (model) / Task the model is used for [visualization, planning, execution and control, development]* [93] [66] [69] - *What the users see the process model as being (they mean different things to different people in different situations) and what roles they think the models play in determining the design process itself* [69] [71]*,* [19], there are two different needs: methods for system design *and methods for process support and coordination of the development tasks* [22] - *In type of business model addressed by the process (solution providers, component providers, product developers)* [19] - *In Industrial domain addressed by the process* [19] [71] - *With respect to the dealing with the complete lifecycle* [19], different management and support needs for processes in the	- In project management styles / approaches [64] [81] [39] [52] [19], activity-based vs information-centric [24] - In project phases [3] - Of teams within the project [39] [24] - With relation to product lines [97] [19] - Between project-level and organization common needs [1] [119] [57] [13] [15] [69] [22]*, autonomy of the project vs. organization* [54] - Between projects [95] [47] [13] [15] [35] [40] [52] [69] [19] [122] [78] - In distribution [75] - Business goals lead to different dominant	- Abstraction level / Level of detail [3] [57] [95] [10] [11] [13] [14] [97] [62] [15] [35] [36] [63] [64] [37] [38] [98] [16] [129] [106] [17] [65] [81] [39] [49] [68] [70] [19] [22] [43] [100] [101] - Jargon/ Local concepts / terminology (acronyms, vocabulary) [semantics/ impreciseness of language] [1] [32] [119] [6] [9] [13] [36] [63] [38] [98] [17] [118] [49] [70] [19]. About tooling concepts [114] - Perspectives / characteristics of the discipline/ domain (with regards to the system/process) [FRAGMENTATION] [29] [3] [4] [57] [95] [8] [59] [10] [61] [11] [47] [13] [84] [80] [97] [62] [35] [36] [63] [63] [64] [37] [85] [38] [98] [106] [17] [65] [81] [49] [66] [40] [50] [67] [70] [20] [19] [22] [43] [24] [100] [91] [23] [104] semantic complexity [101], semantic gaps between disciplines [94]*,* between system design and system analysis [129], with regards to system properties such as reliability and robustness (difference electrical and mechanical) [49], software and physical [68] [74] *(including management and designers)* [69]*, perspectives of PLM vs ALM systems (difference between HW and SW)* [55], how perspectives and viewpoints are integrated in frameworks [127]	- (Sub-) System Definition (views) / Abstraction levels [57] [6] [58] [8] [59] [10] [83] [61] [11] [47] [13] [14] [84] [80] [97] [62] [35] [36] [63] [64] [37] [85] [38] [98] [16][129] [106] [17] [65] [81] [39] [49] [40] [50] [67] [68] [70] [19] [22] [43] [121] [24] [23] decomposition based on physical phenomena [101] [103] frameworks [127] [100] - Type of system [3] [59], product categories (in CPS) [101] - In product lines [97] [65] - Manufacturing style [39] - Types / Definition / of Interfaces and interactions (both direct and indirect) [116] [117] [107] [29] [46] [59] [61] [14] [14] [36] [65] (mech., elect & software) [49] [71] legacy or country to country [110] physical and software [88] [81] [74] [55] [104] - *Variables interdependence* [29] - *In requirements* [29][130] [110] [98] [68] - *In functions* [68] - *Types of variability* [37], in time(i.e. time variant or invariant) [21] - *Sources of variability* [37] - *In components and parts (physical and software)* [81] [55] - *Heterogeneity due to time scales, interface protocols, imposition of constraints* [70] - *System resources geographical heterogeneity* [19] [53]	- Abstraction level / Level of aggregation or detail [3] [4] [95] [59] [10] [61] [11] [13] [14] [97] [62] [15] [35] [36] [63] [64] [37] [85] [38] [98] [16] [129] [106] [17] [81] [39] [49] [70] [19] [22] [43] [121] [100] [101] [102] semantic gaps/heterogeneity in data models at project and team level [94]*,* in requirements [110] - Goals of the models / information (depend on the process phase) [49] [19] - Overlaps and heterogeneity in design information and data (models) [1] [30] [57] [6] [59] [60] [10] [11] [47] [13] [14] [80] [62] [35] [36] [63] [64] [38] [98] [106] [17] [81] [49] [40] [50] [67] [52] [19] [22] [43] [24] [100] [23] HW & SW[104] frameworks [127] system specification, simulation is mono-disciplinary [83] [35] [49], gap between system design and system analysis [129], redundant information due to product families [115], differences between HW and SW (PLM & ALM) [55], different time scales or product structures in each process phase because it facilitates, but they lack of contextual relations [26]*, all information is separated/partial view of the complete information,* [121], non-compatibility of	- Heterogeneity on the design tools and model libraries [23] [122] [101] [102] [94] - Semantic gaps/overlaps in tools and databases (distinguish between local and common concepts in internal data models/structures/ specifications, communication protocols, platform specifications (i.e. PLM vs BIM) etc.) [130] [1] [3] [4] [6] [58] [79] [59] [60] [13] [14] [15] [35] [98] [49] [70]*, tools often apply a range of different terms and/or modelling structures to describe a given data model concept leading to semantically heterogeneous models* [94]*,* (tools have to cover a complex patchwork of different views in terms of functionality and model semantics through the product lifecycle) [26] [43]*)* (missing common platforms for modelling complex systems) (for example statics or dynamics analysis [10]), and in transformation capabilities of artifacts (i.e. generating simulation or code from models) [84], ambiguous operational semantics in SysML [85], gaps and overlaps in contents of tools (models) [38], gap between system design and system analysis [129], per discipline in their data models [19], trade-off between standard-based, tool/vendor-specific and locally defined ontologies [23], SW and HW notations, semantics[104] - Versioning concepts and links of baselines [1] [3] [47] [121] between tools that manage as-designed and as-built statuses [35], PLM and ALM systems [55] [26] - In backbone infrastructure and enterprise architecture concepts [56] - Multiplicity or overlapping of tools (when a person is in several projects, might need to use the same tool for different content, on different roles) [9] [98] - In tool management and use, i.e. tools are often managed by a dedicated IT support team and that can be outside the development team [15] - In time to create or test/simulate the design in the tool [36] - In abstraction levels or levels of fidelity of the tools [36] [38] [49] - Related to (data) exchange standards and Heterogeneous data models [16] [44] [92] - Differences on technical platforms [92] - Variety of process modelling tools [69]

	- *Tension with individual processes for process standardization* [22] - Different language, models, techniques and tools from business, information application and technologies [127] - In relationships to the government (aerospace vs construction) [24] - In competition type (aerospace is expertise, construction is costs) [24] - Type of customer base (narrow or highly diversified) [24]	[50] [21] (with tools) [49] [66] [67] [51] [69] [53] [22] [74] [115] [26] [75] [112] [24] Wide allocation of expertise [25] - Experience level [4] [7] [126] [9] [11] [12] [47] [14] [80] [97] [35] [63] [17] [39] [51] [21] [53] [115] [25] [112] New people [66] (with tools) [49] - Geographical [30] [2] [3] [6] [58] [8] [9] [15] [20] [19] [71] [53] [25] [26] [75] *globalization* [78] - Temporal [3] [6] [8] [9] [40] - In other general skills [14] social [50] - Overlapping of duties [52] - *Priorities and perspectives (managers and designers)* [69], - *Time horizons (managers and designers)* [69] *in tools* [54] - *Organization culture and habits which influence collaboration behaviour of individual persons* [25] [26]	lifecycle (engineering vs production) [26] , with manufacturing [101] - *With respect to the involved engineering disciplines and stakeholders* [19] - *Abstraction level of process (micro-level, meso-level, and macro-level) / point of view* [66] [93] - Iteration perspectives / roles (i.e. iteration to progress the design; iteration to correct problems or implement changes; and iteration to enable coordination within a process, or between a process and its context) [66] - Process models' syntax and context [69] - *Varying process standards (incl. diff. in coverage, scope and precision)* [86] - *Heterogeneity of the results of the process ("processes are characterized by a certain uniqueness", it is not about finding an equal result but a customized solution)* [22] - *Tension with organization for process standardization* [22] - *Heterogeneity of the phases of the development cycles* [92]	dimensions in projects (time-to-market, time-to-serial production, time-to-profit) and these should be balanced [91], project scope [92] - Projects are different depending on the developed project and other issues [91]	- Overlapping of duties of discipline [52] - In Technologies and techniques and paradigms used (ones more advanced in specific aspects than others) [105] [1] [33] [33] [83] [12] [96] [80] [62] [63] [65] [66] [50] [19] [22] (physical systems vs software) [68] data science and manufacturing have different developments [48] - In phenomena and dependencies (physical are multiple couples physical phenomena, vs software with state space, bugs, connectivity) [68] - Dev. Time and iterations/ time for creating or simulating model [36] [68] or testing, length feedback cycles (hardware >> code) [40] - *In difficulty of estimating non-functional properties (such as cost)* [68] - Of the interfaces depending on the discipline [49] - Priorities [69] - *Structure depths heterogeneity (geometry in mechanical, integrated circuits in electronics, source code in software development)* [22] - *Differences in mind-sets of the disciplines* [55] - Different language, models, techniques and tools from business, information application and technologies [127] - Information models can be used for discipline-specific or discipline spanning use [100] - Modelling principles differ in the symbols even if they model similar components (mechanical vs. electrical) [100] - Information models from mechanical and information technology don't usually overlap much [100] - Time resolution (for various disciplines' multi-physics models incorporate dynamics with different frequency ranges) [23] - Theories and notations of disciplines tailored to their needs [104]		information models [91], heterogeneity of models [101] - General heterogeneity of data, information and models (structures) [4] [6] [79] [60] [11] [47] [13] [14] [80][35] [38] [98] [106] [68] [19] [43] [100] *(scientific and experience-based heuristic)* [52], *different formalisms with possible semantic misalignments* [70], *terminology in models* [70] - *Various information / viewpoints needs depending on stages of lifecycle (i.e. production, manufacturing, marketing, supply chain)* [33] [34] [47] [50] [25] [26] [43] [114] - *In time of creating or testing/ simulating information / model* [36] - *Varying quality of information* [53] - Consistency of information [ADD SOURCES] - Traceability of information [55] [104] - Different dominant dimension in project as well as perspectives from discipline need different information models (when is a product information model complete for each viewpoint)? [91]	- Heterogeneity of interfaces between engineering tools [19] - In priorities and perspectives of the tool users, i.e. during system implementation, *managers focus on achieving organizational performance whereas the individual focuses on simplifying the operations of tasks* [54] - Jargon, terminology and vocabulary related to the tooling [114] [94] - Different language, models, techniques and tools from business, information application and technologies [127] [94] - Inconsistency management techniques for tools are still missing work for mechatronics because most assume that it is similar than software development [100] - Different dominant dimensions in projects need different capabilities in tools (they are used for different tasks) (i.e. different information structures) [91] [94] - Most engineering tool suites are vertically integrated by have limited support for integration across disciplinary boundaries [23] - Tension between the product and engineering-process-specific view of the systems' companies and the more generic view of tool vendors [23] - Time resolution (complex multi-physics models frequently incorporate dynamics with highly different frequency ranges) [23] - Difficult in the tools to separate concerns – they are bundled together and cannot easily be distinguished (Multiple viewpoint modelling is ignored by most of the UML tools) [124] - Different "dimensions" or types of tool integration, namely Control, Data, Platform, Presentation and Process Integration [131] - Variety of stakeholders that toolchains serve (application domain experts, project managers, support environment administrators, customers, standardization organizations, etc.) and how they put different demands on tools [131] - *Industries (such as CPS) are too heterogeneous for any single vendor to fully support* [101] - *Levels of abstractions of the tools (difficult to transform system models with higher level of abstraction to more concrete software engineering models)* [102] - *Toolchains can provide different coverage of the development process* [92] - *Variety of tools (simple tool to aid in manual work to a fully configurable and automatic). Even if the tool is automatic and advanced manual intervention can still be needed for rapid application development* [92]
Uncertainty	- Environmental conditions for Organization survival, adaptation and profitability (disruptive technologies, market pressure, competitive pressure, innovation, etc.) [109] [4] [123] [5] [34] [83] [97] [52] [71] [53] [75] [91] - Uncertainty about return of investment of new strategies and associated risks [97] [35] [106] [17][118] [39], about migrating to new tool platforms (for instance cloud) due to cost, time and technical uncertainties [27] - Uncertainty about organizational skillsets [39] - Uncertainty about required resources (Need to use unplanned resources) [76]	- Knowledge gaps (unknown unknowns) [96] [39] [68] - Related to Design risks and with design decisions/ trade-offs [96] [35] [63] [129] [48] [39] [51] [68] [52] [70] [19] [74] number of involved engineering decisions [53] (incl. uncertainty in not having enough information to make a decision and also uncertainty in achieving functional requirements) [21], hard to balance integration over functionality, simplicity, cost, criticality [73] - Uncertainties/ Doubts related to change in paradigms [97] - Risks and uncertainties related to interactions/ human language uncertainty (misunderstandings / multiple possible interpretations) [35] [38] [118] [65] - Various levels (and definitions) of uncertainty (uncertainty, imprecision, inconsistency, inaccuracy, indecision and instability, uncertainty has to do with lack of definition, lack of knowledge or lack of trust in knowledge) [72] - *complexity of communication in the project team triggers unpredictable impacts* [54]	- Ambiguity about the terms process and process model [69] - *Ambiguity in understanding of the process (people understand them in different ways)* [69] - *Ambiguity on the relationship between models and their targets* [69] - Uncertainties / risks in process (innovativeness, novelty, creativity, i.e. conceptual design, lack of definition, non-standard practices) [109] [10] [96] [81] [88] [39] [66] [52] little predictability , high possibility for disruptions [93] [66], *none-routine / unfamiliarity* [52] - *In priorities and decisions regarding them (inappropriate priority decisions generate rework)* [72], [54], creativity and innovation [22] [54], *development process is uncertain and unpredictable* [76] - Process performance uncertainty (Fears on migrating existing processes, missing validation, unknown impact, different forms of designing not accounted for) [97] [39] - Process emergence (to produce a more efficient and consistent design among the numerous technical data used in an engineering cycle [15]; every process is unique, new activities are typically discovered during projects; and decisions must often be based on inadequate or preliminary information [66] - Processes are trial-and error [21] - Various levels (and definitions) of uncertainty (uncertainty, imprecision, inconsistency, inaccuracy, indecision and instability, uncertainty has to do with lack of definition, lack of knowledge or lack of trust in knowledge) [72] - Incomplete and uncertain data and information leads to iterations [22] - In process models (like any model) there is uncertainty. In choosing a model you also choose an the uncertainty associated with the model's characteristics and range [112]	- In Priorities [57] [72] [91] - In Logistics [57] - Risks [14] [36] [39] [68] [52], *technical* [91], market [91], budget [91], schedule [91] - *Uncertainty about required resources (Need to use unplanned resources)* [76] - *In responsibilities definition* [76]	- In whether the discipline can offer the required functionalities (matchmaking) [6] [7] - Boundaries between the disciplines are often ambiguous [100]	- General system uncertainties and risks (regards to shape, material, boundaries, unknown external forces, risks, etc.) [109] [59] [11] [14] [63] [64] [68] [52], [53] [91] [23] *uncertainty in achieving functional requirements* [21] - Use of new technologies, innovation [117] [96] [68] [52] [21] [23] - Trade-offs & design decision-making [109] [95] [6] [58] [79] [8] [59] [61] [14] [96] [35] [63] [37] [129] [48] [66] [40] [51] [68] [19] [21] [104], number of involved engineering decisions [53] - In relationship of system with end users / customer needs (requirements) [109] [7] [14] [35] [36] [118] [88] - Emergence [10][96] [63] [68] emerging qualities of SoS like interconnectivity and interoperability [73] - *In terms of verification quality, testing capabilities* [14] [97] [110] [63] [106] [23] [104] (being able or not to test the [whole] real system, to only do non-destructive testing) [96] [36] [64] [98] [16] [48] [39] [40] [67] - *System boundaries and external environments and systems* [35] [63] [68] - Various levels (and definitions) of uncertainty (uncertainty, imprecision, inconsistency, inaccuracy, indecision and instability, uncertainty has to do with lack of definition, lack of knowledge or lack of trust in knowledge) [72]	- Degree of uncertainty or correctness of the design information [48] [66] [53] [23] (requirements) [118] [49], knowledge and data (in models or documents) [4] [96] [35] [63] [64] [38] [39], the risks [14] [36] level of detail of the models [10] [64] [49], data and information arise only in the context of development, so to work assumptions are needed [22] - Fears on migrating existing data [97] - Product information is structured according to geometry/ assembly which is does not exist in early phases nor is stable [49] - Which information should be in the system-level and which in the sub-system (Detailed) level [49] - Various levels (and definitions) of uncertainty (uncertainty, imprecision, inconsistency, inaccuracy, indecision and instability, uncertainty has to do with lack of definition, lack of knowledge or lack of trust in knowledge) [72] - In product models (like any model) there is uncertainty. In choosing a model you also choose an the uncertainty associated with the model's characteristics and range [112]	- Tooling uncertainty (i.e. in model/simulation extensiveness (simulating several components together) [13] [14], realism, level of detail, fidelity [10] [61] [96] [49] , uncertainty of the estimation of system models behavior [84] [38] [39], and cost-benefit of having too detailed models/simulations [49] - Uncertainty to migrate to newer platforms (for instance cloud) because of technical difficulties [27] risks for transitioning from closed-source to open source-tools to follow research advances especially with legacy models [132]
Change / Dynamics	- Of Organizational structure / Structure of workforce [123] [8] [97] due to cross-company collaboration [25], reorganization [41], *relationship between organizational and operational structure in the development departments and the development process* [22] - Of organizational conventions, strategies, operations and policies [130] [119] [3] [123] [7] [8] [12] [47] [80] [36] [106] - Culture (new ways of thinking/ strategies/vision) [96] [80] [97] [62] [63] [16] [106] [17] [118] - Of market trends / conditions such as new boundary conditions and requirements	- Changes in position (permanent or temporary, i.e. joining a new project) [9] [97] [40] - Group dynamics in teams and dynamics of their competences [3] [8] [14] [97] [35] [40] - Changes in the nature of the professional roles and responsibilities, autonomy mentality, etc. [130] [97][90] - Internalization of the working environment [53] - Changes required in way of thinking or working / cultural [17] (using systems thinking and later systems engineering) [61] or with respect to SoS [96], or joint efforts (safety and SE) [80], new paradigms like MBSE [97] [62]	- Changes in processes (new ways of working, practices and paradigms, new boundaries, etc.) [130] [32] [6] [7] [12] [47] [96][80] [97] [88] [81] [50] [67], IoT [42], cross-company collaboration [25] - Moving from coordinated product development to collaborative product development [25] - Dynamic nature of design process and its ability to deal with changes [29] [109] [57] [8] [59] [83] [15] [36] [65] [48] [39] the process sequence is unpredictable, because tasks are progressively concretized and adjusted as work proceeds [66]; *design tasks need to be rescheduled and reprioritized frequently during product development* [72], progressive nature of design process [72]; Relationship of individuals with processes is often fluid and unreliable [69], requires flexibility /adaptation to situational conditions [88] [81] [39] [68] [91] (interactions and jumps) [93], non-routine designing is difficult to automate [66], *state of the workflows of the system* [18], non-linearity of design process [22] different methods required in different stages (systems thinking vs systems engineering) [61], workflow needs to be flexible [114], required fluidity of the	- In Priorities [57] [8][72] - In Logistics [57] [8] - Unforeseen changes [83] [47]	- Changes or combinations in discipline paradigms including new boundary conditions and requirements (i.e. electric mobility) [107], discipline paradigms (BIM vs PLM) [130] [4] [5] system specification, simulation is mono-disciplinary [83] [84] [50] [53], need to combine product and production design processes [53], physical level abstraction is no longer enough [74]; mechatronics [100] - In Technologies [5] [8] [47] [50] [19] [53], more software [100], disruptive technologies [27] - Dominance of one discipline above others [49] - A variation in one domain may lead to a (performance) variation in another [49] [21] [22] [74]	- (Unforeseen) design changes and their consequences (Likelihood and impact of change propagation) [111] [28] [117] [113] [30] [1] [58] [10] [83] [11] [47] [68] [53] [74] [76] (cost money and delays) [21], change propagation due to interdependencies in design [72] - Changes/Dynamics of / in technology [105] [117] (AI, Big data) [96] [48] [19][53] disruptive technologies [27] - Parameters are incrementally defined and frozen [72] - Dynamic nature of system under design (dynamic features, requirements, system evolution, the speed of change and functional behaviour, timing behaviour [21], i.e. large, time varying, heterogeneous data – time series behaviour and data exchanges between (sub)systems) [65] [53] [111] [116] [133] [119] [57] [7] [8] [59] [10] [83] [11] [14] [96] [36] [63] [98] [16] [65] [81] [39] [49] [68], dynamics of production systems at	- Dynamic nature of design knowledge, information and data ([unforeseen] design changes) [133] [119] [57] [3] [57] [59] [10] [108] [47] [80] [62] [36] [63] [37] [65] [76] consistency [74], flexibility, iterative changes, documentation flexibility needed [114], changes applied by designers to artifacts in tools [122] - Probability of change initiation [72] - Information transformation/ exchange between design disciplines/ process stages/baselines (for example when using systems thinking and systems engineering) [61] [62] [35] [110] [65]	- Unforeseen design changes - Cross-company collaboration required changes in working environment, tools, IT systems, technical infrastructure [25] - Effects of dynamic nature of design (change in design needs to be applied in a multitude of IT systems, etc.) [1] [3] [58] [50] [67], more accurate tools and analyses are gradually brought to bear as the increased confidence in the design justified the increased efforts [72], *Tools need to be flexible in terms of iterative changes to the design (drawings)* [114], interoperability is needed because of reusing data, consistency and traceability as well as for early validation of systems [77] - Effects of dynamic nature of processes and organizations (changes in them apply to IT systems paradigms as well) [130] [12] [96] [15] [50] [67], requirements that must be managed by IT infrastructure emerge from of data and information requirements of development processes (for each activity) - *Changes in information technology and software (globalization, diversity, social networks, big data, software technology changes)* [8] [96] [50] [71] - *Changes in infrastructure* [97]

	[107] [123] [5][4] [34] [8] [7] [83] [11] [47] [36] [90] [71] [55] [25] [75] like electric mobility, sustainability/ environmental [51], big data [96], IoT [42], self-driving cars, drones, [49], digitalization [93] , industrial revolutions/business models [50], globalization [53] [27], agile, servitization, mass-personalization [67], change from only product or part manufacturer to system provider [42], Uncertainty of time-to-market, market demand, fluctuation of demand [53], CPS, IoT, Industry 4.0 [23] - Economic conditions [123] [8] - Technology availability and speed of development [53] - Of relationships with supply chain and customers [130] [35] [51] [19] [53] - Legislative and regulations changes [36] [20] [19] [53] - Internationalization of the working environment [53] - Organizational requirements for the process and information flows are different (per process stage, per project, etc.) [54]	- Dynamics of Reframing the problem [90] - (Unforeseen or progression) design changes [83] [53] [74] A variation/change in made by the work of one person may lead to a (performance) variation in another's work (business process, decision making, policy making) [49] [21] [72] [22] [74] in SoS [73], *changes cause work overload* [76] - Design changes arise because stakeholders' knowledge has not been integrated to the design process [91]	process [27], construction industry needs process to be flexible (government demands this) [24], difference between as-planned and as-performed activities [24] - *Level of standardization of process* [56] [59] [84] [22] - (Unforeseen) design changes [108] solved by iterations [91] (cause cost overruns and problems with time schedule, rework) [21] [76] - Dynamics of Reframing the problem [90] - Uncertainty in work division, who is expected to perform what and in which phase [50] - Process considerations related to (new) system key drivers/aspects pressures [50] (CPS, IoT, SoS) [67] - Target of the process model can shift during the course of the design without the model itself changing [69] - *A change/delay in one activity has the potential to influence other activities* [41] [18] - *Probability of change initiation* [72] - *Parallelization (dynamics) of design steps transitions between domain specific and interdisciplinary work* [22] - *Change between synthesis and analysis of several characteristics of data* [22] - *Relationship between organizational and operational structure in the development departments and the development process* [22] - *Lack of agility to update* [121] - Design changes arise because stakeholders' knowledge has not been integrated to the design process [91] -		- Parallelization (dynamics) of the work that is domain specific and the work that is interdisciplinary [22]	design time and runtime [19], system behavior is no longer derived from the sum of individual partial functions, i.e. more diverse [22] - More and more is demanded in terms of capabilities, features, functions [96] [48] [39] [66] [50] [67] [51] [68] [99] [71] [53] [27] more intelligence incl. learning and decision-making abilities [49], run-time flexibility of production systems [19],[78] IoT [42], PSS AND IoT [25][26], *multi-causation, multi-variability, multi-dimensionality, interdependence with the environment* [26], *balance complexity with performance requirements and quality* [82], *a lot more software and electronics* [100] - Dynamics of Reframing the problem [90] - Probability of change initiation [72] - Systems scalability [23]	- Cross-company collaboration requires changes in information and data models and has security and IP concerns [25] - *Information flows are versatile, subject to different organizational requirements in different stages of product design and development* [54] - *Information models scalability* [23] - *Universal Modelling languages have to respond rapidly to changes in the product line, engineering processes* [23] - *Various data sets managed at the different process stages* [77] - *Creating common data models for integration of disciplines typically includes continuously collecting new knowledge of the related application domains* [94]	- *Changes in versioning and storage concepts/ paradigms (need to store at element level and not at file level)* [62] - *Longer Longevity of the systems sets new requirements for IT infrastructure in terms of longevity, stability and scalability* [20] - *Tools need to support the specific characteristics and requirements of more demanding and complex systems* [26], *more interdisciplinary systems (CPS)* [23] - *For exchange standards there is* inflexibility in accommodating changing information requirements in supporting multiple and sometimes conflicting viewpoints of different types of users [43] - *Changes in industrial practices have required standards to take tools and tool integration more and more into account* [131]
Limitations / Pressures / Anxieties	- Monetary resources ((re-) design costs, investment capacity and profit) [107] [120] [2] [4] [123] [33] [57] [58] [126] [60] [83][108] [11] [12] [47] [14] [80] [62] [35] [36] [63][64] [38] [16] [106] [17] [65] [88] [39] [93] [90] [40] [51] [20] [18] [72] [73] [43] [75] [114] [27] [134] [100] [23] [101] expensive revisions and redesign [21] [101] (economic success) [52] - Tooling resources [123] [7] [10] [12] [47] [38] [16] [106] [17] [39] [19] [72] [114] [23], The choice of a particular approach or technology for tooling could make perfect sense to one stakeholder, while another discounted it outright. So it is difficult to prioritize/ trade-off business models, stakeholders, etc. [131], *because tool vendors cannot satisfy the complete end-to-end tooling, companies need to develop, maintain, and integrate in-house proprietary tools to remain competitive, but this is fragile, intractable, error prone and extremely costly* [101] - Time resources [89] [29] [31] [57] [2] [6] [79] [128] [59] [60] [108] [12] [47] [14] [97] [36] [63] [38] [16] [106] [17] [65] [88] [39] [40] [51] [72] [53] [43] [75] [104], shorter development processes [25][82] [114][127] [134] [100] *(i.e. to document* [119]) [4] [123]) (to learn a modeling language or solve underlying issues [10] [11] [97][35] [16] to implement and refine new methods [61]), time-to-market [93], missing original schedules [21] - Human resources [123] [10] [12] [14] [80] [97] [38] [16] [106] [17] [118] [21] [72], availability of technical skills [92] - Quality pressures [97] [35] [63] [38] [16] [17] [39] [93] [40] [51] [86] [55] [75] [100] - Market pressure [97] [63] [16] [17] [67] [51] [20] [53] [55] [55] [43] [114] safety [23] time to market [76] - Competitive pressure [120] [2] [109] [4][123][126] [83] [108][17] [49] [68][86] [20][99] [41] [53] [55] [75] [114] [27] [100] - Fragmentation of the industrial/ market sector [133] [56] [53][24] - Resistance/ Support to organizational change [119] [133] [5] [6] [10] [47] [80] [97] [62] [35] [63] [16] [106] [17] [118] - Organization psychological climate and motivational aspects [135] [5] [60] [12] [97] [35] [16] [118]	- Knowledge and training limitations [96] [97] [16] [17] [50] [75] - Related to extraction tacit knowledge [30] [29] [32] [119] [135] [4] [123] [33] [5] [34] [7] [79] [9] [60] [61] [47] [68] [52] [19] [22] [25] [43] extract properties [14] [35] [63] change from system thinking to systems engineering [61], design engineers create mental product models, from which explicit models are derived, however much of the knowledge contained in the mental model remains implicit as it is not needed to complete the explicit model [112] - Experience/ expertise level and/ or skills / competences [116] [117] [107] [4] [123] [7] [9] [60] [10] [14] [96] [80] [63] [106] [48] [39] [51] [52] [21] [22] [74] [115] [43] [75] experienced design engineers can build powerful mental models [112], availability of the technical skills (affects also selection of tools) [92] - Individual Psychological climate and motivation aspects *(Perception of the organizational (IT) environment such as support, trust and care)* [128] [135] [5] [7] [9] [60] [10] [12] [47] [97] [63] [16] [106] [118], cross-company collaboration trust [25], bad experiences and frustration with (new) strategies, approaches, tools, etc. [97] [35], motivation to share or provide knowledge [25] - Memory capacity [29] [63] [52] (short and long term) [68] - *Information management capability* [128] [96] [80] [63] [38] [118] [22], [75] [45] *manage multiple digital definitions of the same product* [43] - *Decision making ability and support* [135] [109] [58] [10] [83] [35] [129] [51] [68] [70] [22] *design space exploration* [104] - *Quality/Stability of relationships/ communication (among works/ stakeholders)* [128] [57] [5] [7] [9] [61] [35] [63] [65] [40] [68] *across companies* [25], *complexity of communication in the project team triggers unpredictable impacts* [54] - Cognitive capacity [29] [30] [128] [7] [14] [96] [63] [75] (to understand/learn and use modelling languages) [11] [12] [16] [106] [17], to use tooling [35] [16] [106] [17], to learn a design methodology [21], cognitive overload [68], intellectual capacity [21], to handle the information [74]	- Intangibility of the process, it has to be expressed in terms of process models [69] - Immaturity or lack of quality of the existing/ defined (engineering and/or management) processes, methods and ways of working [133] [120] [109] [33] [57] [6] [7] [8] [126] [60] [10] [83] [61][11] [14] [96] [84] [80] [97] [15] [35] [36] [64] [98] [106] [118] [118] [65] [48] [88] [81] [49] [71] [26], processes are lacking in industry [91] rethought or adapted (persistence of a sequential process) [67] most process models set out only what should be undertaken not why or how [51], high level prescriptive processes are only part of the picture [69], *current limitations of concurrent engineering* [27] - Limitations related to practical adoption of processes or combinations thereof [119] [5] [61] [11] [12] [97] [63] [17] agile in engineering [36] [48] (from research [39]) agile methods difficult to use in large scale systems, but principles might? [67] , today's CPS practices are ad-hoc [68], V-model *From research and literature processes are clear, but adaptation or implementation of them in practice is not well-established* [42], frameworks are generally complex to implement [127] - Limitations from tacit/explicit process information of the company (like processes description/ representation) [89] [29] [30] [120] [4] [123] - Pressure on design efficiency and quality (short lead times- or do more activities in the same time) / Time-consumption of process [29] [47] [16] [106] [39] [53] [75] [114] [127] - *Related to considerations/ tensions from other stages of lifecycle (i.e. production, manufacturing, marketing, supply chain, in service, i.e. big data, how to incorporate that to process) or with management processes* [83] [47] [14] [96] [35] [85] [90] [26] and other stakeholders [63], partnerships with other organizations at other stages of lifecycle [25] - *Related to appropriateness and/or customization of the process to a defined purpose/scope / focus (i.e. approaches only apply to a specific part or do not fit the purpose /application/ system domain)* [84] [97] [17] [39] [66] not multidisciplinary (ignores software, etc.) [65] [50] [71] [74] [42] detailed design is very domain specific [67] [22] mono-disciplinarity of processes (improvements) [40] software and physical [68] [74] [42] MBSE embraces hard but not soft parts [63], agile [48], only for conceptual and not for detailed design (FBS, MBSE, etc.) [88] [81] [67], *methods for modelling and simulation with emphasis on the system view* [49] and for dealing with conceptual and preliminary design [49], too abstract [93] [69], total change of paradigm of design as a discipline [90], too abstract to be applied [66], *process must be adapted to the problem* [52], *difficult to apply to day-to-day activities* [69], *do not depict the reality of what designers experience* [69], *traditional methods fail to consider integration perspective in SoS* [73], *processes are generic and need to be adapted or tailored to specific conditions* [22], *framework solutions aimed at representing business user's concerns and have no direct link to IT implementation* [127] - (Managers complain that) processes do not get followed or that designers do not carry out the tasks in the right way, at the right order or at the right time [69] - *Related to evaluation in industry (such as MBE, MDE, MBSE)* [16] [39] - *Related to process support from management* [39] - *Related to presentation of the process/ methodology (inadequate advertisement, representation, addresses knowledge not application, no differentiation along design principles* [39] - *Difficulty to trace and analyse* [40] - *Process analysis methods inappropriateness (usually don't go beyond generic stakeholder, tasks, and input/output artefact identification, plus they tend to not consider individual disciplines, interfaces between workgroups and overall collaboration)* [40] - *Limitations related to the representation of processes (languages)* [40] - *Creators vs. outsiders: Process models feel like something the designers are excluded from (opaque to them), so they are not perceived as proposals but demands. The designers create their own models and see them among themselves as requests* [69] - *Limited usefulness of the process standards* [86]	- Resources (monetary, tools, time, people) [57] [68] [72] [134] - *Processes are anchored in definition of departments, team structures and responsibilities, which shape release decisions and actual project/process execution* [22] - *Difficult to link tools to project management flows* [124] - *Project scope (can influence also choices of tools and methods)* [92]	- Discipline boundaries, traditions [5] [7] [10] [12] [68] (i.e. not parametrizing the CAD models) [62] [26] - Design optimization needs be multi-objective (optimization separately within each discipline will not result in optimum system design) [49], total change of design and engineering as a discipline [90] - Knowledge of the entire system does not equal the sum of knowledge from corresponding domains [49] - Interdisciplinary work is connected to several bodies of knowledge [22]	- Technological/ Technical limitations [53] - System design key driver/aspects pressures and their monitoring [90] [14]: quality [6] [38] [52], size [39] [67] [21], weight [67], costs [52], materials [107] [109] [21], cost [112], degree of modularity [28] [29] [59] [97] [98], *performance* [28] [47] [14] [52], *sustainability* [83] [50] [20] [26] , *environmental* [47], interoperability [26], *predictability* [26], quality[36][36][36][36][92][91][90][89][88][87][86][85][84][83][82][81][80][79][78][77][76] [75][74][73], reliability [36] [38] [16] [65] [49] [67] [26], availability [16], maintainability [16], safety [38] [16], security [14] [63] [16] [26], dependability [36] [26], adaptability [63], resilience [63], interoperability [98], virtualization [98], decentralization [98], real-time capabilities [98], tolerances [39], robustness [49], user-friendliness [99], usability [99], ergonomics [99], robustness [22], *flexibility* [22] - Incidental vs essential complexity [68] - Unintended and accidental behavior of the system [68] - In the needs from relationship between product and production systems [53] - Related pressures to incorporate other parts of the lifecycle of systems, such as production, maintenance, etc. [25] [26]	- Quality and accuracy of information (ambiguity) [89] [119] [56] [4] [95] [79] [14] [24] (required level o[106] f detail of models [10], necessary characteristics [14]), in requirements [36], functionality [110] - Information models are lacking in industry [91] - Interdisciplinary information models (like PLM) lack dynamism and configurability for different types of industries and businesses [91] - Tacit/explicit design information/ rationale (i.e. design decisions/ motivation, requirements) [32] [30] [119] [120] [135] [2] [4] [123] [33] [95] [7] [79] [8] [60] [83] [61] [47] [14] [35] [63] [66] [40] [68] [19] [25] [112] *extracting encapsulated knowledge and information of CAD models* [26], owning to time pressure information quality can decrease and insufficient quality causes complexity [53] - Regarding to quality/ extensiveness existing information /knowledge organization [10] [61] [47] [96] [62] [35] [68] [41] [25] - Tacit/ explicit design / discipline context / semantics [119] [135] [33] [95] [79] [8][60] [83] [61] [47] [13] [80] [35] [63] [85] [68] [19] [25] [112] (parametrized vs un-parametrized CAD) [62] *annotate knowledge in artefacts* [40] - *With regards to brownfield development (legacy systems)* [119] [12] [62] [37] [27] - *Information (cyber) security / confidentiality / intellectual property, access management* [2] [3] [10] [35] [36] [16] [40] [78] - *Limited* practical applicability (search, retrieve, store, visualize, reuse, etc.) of data, information and knowledge (structures) [95] [79] [60] [13] [17] - Tension between expressiveness (natural language) and fixed and formalized (machine readable) [60] [61] - Information and formats overload of designer/ stakeholders (either for visualization or processing) [60] [108] [14] [96] [35] [63] [118] [68] [41] [25] visualizations and models are seldom in practice because of the very large entangled graphs that cannot be understood [121] Related to the data from other parts of the lifecycle (with manufacturing, maintenance) [25] (big data) [96] [63] example: lack of maturity/ support for other activities (program or product management) of modelling languages [106] - Knowledge of the entire system does not equal the sum of knowledge from corresponding domains [49] Related to the limits expressiveness to describe artifacts and the way the interact [70] - *Analysis models are disconnected from parent design models (CAD/CAE)* [82] - *Difficulty to find suitable models to represent the necessary information,*	- In Tooling functionality, capabilities, current usage, support and/or maturity (usability, maintenance aspects, scalability, visualization, personalization, knowledge capturing) [1] [135] [120] [95] [58] [7] [8] [9] [59] [60] [10] [108] [15] [110] [63] [16] [106] [17] [51] [68] [70] [20] [25] for example analog tools or COTS software can help but it is limiting [11], what can be simulated/tested [96], in automated simulation from models (lack of libraries, etc., functionality and expressiveness of simulation) [84], general immaturity of tools [97] [62], tools to assess the level of complexity of functional interfaces between domains as well as evaluating vulnerability of a single functional interface and its effect on robustness [49], user-friendliness [25], use of scenarios [50], that support multidisciplinarity [67], current tools for workflow modeling are not enough to represent all the variables that affect efficiency such as available resources [18], *companies have requirements that may not be fulfilled by PLM and this causes islands of data due to lack of full integration between data sets, lack of collaborative possibilities for multiple users* [124], - Artifacts Consistency issues (consistency from small vs. large number of artifacts (i.e. consistency and transformation approaches are generally done in pairs) [70], [77] - Artifacts Transformation issues (i.e. non-determinism of bi-directional transformations) [70] - Tool Vendor restrictions/limitations [3] [60] [15] [19] s[44] election of PLM vendor can be a difficult decision due to varying capabilities [43], vendors provide limited integration, primarily of their own tools, with a few cross-vendor integrations for particularly dominant tools (i.e. DOORS, Matlab, Word or Excel) [23], lack of tool's user support materials or interaction (incl. interactive guidance mechanisms) are rarely supported [124], *because tool vendors cannot satisfy the complete end-to-end tooling, companies need to develop, maintain, and integrate in-house proprietary tools to remain competitive, but this is fragile, intractable, error prone and extremely costly* [101], vendor lock-in [104] - Limitation to interact/ exchange/ connect with other (commercial/ legacy) tools/ databases [119] [120] [3] [95] [34] [79] [9] [60] [10] [11] [12] [13] [96] [84] [80] [97] [15] [35] [63] [98] [16] [129] [106] [17] [65] [48] [68] [25] [24] [92], *legacy* [27], many tools are offered as a fixed application that cannot be extended, customized or integrated with different platforms [124], *difficulty for creating adapters to integrate tools* [122], *some transformation require the transformation author to understand the transformation language as well as the details of use and semantics for both the source and the target models (in a real enterprise there will be experts within one domain that can understand one modelling language but rarely both* [45], *too loose coupling between engineering tools* [78], - *Limitations to extract internal data from tools: lack of available interfaces, for various reasons tools don't provide mechanisms for accessing their internal data* [122], *for MBSE and safety, plus Support for interoperability standards is still limited so it is necessary to rely on proprietary methods to integrate tools* [77], some information in the native model representation cannot be exported [132] - Limitations in data structures / architectures of tooling [1] [120] [58] [79] [10] [96] [15] [35] [25] - In Information Technology infrastructures [133] [128] [135] [5] [10] [38] [106] [48] [68] [53] [25] bringing various simulators together is more than just integrating the tools [96] [49], (separation of) databases [80] [63], Services and APIs [62], lack of common platform where all activities of working and sharing can be carried out [27] - Limitations related to the existing data interoperability and exchange standards [98] [16] [65] [43] [44] lack of specific workflow for OSLC [78] - Limitations regarding tool scope/ focus/ domain / or genericity vs. specificity (technical, social, etc.) [5] [7] [9] [110] [129] [106] for example analog tools or COTS software can help but it is limiting [11] or tools too discipline-specific and not for synthesis [11], simulation tools normally on a specific domain [84] [35] but not specific enough for a program [15], SysML only used in design and validation phase (not implementation – too generic) [85]. If tool is too generic than has a lot of challenges in expectations (usability, content, etc.) [38], software source code integration with mechanical and electrical [65], simulations at discipline level are different than those at system level [49], unclear use cases of tools for engineering process [40], *lack of flexibility* [50], support for more than one discipline [67], lack of support for concept design and mostly for

	- Knowledge of the industry environment and society/customer needs [123] [8] [20] [19] - Legal and ethical issues: Protection of intellectual property / know-how/ confidentiality [120] [2] [3] [36] [64] [16] [50] in cross-company cooperation [25], compliance with regulations [97] [16] [52] [20] [19] [53] [55], ethical concerns [50] - Limitations of organizational and management practices (related to organizational knowledge and experience) [123] [33] [5] [7] [35] [106] - Limitations related to current organizational strategies and vision, policies i.e. (preferences for a specific a modelling language, tooling, frameworks, processes, etc.) [10] [12] [14] [35] [63] [106] [17] [68] multidisciplinarity [83] [80], current company boundaries [35] - *Workforce age* [47] - *Inappropriate (false or too high) expectations of strategy, approach, tool, etc.* [97] [17], acknowledging the benefits but prefer to avoid the costs and organizational impacts needed for cooperation and collaboration in SoS [73] - *Organizations rely on vendors to tell them what software tools (PLM) are best for them and how to implement them* [134] - *Tools require strategies and structures to be implemented* [134] - *Bureaucracy in processes* [76] - Related to tool investments, successful examples of seamless end-to-end tool integrations are rare, even after massive investments of companies [23]	- *Other General personal abilities, intuition* [52], *creativity* [52] *visualization capacity* [29] [14] [63], *human behaviour* [30] [119] [5], [10], - *Limitations by other organizational stakeholders (managers, project leaders)* [11] [12] [47] [80] [97] [63] [106] [17] [118] [39] [41] - *Limitations by legislative barriers (confidentiality, intellectual property)* [36] in cross-company collaboration [25] - *Tendency to focus on the more technical aspects and neglect process aspects* [40][97] - *Bounded rationality and biases* [68] - *Attention span of the brain* [68] - *Not following design process and/or principles (starting with solution before defining functional requirements, incomplete or incorrect specifications, poor decomposition, coupled designs, etc.)* [21] - *Preference for functionality over integration* [73] - *Overload (or feeling lost) from engineers/designers : with interfaces* [25], *with activities and amount of work* [76] - *Designers time is taken up with something other than technical work* [114] - *When implementing a new tool, organizations discover they actually need a process definition, procedures, etc.* [134] - *For tooling interconnection, it is necessary that the expert understands the tools within his/her domain but also that of the target to integrate, and that is in practice rare (a skill that is hard to find is someone who understands the two domains)* [45]	- *Other variables that can affect processes' efficiency like availability of resources* [18] - *Hierarchical structure of processes is idealized* [19] - *Processes of system design and implementations are mostly company-specific/ ad-hoc (starting with solutions, incomplete or wrong specifications, etc.)* [21] companies often are used to handle their structure management per design project with less or no predefined set of methods, processes or IT tools [42], processes are voluntarily informal and seldom documented [24] - Processes also take into account product evolution, effects of lessons learned and historical data and information of the company [22] - *Processes are anchored in definition of departments, team structures and responsibilities, which shape release decisions and actual project/process execution* [22] - *Engineers are already overwhelmed by process instructions* [25] - *.Product development process is often executed in the "black box" of the development team (information flows are versatile)* [54] - *Humans still prefer to communicate via emails, sharing data with resources like pen-drives, etc.* [27] - *For process models Hard to balance between modelling need and comprehension efficiency* [121]				*because there a large plethora of existing product models* [112] *Product models sometimes don't have the elements to store important information from the mental models, related to tacit/explicit* [112] - *Models of the product are mostly tool-based and methods for using them, if existing at all, focus on how to use the tool* [112] - Information consumption needs from other stages of the product lifecycle and non-technical departments (*Difficulties to visualize (consume) models/information without needing installation the software tool)* [44] - Product development deals with untouchable flow of data, information and knowledge, work in progress is mainly constituted of information and data stored in computers [76]	later detail phases [51], PLM are predominantly mechanical engineering focus + monolithic approaches with one single leading system for PLM do not work anymore [26], *only expose collections of parameters and not the structural model information* [45], *in the transformation, engineering organization want to retain ownership of their pieces of analysis software (each software owner wants some degree of control and ability to refactor approaches to deal with problems) this adds friction* [45] - In tool's practical application/ adoption [5] [34] [7] [60] [11] [12] [35] [43] - Sense of confidence/user-friendliness/ preference for certain tools [128] [6] [35] [16] - Tension between access and other permissions to the tools (by engineers and other stakeholders) and resulting (annoying) constraints to the user [12] [62] [35] [78], IT security considerations, access management [40] [50] [78] - Poor tool performance (quality of the transformations, exchanges, and automated generations is poor) [16], updating traceability information takes too long which affects acceptance of the tool in practice [78] - Integration costs [16] - Concepts and methods guiding systems engineering do not address adequately the use of IT-Systems [55] - The development of tools and IT-system's to support collaboration often tends to focus on technical requirements and the humans get lost in the process of digitalization [25] - Unreliability and missing information within IT systems [25] - Tooling and formats overload for engineers [25] *("learn yet another tool")* [75] - *Use or introduction of tools without clear processes and roadmaps* [43] - *Most communication tools are used as separate services and are disconnected from development environments* [75] - *Most engineering collaboration tools are based only on CAD models for design reviews and discussions (difficult to integrate CAE)* [82] - *When a CAD model is converted from one system to another, the modelling history is lost and the designer intent cannot be recovered* [82] - *The degree of adaptation of tools depends on the interaction condition and appropriation among the technology, the organization and the groups.* [54] - *Limitations of information consumption due to the tooling* [44] - *The use of some tools did not affect the time spent on technical work as much as it should have* [114] - *Limitation due to standalone tooling systems required to be installed individually on premise, the associated maintenance and operational costs of applications hosted by different providers* [27] - *Idealized view of tools to support human expert activity has not been reached* [121] - *Because of the need to use heterogeneous tool components we experience "tyranny of tools", when not the design problems but the available tools dictate the abstractions that designers should use in problem solving* [23] - *Complex product lines span too many technical areas for single vendors to fully cover* [23] - *A significant part of the companies' design flow is supported by in-house tools that are proprietary and capture high-value design intellectual property* [23] - *Tools are selected covering only parts of the process: Often design flows consist of islands of integrated tool sub-chains, bridged by various ad-hoc, semi-automated, or manual stopgaps; which impost costs (of additional manual transformations, additional work of guarding divergence of multiple representations, and forgone analysis opportunities)* [23] - *Difficulty to compare tools based on practitioners needs* [124] - *Lack of support for model simulations, well-formedness rules checking and formal verification requirements at the same time* [124] - *Some features are more ignored than others (generate code from models is supported but not generate models from code)* [124] - *Lack of support for automation of modelling asks (few tools support scripting languages)* [124] - *Difficult to link tools to project management flows* [124] - *Scalability or flexibility of existing solutions: All-in-one solutions from one tool vendor have many drawbacks (dependency on a vendor, costly and time consuming, not for all activities), point-to-point integration is not scalable, current limitations of open interoperability specifications* [78] - *High-level issues and the contexts in which tool integration can be found are treated indifferently in research, much of this focuses only on technology* [131] - The choice of a particular approach or technology for tooling could make perfect sense to one stakeholder, while another discounted it outright. So it is difficult to prioritize/ trade-off business models, stakeholders, etc. [131] - Research revolves around open-source frameworks, and the use of commercial tools is neglected, while they are the norm in the industry [132] - Overhead imposed by solutions in terms of even with small changes requires a lot of work [132] - Difficulty of the need to handle multiple users at the same time [132] - Development teams are not clear on the concept of tool chains, and lack of knowledge on how to create a fundamental model against which a set of tools can be arranged in a chain setup [92] - *Lack of support for the complete lifecycle* [92] - *Selection of tools is affected by availability of technical skills and project scope* [92] - *Difficulties in model co-simulation based on connectors for the models and algebraic loops* [103]

References

[1] R. Mordinyi, E. Serral, and F. J. Ekaputra, "Semantic Data Integration: Tools and Architectures," in *Semantic Web Technologies for Intelligent Engineering Applications*, Springer International Publishing, 2016, pp. 181–217.

[2] S. Zhang, "Knowledge Management Environment for Collaborative Design in Product Development," in *IFIP Advances in Information and Communication Technology*, 2019, vol. 567, pp. 475–480.

[3] S. E. Schapke, J. Beetz, M. König, C. Koch, and A. Borrmann, "Collaborative data management," in *Building Information Modeling: Technology Foundations and Industry Practice*, Springer International Publishing, 2018, pp. 251–277.

[4] W. qiang Li and Y. Li, "A study on the collaborative management method of product design cycle knowledge," *Multimed. Tools Appl.*, vol. 77, no. 21, pp. 27877–27894, Nov. 2018.

[5] M. Levy, I. Hadar, and I. Aviv, "A requirements engineering methodology for knowledge management solutions: integrating technical and social aspects," *Requir. Eng.*, vol. 24, no. 4, pp. 503–521, Dec. 2019.

[6] M. Sabou, O. Kovalenko, F. J. Ekaputra, and S. Biffl, "Semantic Web Solutions in Engineering," in *Semantic Web Technologies for Intelligent Engineering Applications*, Springer International Publishing, 2016, pp. 281–296.

[7] S. Sukumaran and K. Chandran, "The unspoken requirements - eliciting tacit knowledge as building blocks for knowledge management systems," *Lect. Notes Bus. Inf. Process.*, vol. 224, pp. 26–40, 2015.

[8] T. McDermott, J. Zentner, and N. Bollweg, "Thinking from the system architecture: A better methodology for capture of collaborative knowledge," in *2014 International Conference on Collaboration Technologies and Systems (CTS)*, 2014, pp. 472–473.

[9] L. C. Anderson and C. A. Kieliszewski, "A Socio-Technical Perspective in Support of Information Sharing for Diverse Teams in Today's Workplace," in *Proceedings - 2017 IEEE 3rd International Conference on Collaboration and Internet Computing, CIC 2017*, 2017, vol. 2017-Janua, pp. 416–420.

[10] W. K. Vaneman and R. Carlson, "Model-based systems engineering implementation considerations," in *SysCon 2019 - 13th Annual IEEE International Systems Conference, Proceedings*, 2019.

[11] C. Tschirner, L. Bretz, R. Dumitrescu, and J. Gausemeier, "Applying Model-Based Systems Engineering for Product Engineering Management Concepts for Industrial Application," *IEEE Syst. J.*, 2015.

[12] T. Amorim, A. Vogelsang, F. Pudlitz, P. Gersing, and J. Philipps, "Strategies and Best Practices for Model-Based Systems Engineering Adoption in Embedded Systems Industry," in *Proceedings - 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEIP 2019*, 2019, pp. 203–212.

[13] U. Shani and H. Broodney, "Reuse in model-based systems engineering," *9th Annu. IEEE Int. Syst. Conf. SysCon 2015 - Proc.*, pp. 77–83, 2015.

[14] A. Garro and A. Tundis, "Modeling of system properties: Research challenges and promising solutions," in *1st IEEE International Symposium on Systems Engineering, ISSE 2015 - Proceedings*, 2015, pp. 324–331.

[15] V. Arnould, "Using model-driven approach for engineering the system engineering system," in *2018 13th System of Systems Engineering Conference, SoSE 2018*, 2018, pp. 608–614.

[16] G. Liebel, N. Marko, M. Tichy, A. Leitner, and J. Hansson, "Model-based engineering in the embedded systems domain: an industrial survey on the state-of-practice," *Softw. Syst. Model.*, vol. 17, no. 1, pp. 91–113, Feb. 2018.

[17] M. Chami, A. Aleksandraviciene, A. Morkevicius, and J.-M. Bruel, "Towards Solving MBSE Adoption Challenges: The D3 MBSE Adoption Toolbox," in *28th Annual INCOSE International Symposium*, 2018, pp. 1463–1477.

[18] A. Pla, P. Gay, J. Meléndez, and B. López, "Petri net-based process monitoring: A workflow management system for process modelling and monitoring," *J. Intell. Manuf.*, vol. 25, no. 3, pp. 539–554, Oct. 2014.

[19] S. Biffl, A. Lüder, and D. Winkler, "Multi-Disciplinary Engineering for Industrie 4.0: Semantic Challenges and Needs," in *Semantic Web Technologies for Intelligent Engineering Applications*, Springer International Publishing, 2016, pp. 17–51.

[20] R. Curran, X. Zhao, and W. J. C. Verhagen, "Concurrent engineering and integrated aircraft design," in *Concurrent Engineering in the 21st Century: Foundations, Developments and Challenges*, Springer International Publishing, 2015, pp. 571–605.

[21] N. P. Suh, "Challenges in Designing and Implementing Large Systems (Overcoming Cost Overruns and Missed Project Schedules)," in *Axiomatic Design in Large Systems*, Springer International Publishing, 2016, pp. 273–309.

[22] K. Paetzold, "Product and Systems Engineering/CA* tool chains," in *Multi-Disciplinary Engineering for Cyber-Physical Production Systems: Data Models and Software Solutions for Handling Complex Engineering Projects*, Springer International Publishing, 2017, pp. 27–62.

[23] J. Sztipanovits, T. Bapty, X. Koutsoukos, Z. Lattmann, S. Neema, and E. Jackson, "Model and Tool Integration Platforms for Cyber-Physical System Design," *Proc. IEEE*, vol. 106, no. 9, pp. 1501–1526, Sep. 2018.

[24] C. Boton, L. Rivest, D. Forgues, and J. Jupp, "Comparing PLM and BIM from the product structure standpoint," in *IFIP Advances in Information and Communication Technology*, 2016, vol. 492, pp. 443–453.

[25] W. Wang, P. Lünnemann, S. Neumeyer, H. Hayka, and R. Stark, "Product Development in Collaborative Networks-An Expert View on Current Challenges and Future Trends," in *17th Working Conference on Virtual Enterprises (PRO-VE)*, 2016, pp. 302–312.

[26] D. Gerhard, "Product lifecycle management challenges of CPPS," in *Multi-Disciplinary Engineering for Cyber-Physical Production Systems: Data Models and Software Solutions for Handling Complex Engineering Projects*, Springer International Publishing, 2017, pp. 89–110.

[27] S. Sharma, F. Segonds, N. Maranzana, D. Chasset, and V. Frerebeau, "Towards cloud based collaborative design – analysis in digital PLM environment," in *IFIP Advances in Information and Communication Technology*, 2018, vol. 540, pp. 261–270.

[28] T. Habib, "Multidisciplinary Product Decomposition and Analysis Based on Design Structure Matrix Modeling," in *7th World Conference on Mass Customization, Personalization, and Co-Creation (MCPC)*, 2014, pp. 409–423.

[29] G. B. Loureiro, J. Carlos, E. Ferreira, P. Henrique, and Z. Messerschmidt, "Design structure network (DSN): a method to make explicit the product design specification process for mass customization," *Res. Eng. Des.*, vol. 31, pp. 197–220, Feb. 2020.

[30] P. E. Arduin, J. Le Duigou, M. H. Abel, and B. Eynard, "Knowledge sharing in design based on product lifecycle management system," in *Smart Innovation, Systems and Technologies*, 2015, vol. 35, pp. 507–517.

[31] G. Bruno, D. Antonelli, R. Korf, J. Lentes, and N. Zimmermann, "Exploitation of a Semantic Platform to Store and Reuse PLM Knowledge," in *IFIP Advances in Information and Communication Technology*, 2014, vol. 438, no. PART 1, pp. 59–66.

[32] T. Boge and K. Falk, "A3 Architecture Views-A Project Management Tool?," in *29th Annual INCOSE International Symposium*, 2019, pp. 971–987.

[33] A. Mayr *et al.*, "Towards a Knowledge-Based Design Methodology for Managing the Complexity in the Integrated Product and Process Development of Electric Motors," in *Advances in Production Research*, Springer International Publishing, 2019, pp. 112–125.

[34] G. Bruno, E. Traini, and F. Lombardi, "A Knowledge-Based System for Collecting and Integrating Production Information," in *IFIP Advances in Information and Communication Technology*, 2019, vol. 568, pp. 163–170.

[35] M. Pasquinelli, L. Molina-Tanco, A. Reyes-Lecuona, and M. Cencetti, "Extending the System Model," in *Dynamics of Long-Life Assets: From Technology Adaptation to Upgrading the Business Model*, S. Grösser, A. Reyes-Lecuona, and G. Granholm, Eds. 2017, pp. 169–189.

[36] J. Denil, R. Salay, C. Paredis, and H. Vangheluwe, "Towards agile model-based systems engineering," *CEUR Workshop Proc.*, vol. 2019, pp. 424–429, 2017.

[37] C. Dumitrescu, P. Tessier, C. Salinesi, S. Gérard, A. Dauron, and R. Mazo, "Capturing Variability in Model Based Systems Engineering," in *Complex Systems Design & Management*, Aiguier. M., F. Boulanger, D. Krob, and C. Marchal, Eds. Springer International Publishing, 2014, pp. 125–139.

[38] H. Hick, M. Bajzek, and C. Faustmann, "Definition of a system model for model-based development," *SN Appl. Sci.*, vol. 1, no. 9, pp. 1–15, Sep. 2019.

[39] C. Melville, X. T. Yan, and L. Gu, "TiV-model-an attempt at breaching the industry adoption barrier for new complex system design methodologies," in *Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and Their Designers*, Springer International Publishing, 2016, pp. 41–57.

[40] L. Kathrein, A. Lüder, K. Meixner, D. Winkler, and S. Biffl, "Product/ion-Aware Analysis of Collaborative Systems Engineering Processes," in *Security and Quality in Cyber-Physical Systems Engineering*, Springer International Publishing, 2019, pp. 151–185.

[41] C. Haskins and K. S. Ruud, "Systems engineering: Making people talk!," in *Disciplinary Convergence in Systems Engineering Research*, Springer International Publishing, 2017, pp. 1081–1093.

[42] M. Bitzer, M. Vielhaber, and J. Kaspar, "Product lifecycle management - How to adapt PLM to support changing product development processes in industry?," *Proc. Nord. Nord. 2016*, vol. 1, 2016.

[43] A. K. Behera *et al.*, "Sharing design definitions across product life cycles," *Res. Eng. Des.*, vol. 30, no. 3, pp. 339–361, Jul. 2019.

[44] A. Katzenbach, S. Handschuh, R. Dotzauer, and A. Fröhlich, "Product lifecycle visualization," in *Concurrent Engineering in the 21st Century: Foundations, Developments and Challenges*, Springer International Publishing, 2015, pp. 287–318.

[45] B. Cole and K. Dinkel, "Multidisciplinary model transformation through simplified intermediate representations," in *IEEE Aerospace Conference Proceedings*, 2016, vol. 2016-June.

[46] T. R. Browning, "Design Structure Matrix Extensions and Innovations: A Survey and New Opportunities," *IEEE Trans. Eng. Manag.*, vol. 63, no. 1, pp. 27–52, Feb. 2016.

[47] J. D'Ambrosio and G. Soremekun, "Systems engineering challenges and MBSE opportunities for automotive system design," *2017 IEEE Int. Conf. Syst. Man, Cybern. SMC 2017*, vol. 2017-Janua, pp. 2075–2080, 2017.

[48] Y. Li, U. Roy, and J. S. Saltz, "Towards an integrated process model for new product development with data-driven features (NPD 3)," *Res. Eng. Des.*, vol. 30, no. 2, pp. 271–289, Apr. 2019.

[49] P. Hehenberger, T. J. Howard, and J. Torry-Smith, "From mechatronic systems to cyber-physical systems: Demands for a new design methodology?," in *Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and Their Designers*, Springer International Publishing, 2016, pp. 147–163.

[50] M. V. Pereira Pessôa and J. M. Jauregui Becker, "Smart design engineering: a literature review of the impact of the 4th industrial revolution on product design and development," *Res. Eng. Des.*, vol. 31, no. 2, pp. 175–195, Apr. 2020.

[51] M. Marchesi and D. T. Matt, "Application of Axiomatic Design to the Design of the Built Environment: A Literature Review," in *Axiomatic Design in Large Systems*, Springer International Publishing, 2016, pp. 151–174.

[52] W. E. Eder, "Engineering Design: Role of Theory, Models, and Methods," in *An Anthology of Theories and Models of Design*, Springer London, 2014, pp. 197–217.

[53] R. Mehr and A. Lüder, "Managing Complexity Within the Engineering of Product and Production Systems," in *Security and Quality in Cyber-Physical Systems Engineering*, Springer International Publishing, 2019, pp. 57–79.

[54] C. C. Wu, C. F. Ho, W. H. Hung, and K. H. Kung, "PLM usage behavior and technology adaptation," in *Communications in Computer and Information Science*, vol. 473, Springer Verlag, 2014, pp. 76–91.

[55] A. Deuter, A. Otte, M. Ebert, and F. Possel-Dölken, "Developing the Requirements of a PLM/ALM Integration: An Industrial Case Study," in *Procedia Manufacturing*, 2018, vol. 24, pp. 107–113.

[56] Y. Chen and J. Jupp, "BIM and Through-Life Information Management: A Systems Engineering Perspective," in *Advances in Informatics and Computing in Civil and Construction Engineering*, Springer International Publishing, 2019, pp. 137–146.

[57] A. Zouari, “Knowledge versioning dynamics during the design process in a concurrent engineering environment,” *Lect. Notes Mech. Eng.*, vol. 789, pp. 11–20, 2015.

[58] A. Al Khatib, D. Fleche, M. Mahdjoub, J. B. Bluntzer, and J. C. Sagot, “Preparation of CAD model for collaborative design meetings: Proposition of a CAD add-on,” *Adv. Mech. Des. Eng. Manuf. Lect. Notes Mech. Eng.*, pp. 861–870, 2017.

[59] F. Toepfer and T. Naumann, “Towards Cross-Linked Development of Highly Complex Products,” in *Proceedings of the 21st International Conference on Engineering Design (ICED17)*, 2017, vol. 2, no. August, pp. 279–288.

[60] J. Camba, M. Contero, and G. Salvador-herranz, “Implementation challenges of annotated 3D models in collaborative design environments,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8683, pp. 222–229, 2014.

[61] R. Cloutier, B. Sauser, M. Bone, and A. Taylor, “Transitioning systems thinking to model-based systems engineering: Systemigrams to SysML models,” *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 45, no. 4, pp. 662–674, Apr. 2015.

[62] M. Bajaj, B. Cole, and D. Zwemer, “Architecture To Geometry - Integrating System Models With Mechanical Design,” *Aiaa Sp. 2016*, pp. 1–19, 2016.

[63] A. M. Madni and M. Sievers, “Model-based systems engineering: Motivation, current status, and needed advances,” in *Disciplinary Convergence in Systems Engineering Research*, Springer International Publishing, 2017, pp. 311–325.

[64] R. Leroux, M. Pantel, I. Ober, and J.-M. Bruel, “Model-Based Systems Engineering for Systems Simulation,” in *Leveraging Applications of Formal Methods, Verification and Validation. Distributed Systems. ISoLA 2018. Lecture Notes in Computer Science*, T. Margaria and B. Steffen, Eds. Springer, Cham, 2018, pp. 429–448.

[65] C. Zheng, M. Bricogne, J. Le Duigou, and B. Eynard, “Survey on mechatronic engineering: A focus on design methods and product models,” *Adv. Eng. Informatics*, vol. 28, no. 3, pp. 241–257, 2014.

[66] D. C. Wynn and P. J. Clarkson, “Process models in design and development,” *Res. Eng. Des.*, vol. 29, no. 2, pp. 161–202, Apr. 2018.

[67] M. Bricogne, J. Le Duigou, and B. Eynard, “Design Processes of Mechatronic Systems,” in *Mechatronic Futures*, P. Hehenberger and D. Bradley, Eds. Springer International Publishing Switzerland, 2016, pp. 75–89.

[68] M. Törngren and U. Sellgren, “Complexity Challenges in Development of Cyber-Physical Systems,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10760 LNCS, Springer Verlag, 2018, pp. 478–503.

[69] M. Stacey, C. Eckert, and R. Hillerbrand, “Process models: plans, predictions, proclamations or prophecies?,” *Res. Eng. Des.*, vol. 31, no. 1, pp. 83–102, Jan. 2020.

[70] A. Cicchetti, F. Ciccozzi, and A. Pierantonio, “Multi-view approaches for software and system modelling: a systematic literature review,” *Softw. Syst. Model.*, vol. 18, no. 6, pp. 3207–3233, Dec. 2019.

[71] C. Weber, “Modelling Products and Product Development Based on Characteristics and Properties,” in *An Anthology of Theories and Models of Design*, Springer London, 2014, pp. 327–352.

[72] J. F. Maier, D. C. Wynn, W. Biedermann, U. Lindemann, and P. J. Clarkson, “Simulating progressive iteration, rework and change propagation to prioritise design tasks,” *Res. Eng. Des.*, vol. 25, no. 4, pp. 283–307, Jun. 2014.

[73] Y. Mordecai and D. Dori, “I5 : A Model-Based Framework for Architecting System-of-Systems Interoperability, Interconnectivity, Interfacing, Integration, and Interaction,” in *INCOSE International Symposium*, 2013, pp. 1234–1255.

[74] L. Horvath, “New method for enhanced driving of entity generation in RFLP structured product model,” in *Proceedings of the 2017 12th IEEE Conference on Industrial Electronics and Applications, ICIEA 2017*, 2018, vol. 2018-Febru, pp. 541–546.

[75] J. D. Camba, M. Contero, G. Salvador-Herranz, and R. Plumed, “Synchronous communication in PLM environments using annotated CAD models,” *J. Syst. Sci. Syst. Eng.*, vol. 25, no. 2, pp. 142–158, 2016.

[76] M. Rossi, L. Cattaneo, J. Le Duigou, S. Fugier-Garrel, S. Terzi, and B. Eynard, “Lean product development and the role of PLM,” in *IFIP Advances in Information and Communication Technology*, 2016, vol. 492, pp. 183–192.

[77] I. Vagliano, D. Ferretto, E. Brusa, M. Morisio, and L. Valacca, “Tool Integration in the Aerospace Domain: A Case Study,” in *Proceedings - International Computer Software and Applications Conference*, 2017, vol. 1, pp. 731–736.

[78] A. Leitner, B. Herbst, and R. Mathijssen, “Lessons Learned from Tool Integration with OSLC,” in *Communications in Computer and Information Science*, 2016, vol. 639, pp. 242–254.

[79] D. Ahlers, M. Mehrpoor, K. Kristensen, and J. Krogstie, “Challenges for information access in multi-disciplinary product design and engineering settings,” in *The 10th International Conference on Digital Information Management, ICDIM 2015*, 2016, pp. 109–114.

[80] Z. Huang, S. Swalgen, H. Davidz, and J. Murray, “MBSE-assisted FMEA approach - Challenges and opportunities,” in *Proceedings - Annual Reliability and Maintainability Symposium*, 2017.

[81] Y. Cao, Y. Liu, X. Ye, J. Zhao, and S. Gao, “Software-physical synergetic design methodology of mechatronic systems based on formal functional models,” *Res. Eng. Des.*, vol. 31, no. 2, pp. 235–255, Apr. 2020.

[82] V. Nosenzo, S. Tornincasa, E. Bonisoli, and M. Brino, “Open questions on Product Lifecycle Management (PLM) with CAD /CAE integration,” *Int. J. Interact. Des. Manuf.*, vol. 8, no. 2, pp. 91–107, Apr. 2014.

[83] M. Eigner, T. Dickopf, H. Apostolov, P. Schaefer, K.-G. Fai\s st, and A. Ke\s sler, “System Lifecycle Management: Initial Approach for a Sustainable Product Development Process Based on Methods of Model Based Systems Engineering,” no. July, pp. 287–300, 2014.

[84] M. Nikolaidou, G. D. Kapos, A. Tsadimas, V. Dalakas, and D. Anagnostopoulos, “Simulating SysML models: Overview and challenges,” in *2015 10th System of Systems Engineering Conference, SoSE 2015*, 2015, pp. 328–333.

[85] S. Wolny, A. Mazak, C. Carpella, V. Geist, and M. Wimmer, “Thirteen years of SysML: a systematic mapping study,” *Softw. Syst. Model.*, vol. 19, no. 1, pp. 111–169, Jan. 2020.

[86] A. Jakjoud, M. Zrikem, C. Baron, and A. Ayadi, “SysPEM: Toward a consistent and unified system process engineering metamodel,” *J. Intell. Manuf.*, vol. 27, no. 1, pp. 149–166, Mar. 2016.

[87] I. Westphal, M. Freitag, and K. D. Thoben, “Visualization of interactions between product and service lifecycle management,” in *IFIP Advances in Information and Communication Technology*, 2015, vol. 460, pp. 575–582.

[88] C. Zheng, P. Hehenberger, J. Le Duigou, M. Bricogne, and B. Eynard, “Multidisciplinary design methodology for mechatronic systems based on interface model,” *Res. Eng. Des.*, vol. 28, no. 3, pp. 333–356, 2017.

[89] C. A. Shajahan, N. Firoz, S. Mohammed, V. R. Pramod N, and Ashim, “Managing Iterations in Product Development Process Using Dependency Structure Matrix,” *J. Phys. Conf. Ser.*, vol. 1355, p. 12024, 2019.

[90] P. E. Vermaas, “Design Methodology and Engineering Design,” in *Engineering Identities, Epistemologies and Values*, Springer, Cham, 2015, pp. 147–159.

[91] S. P. Leino, L. Jokinen, J. P. Anttila, and A. Pulkkinen, “Case study on engineering change management and digital manufacturing,” in *IFIP Advances in Information and Communication Technology*, 2015, vol. 467, pp. 591–600.

[92] S. Imran, M. Buchheit, B. Hollunder, and U. Schreier, “Tool chains in agile ALM environments: A short introduction,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, vol. 9416, pp. 371–380.

[93] C. Hollauer, L. Becerril, N. Kattner, D. Weidmann, N. Chucholowski, and U. Lindemann, “Adaptable mechatronic engineering design processes: Process reference model and methodology,” in *Smart Innovation, Systems and Technologies*, 2017, vol. 65, pp. 597–607.

[94] R. Mordiny, D. Winkler, F. Waltersdorfer, S. Scheiber, and S. Biffl, “Integrating heterogeneous engineering tools and data models: A roadmap for developing engineering system architecture variants,” in *Lecture Notes in Business Information Processing*, 2015, vol. 200, pp. 89–107.

[95] C. Hennig, A. Viehl, B. Kämpgen, and H. Eisenmann, “Ontology-based design of space systems,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9982 LNCS, pp. 308–324.

[96] B. P. Zeigler, J. W. Marvin, and J. J. Cadigan, “Systems engineering and simulation: Converging toward noble causes,” in *Proceedings - Winter Simulation Conference*, 2019, vol. 2018-Decem, pp. 3742–3752.

[97] A. Vogelsang, T. Amorim, F. Pudlitz, P. Gersing, and J. Philipps, “Should i stay or should i go?: On forces that drive and prevent mbse adoption in the embedded systems industry,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, vol. 10611 LNCS, pp. 182–198.

[98] L. Berardinelli, A. Mazak, O. Alt, M. Wimmer, and G. Kappel, “Model-Driven Systems Engineering: Principles and application in the CPPS domain,” in *Multi-Disciplinary Engineering for Cyber-Physical Production Systems: Data Models and Software Solutions for Handling Complex Engineering Projects*, Springer International Publishing, 2017, pp. 261–299.

[99] O. Bielefeld, V. Sizikov, and N. Schlüter, “Research of the Possibilities for Using and Linking TRIZ Methods with Systems Engineering,” in *IFIP Advances in Information and Communication Technology*, 2019, vol. 572, pp. 174–186.

[100] D. Weidmann *et al.*, “Product models in mechatronic design: Literature analysis on the interdisciplinary character of product models,” in *PICMET 2017 - Portland International Conference on Management of Engineering and Technology: Technology Management for the Interconnected World, Proceedings*, 2017, vol. 2017-Janua, pp. 1–7.

[101] J. Sztipanovits, T. Bapty, S. Neema, L. Howard, and E. Jackson, “OpenMETA: A model- and component-based design tool chain for cyber-physical systems,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, vol. 8415 LNCS, pp. 235–248.

[102] G. Macher, E. Armengaud, and C. Kreiner, “Integration of heterogeneous tools to a seamless automotive toolchain,” in *Communications in Computer and Information Science*, 2015, vol. 543, pp. 51–62.

[103] N. Amálio, R. Payne, A. Cavalcanti, and J. Woodcock, “Checking sysML models for co-simulation,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 10009 LNCS, pp. 450–465.

[104] P. G. Larsen, J. Fitzgerald, J. Woodcock, C. Gamble, R. Payne, and K. Pierce, “Features of integrated model-based co-modelling and co-simulation technology,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, vol. 10729 LNCS, pp. 377–390.

[105] S. Diagne, A. Coulibaly, and F. De Bertrand De Beuvron, “Complex product modeling based on a Multi-solution eXtended Conceptual Design Semantic Matrix for behavioral performance assessment,” *Comput. Ind.*, vol. 75, pp. 101–115, Jan. 2016.

[106] T. Huldt and I. Stenius, “State-of-practice survey of model-based systems engineering,” *Syst. Eng.*, vol. 22, pp. 134–145, Jul. 2019.

[107] M. Wang, A. Höfer, and H. Friedrich, “Conceptual design of suspensions with integrated electric motors on the basis of DSM,” in *20 TH INTERNATIONAL DEPENDENCY AND STRUCTURE MODELING CONFERENCE*, 2018, pp. 105–114.

[108] S. Li, H. El-Mounayri, W. Zhang, B. Schindel, and J. Sherey, “Implementation of systems engineering model into product lifecycle management platform,” in *IFIP Advances in Information and Communication Technology*, 2016, vol. 467, pp. 601–608.

[109] L. Hou and R. J. Jiao, “Data-informed inverse design by product usage information: a review, framework and outlook,” *Journal of Intelligent Manufacturing*, vol. 31, no. 3. Springer, pp. 529–552, 01-Mar-2020.

[110] M. Schäfer, F. Bitsch, S. Weißleder, and F. Wartenberg, “Challenges for MBSE and PLE for legacy product-based system environments,” in *Proceedings of the 7th International Conference on Complex Systems Design and Management, CSD and M Paris 2016*, 2016, no. 131030, pp. 3–15.

[111] S. Li and L. Chen, “Identification of clusters and interfaces for supporting the implementation of change requests,” *IEEE Trans. Eng. Manag.*, vol. 61, no. 2, pp. 323–335, 2014.

[112] S. Matthiesen, P. Grauberger, F. Bremer, and K. Nowoseltschenko, “Product models in embodiment design: an investigation of challenges and opportunities,” *SN Appl. Sci.*, vol. 1, no. 9, pp. 1–11, Sep. 2019.

[113] O. Ezzat *et al.*, “A DSM Clustering Method for Product and Service Modularization,” in *IFIP Advances in Information and Communication Technology*, 2019, vol. 566, pp. 375–382.

[114] M. Huhtala, M. Lohtander, and J. Varis, “Product data management – Defining the used terms,” *IFIP Adv. Inf. Commun. Technol.*, vol. 442, pp. 387–396, 2014.

[115] F. Zeng, B. Li, P. Zheng, and S. S. Q. Xie, “A modularized generic product model in support of product family modeling in One-of-a-Kind Production,” in *2014 IEEE International Conference on Mechatronics and Automation, IEEE ICMA 2014*, 2014, pp. 786–791.

[116] V. Holley, M. Jankovic, and B. Yannou, “Physical interface ontology for management of conflicts and risks in complex systems,” *Concurr. Eng. Res. Appl.*, vol. 22, no. 2, pp. 148–161, 2014.

[117] S. Jung, O. Asikoglu, and T. W. Simpson, “A method to evaluate direct and indirect design dependencies between components in a product architecture,” *Res. Eng. Des.*, vol. 29, no. 4, pp. 507–530, Oct. 2018.

[118] B. A. Morris, D. Harvey, K. P. Robinson, and S. C. Cook, “Issues in Conceptual Design and MBSE Successes: Insights from the Model-Based Conceptual Design Surveys,” in *26th Annual INCOSE International Symposium (IS 2016)*, 2016, pp. 269–282.

[119] Z. Alexeeva, D. Perez-Palacin, and R. Mirandola, “Design decision documentation: A literature overview,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9839 LNCS, pp. 84–101.

[120] J. Stjepandić, W. J. C. Verhagen, H. Liese, and P. Bermell-Garcia, “Knowledge-based engineering,” in *Concurrent Engineering in the 21st Century: Foundations, Developments and Challenges*, Springer International Publishing, 2015, pp. 255–286.

[121] F. Noël and D. Dori, “Towards 3D visualization metaphors for better PLM perception,” in *IFIP Advances in Information and Communication Technology*, 2016, vol. 467, pp. 461–475.

[122] D. Ziegenhagen, A. Speck, and E. Pulvermüller, “Using Developer-tool-Interactions to Expand Tracing Capabilities,” in *Proceedings of the 14th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2019),* 2019, pp. 518–525.

[123] Y. Sen, “Knowledge as a Valuable Asset of Organizations: Taxonomy, Management and Implications,” Springer, Cham, 2019, pp. 29–48.

[124] M. Ozkaya, “Are the UML modelling tools powerful enough for practitioners? A literature review,” *IET Software*, vol. 13, no. 5. Institution of Engineering and Technology, pp. 338–354, 01-Oct-2019.

[125] J.-J. Urban-Galindo and S. Ripailles, “PLM at GROUPE PSA,” in *Product Lifecycle Management (Volume 4): The Case Studies*, Springer, Cham, 2019, pp. 29–50.

[126] C. A. Rivera, J. Poza, G. Ugalde, and G. Almandoz, “A Knowledge Based System architecture to manage and automate the electrical machine design process,” in *Proceedings of the 2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and their Application to Mechatronics, ECMSM 2017*, 2017, pp. 1–6.

[127] T. Vosgien, E. Rigger, M. Schwarz, and K. Shea, “A Federated Enterprise Architecture and MBSE Modeling Framework for Integrating Design Automation into a Global PLM Approach,” vol. 517, pp. 36–48, 2017.

[128] W. J. Kettinger, Y. Li, J. M. Davis, and L. Kettinger, “The roles of psychological climate, information management capabilities, and IT support on knowledge-sharing: An MOA perspective,” *Eur. J. Inf. Syst.*, vol. 24, no. 1, pp. 59–75, Jan. 2015.

[129] N. Albarello and H. Kim, “Application of an MBSE Approach for the Integration of Multidisciplinary Design Processes,” in *Complex Systems Design & Management*, Springer International Publishing, 2014, pp. 85–96.

[130] J. R. Jupp and M. Nepal, “BIM and PLM: Comparing and learning from changes to professional practice across sectors,” *IFIP Adv. Inf. Commun. Technol.*, vol. 442, pp. 41–50, 2014.

[131] F. Asplund and M. Törngren, “The discourse on tool integration beyond technology, a literature survey,” *J. Syst. Softw.*, vol. 106, pp. 117–131, Aug. 2015.

[132] A. Zolotas *et al.*, “Bridging proprietary modelling and open-source model management tools: the case of PTC Integrity Modeller and Epsilon,” *Softw. Syst. Model.*, vol. 19, no. 1, pp. 17–38, Jan. 2020.

[133] Y. Chen and J. Jupp, “Model-based systems engineering and through-life information management in complex construction,” in *Product Lifecycle Management to Support Industry 4.0. PLM 2018. IFIP Advances in Information and Communication Technology*, 2018, vol. 540, pp. 80–92.

[134] B. Koomen, “Set based PLM implementation, a modular approach to PLM process knowledge, management and automation,” in *IFIP Advances in Information and Communication Technology*, 2017, vol. 517, pp. 3–12.

[135] S. Liu and P. Zaraté, “Knowledge based decision support systems: A survey on technologies and application domains,” in *Lecture Notes in Business Information Processing*, 2014, vol. 180 LNBIP, pp. 62–72.