Proceedings of the 5th Indian International Conference on Industrial Engineering and Operations Management, Vellore, Tamil Nadu, India, November 6-8,2025

Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/IN05.20250047
Published: November 6, 2025

Enhancing the Drone Systems Architecture Using Model-Based Systems Engineering Approach

Ram Dayal Pandey
Ph.D. Research Scholar (Industrial Engineering and Management)
Department of Management Studies and Industrial Engineering
Indian Institute of Technology (ISM)
Dhanbad, Jharkhand 826004, India
22dr0187@iitism.ac.in

Prajwal Kayalkar
M. Tech. Industrial Engineering and Management)
Department of Management Studies and Industrial Engineering
Indian Institute of Technology (ISM)
Dhanbad, Jharkhand 826004, India
prajwalkayalkar12@gmail.com

Shikha Singh
Assistant Professor
Department of Management Studies and Industrial Engineering
Indian Institute of Technology, ISM
Dhanbad, Jharkhand 826004, India
shikhas@iitism.ac.in

Abstract

The development of the drone systems needs to be systemized in terms of the efficient operation and functional design. This study examines the use of the Architecture Analysis and Design Integrated Approach (ARCADIA) methodology in the Capella software to design a drone system. ARCADIA presents the method that breaks down the design work into four sequential stages such as operational analysis (OA), system analysis (SA), logical analysis (LA), and physical analysis (PA). Capella offers the possibility to develop models at every stage of the process. The OA stage is used to determine mission actors, activities, and situations that can be applied in the drone system. The SA stage entailed defining the system limits, distribution of operating roles between the drone system and external actors. LA performed an analysis of the drone by breaking down its functionalities into logical units. Furthermore, PA had a role in transformation of the program logic to the practical parts that could be implemented in real world application. This paper demonstrated that the model-based systems engineering (MBSE) is an effective methodology that can be applied in the design of a drone system, particularly ARCADIA. The model-based approach had a systematic representation of the system during the design phase. These phases divided the system development process into a sequence of steps that allowed a systematic analysis and started with the high-level mission objectives and was completed with the component definitions, including defined functionalities and communication protocols. The capabilities of Capella formed the basic operations of building models, as well as their analysis, and traceability. The study offers a crucial foundation for the researchers interested in designing the drones that my be apply in real-world situations and incorporating other design systems with such functions.

Keywords
MBSE, Drone system, Capella, ARCADIA.

1. Introduction
The evolution of drone technology allows various industrial sectors to experience remarkable changes with applications involving delivery services, aerial imaging, emergency search and environmental assessment tools (Xu et al., 2023). Unmanned Aerial Vehicles (UAVs) drones are used in various applications across many industries due to their ability to perform a variety of tasks and fulfill specialized functions. The global drone industry is projected to grow USD 17.00 billion in 2023 and USD 57.8 billion in 2028 (Markets and Markets, 2023), which highlights its increased importance. The design of advanced UAVs requires an organised engineering solution, which involves the whole system. Conventional paper-based approaches tend to be ineffective in dealing with the complex relationship between various fields of study in the creation of UAV which results to the occurrence of gaps in communication, design discrepancies, and the chance of higher risks in development. (Carolan et al., 2022). The challenges of complex systems engineering are increasing due to complex systems, changing industrial settings, constrained timelines, uncertain needs, constantly changing environments and users. The agile principles of software development can be applied to complex systems engineering particularly when it is used with MBSE. The fundamental strength of agility is the capability to provide a least feasible product during the initial phase of product development. This enables early assessment of the changing needs so that final system is in accordance with the real needs and creates the promised value. Furthermore, frequent customer feedback during development (relocation of defect correction to earlier phases) and effective allocation of resources also lead to decreased rework.

MBSE ensures accuracy and coherence by using models as a primary source of accuracy. These models promote the communication among engineering teams because they offer a common approach to design and development process. Moreover, they make it possible to automate such operations as model exchange and synchronization. Agile MBSE has three main benefits: reduced rework (working towards the same direction), quicker impact analysis (keeping up with the evolving needs), and demonstrating value consistently (giving clear progress reports to stakeholders). It is an effective combination of complicated engineering projects. Through the integration of agility and MBSE, systems engineering will be able to overcome the complications of the modern world and provide successful projects that can address changing requirements (Plazanet & Navas, 2023). The Requirements Functional-Logical-Physical (RFLP) framework becomes unique framework within MBSE, which contributes to structuring and organizing the modelling process. It divides the system into four major domains:

- The requirements domain captures the objectives and constraints that the system must fulfil. It outlines what the system must accomplish according to the stakeholder view.
- The system functionality involves in providing the evidence of certain behaviors and capacities in order to meet the requirements. The system operates for achieving its described objectives by the actions it undertakes within definite operational procedures.
- The logical domain displays the logical functioning of the system in terms of its functional relationships between the system components and the mode of operation. The logical domain is used as an interface that links functions to physical deployment.
- The physical domain lead to the development of systems in terms of hardware, software and other physical components that are related to the implementation of the systems. Physical domain also includes the deployment and construction of the system.

MBSE is a modelling approach that provides system development frameworks and the organizational structure offered by RFLP (Jing et al., 2023). MBSE is an analytical method where a single digital model acts as a dependable source that monitors system requirements and functionality, as well as how they are distributed across the design process (Dassault Systèmes, 2023). The application of MBSE within complex system development provides recent research findings on significant benefits, including enhanced communication, improved collaboration, early risk detection, and optimized system architecture design outcomes through informed design choices (Grieves and Weilkiens, 2022; INCOSE, 2020). MBSE is a new technology that is becoming popular among researchers in terms of its uses in research activities. Several researchers apply this technology to examine complex systems because of its popularity (Figure 1).

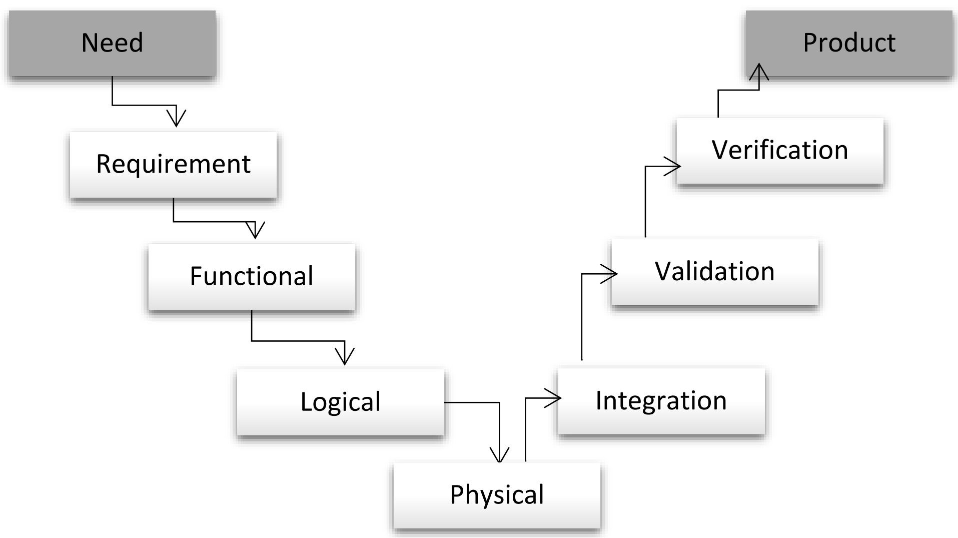

Figure 1 RFLP Framework

However, Lu et al. (2024) suggested that MBSE utilizes KARMA as its multi-architecture modeling language to create semantic designs for complete metal forming processes. Feng et al. (2024) proposed a method for building a stakeholder value network model to address stakeholders' needs and requirements in vertical farming. For this purpose, researchers used the MBSE modelling method ARCADIA with Capella software; Harkat et al. (2024) conducted a systematic literature review (SLR) on the security characteristics of CPS, particularly in the domain of industrial applications and smart grids; Jiang et al. (2023) discussed the development and application of the digital thread modelling method mainly in manufacturing processes. They also explored the simulation method that enable identification of possible defects and optimization of product structures; Garcia et al. (2022) conducted a SLR on digital twin technology, aiming to provide a better understanding of the learning opportunity presented by evolving Industry 4.0 digital twin ecosystems in manufacturing; Bao et al. (2024) conducted research on social product development in an automotive industry for validating two theoretical models. This study also highlights human-centric collaboration for new product development. Lins and Oliveira (2020) discussed the development of a standardized process for retrofitting outdated industrial equipment into Cyber-physical production systems (CPPS) to support Industry 4.0. Technological integration is presented based on a standardized platform complying with the reference architectural model of Industry 4.0 and covering all models of equipment. In this research work, authors explore the extent to which the requirement engineering methodology can be used with MBSE practices in the development of a drone system.

1.1 Objectives

This study offers the following research objectives:
RO1: To examine the traceability and alliance of ARCADIA approach in the design of drone system.

RO2: To employ requirement engineering with MBSE for modelling and managing system requirements during the design life cycle.

RO3: To employ MBSE techniques and tools to capture, analyze, and manage drone system requirements, ensuring traceability and effective design integration.

The purpose of this research work is to discuss the feasibility of the MBSE tool Capella. The study also seeks to present the way these tools can be used in terms of requirement engineering in the MBSE model of drone development. The research work explores how the MBSE can be used to streamline and enhance the design process of complex systems. The study also suggests that the creation of a central digital model which can become the ultimate source of information.

The remaining segment of this research paper is organized as follows: Section 2 discusses the existing academic literature, Section 3 summaries the research methodology employed in this study, Section 4 presents the results and discussion, Section 5 addresses the implications of the work, and this research work is concluded in Section 6.

2. Literature Review

MBSE is a systematic approach where a centralized model serves as the core for system requirements, analysis, verification, and validation activities throughout the system's entire lifecycle (INCOSE, 2023). The MBSE controls model, creates a central source for capturing, analyzing, and sharing all aspects of product specifications. Since, the traditional engineering methods mostly rely on human text documentation processes, managing the complexity of drone systems becomes difficult. This approach divides important system data among numerous documents, such as trade studies, descriptions of the system, interface control documents (ICDs), specifications, analysis reports, verification plans, and procedures (Carolan et al., 2022). MBSE provides standardised models for applying systems engineering concepts in an uniform manner. This method offers effective advantages during the complex system development process (MBSE IBM, 2023).

Implementation of the MBSE technique aids in achieving the sustainable development goals (Table 1). However, the organisations receive better energy conservation, less waste generation and better resource utilisation, which results in the development of sustainable system designs. Design and operational performance are optimised by MBSE, which results in low environmental impacts throughout the development of new projects. Since, fewer resources of energy are needed to translate the information, carbon emissions will drop, it leads for creating more sustainable and ecological friendly biosphere (IBM, 2023). MBSE is a multifaceted engineering process that uses models at every stage of design development to improve the tracking of documentation, increase consistency, while rendering system information easily accessible (Navas et al., 2018). Finding the optimal MBSE methodology for a project is still difficult due to the vast number of approaches (Weilkiens et al., 2016).

Weilkiens et al. (2016) have created an useful selection framework for assessing MBSE approaches (FEMMP) in order to help practitioners for making decisions. The FEMMP standardises the evaluation of MBSE technique, whereas most studies focus on specific areas such as adoption challenges and influences on the engineering discipline (Aigner & Khelil, 2019; Madni & Sievers, 2018). FEMMP provides a standardised approach relying on standard evaluation criteria to address this gap (Weilkiens et al., 2016). Vitech relies on its MBSE framework, STRATA, in order to focus on system design creation in multiple interconnected layers (Baker & Long, 2007). The framework which provides a detailed structure of systems engineering contains all the technical processes as detailed in the ISO/IEC/IEEE 15288:2018 standards (Vitech Corporation, 2018). The STRATA software includes a metamodel framework which allows proper recordings of the design elements so that it can be connected to its detection tools (GENESYS and CORE) (GENESYS and CORE) (Vitech Corporation, 2018).

Thales combined ideas from SysML (Systems Modelling Language) and UML (Unified Modelling Language) principles to create the integrated MBSE approach, ARCADIA (Voirin, 2017). Traceability of system development at various levels is ensured through automated information interchange (Roques, 2017). Traceability is the capacity to monitor the effects of system design choices made at one level on those at subsequent stages. The ARCADIA framework utilizes Capella, an open-source software whose model visulization capabilities allow for collaborative editing across several disciplines (Roques, 2017). MBSE utilizes a range of models to support system engineering activities. Different system descriptions emerge in written, tangible, mathematical, or rational forms, representing the system (Zhang et al., 2023). A system model can combine both computational and explanatory models to demonstrate different facets of the system, where a descriptive prototype outlines the requirements and is subsequently linked to a simulation prototype to evaluate system efficiency (Noguchi, 2023).

Table 1. MBSE Software tools (MBSE Tools, Incosewiki)

Tool	Vendor	License	Languages Supported	Comment(s)
Astah SysML	Astah	Commercial	SysML	
Cameo Systems Modeller	Dassault Systems	Commercial	SysML	Previously supplied by No Magic, who were acquired by DS in 2018.
Capella	PolarySys	Open Source	Proprietary (SysML-like)	Alternative modelling tool to Papyrus using the Arcadia Method.
ConceptDraw Diagram	CS Odessa	Commercial	SysML	A SysML' stencil' is provided for diagramming.
CORE 9	Vitech	Commercial		
Enterprise Architect	Sparx Systems	Commercial	SysML	
GENESYS	Vitech	Commercial		
IBM Engineering Systems Design Rhapsody	IBM	Commercial		Previously Rational Rhapsody.
Innoslate	Innoslate	Commercial	SysML	Browser-based SysML modelling.
Lattix Architect	Lattix	Commercial		Provides a view of software dependencies as a Dependency Structure Matrix.
Modelio	Modeliosoft	Open Source	SysML	Modelio uses a SysML Architect plugin.
Papyrus	PolarySys	Open Source		Polarsys is the Industrial Working Group of Eclipse.
PREEvision	Vector	Commercial		Embedded electrical/electronic modelling.
Software Ideas Modeler	Dušan Rodina	Commercial		
SCADE Architect	Ansys	Commercial	SysML	Aims to support engineers with SysML under the hood.
SysML Designer	Obeo	Open Source		Add-on for UML Designer.
Topcased	PolarySys	Open Source		Discontinued in favour of Papyrus.
Visual Paradigm for UML	Visual Paradigm	Commercial	SysML	UML modelling tool with SysML included.
Windchill Modeler	PTC	Commercial		Previously, Integrity Modeler & Artisan Studio was supplied by Atego, which PTC has acquired.

3. Research Methodology

MBSE offers a systematic and comprehensive method for designing the drone system, integrating functional requirements and components into a single model (Zhang et al., 2025). In contrast to the traditional document-based method, MBSE improves traceability, teamwork, and change management by integrating models and automating them (Ramirez et al., 2025). However, conventional techniques are sequential and less responsive to change. Therefore, MBSE provides a more valid and effective framework for managing the multidisciplinary and complex nature of existing drone systems.

ARCADIA comprises four primary levels adaptable to top-down, bottom-up, and iterative approaches. Each level develops deeper into the system design with specific objectives:

- Operational Analysis (OA) focuses on understanding what system users need to accomplish.
- System Analysis (SA) determines what the system must achieve for actors.
- Logical Architecture (LA) defines how the system will operate to meet expectations.
- Physical Architecture (PA) specifies how the system will be constructed.

The ARCADIA method divides the system architecture stage of the V-cycle into four steps, corresponding to Capella's four levels as described above (Thales, 2023). The V-model (verification and validation model) is an orderly approach to software development that concentrates on testing throughout the entire product lifecycle. It receives the term because it creates a simple V-shape when the development stages and the testing goes along with them are depicted. The V-model offers a precise development roadmap that reduces uncertainty and guarantees that every stage produces results that are clearly specified (Leroux et al., 2018).

Capella is an unique tool that applies the ARCADIA approach and assuring uniformity in the modelling level. ARCADIA encourages a perspective-based approach that incorporates four views enabling systematic examination of complex systems at all phases of growth. The default Capella perspective provides a full overview with the ability to create reports, update models, and search project components. Both perspectives are further subdivided into various views which outline specific details of the project. Critical factors include the diagram editor (to create and edit visual models), the outline (to view the model hierarchy) and the project explorer (to view the project structure). ARCADIA/Capella methodology is made up of several distinct processes, all of which contribute to the comprehensive analysis, design and deployment of a system.

The aim of OA phase is to specify mission actors, activities, and scenarios that are pertinent to the system. Important tasks include developing an operational architecture diagram to visually represent stakeholder needs and activities, identifying stakeholders and their needs, illustrating system capabilities with an operational capabilities diagram, and building operational entity scenarios to describe system behaviour in particular scenarios. SA seeks to assign functions to system components in order to meet operational needs. During this phase, the system's boundaries are established, roles given to the system and external actors are distinguished, and functional exchanges between the system and actors are distributed. System Analysis Blank (SAB) are created to show how it interacts with the system actors.

LA, breaks down the system into logical components to create a coherent and consistent architecture. In this stage, logical components or subsystems are identified using the functions specified in FA. In order to ensure coherence between the logical architecture and functional breakdown, LA diagrams are designed to illustrate the links between functions and logical elements. After that, PA is carried out in order to obtain physical architectures from logical elements. Mapping logical elements to their physical representations and creating physical architecture diagrams that represent the structure and design of the system are the part of this process. The physical architecture is developed using a variety of technologies and solutions, to ensure that they are in line with the boundaries and specifications of the system. Capella is used for architecture representations that corresponds to the suggested MBSE methodology (Figure 2).

.

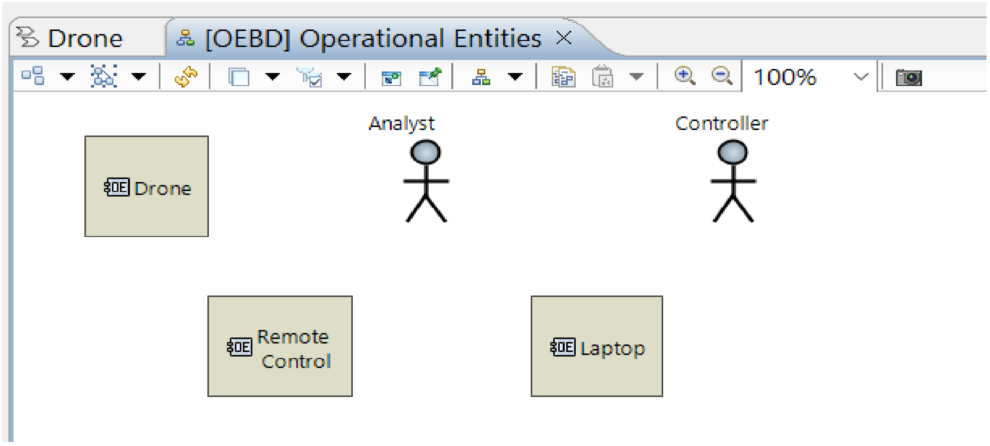

Figure 2. Operational Entity Breakdown Diagram (OEBD) (Capella Software)

3.1 Operational Analysis

The first step involves capturing operational entities (OE) and OA using the OEBD. Here, this step involves determining the different factors that interact with the operating system. These aspects can either be classified as actors or entities. Entities are passive objects such as files or data and actors are active objects, such as users or other application software. Three OEs were established such as drone, remote control and laptop. The controller and analyst has been labelled as operational actors. hen there is the seizure of operational capabilities with the operational capability blank diagram (OCBD). The next step is to define the high-level goals and features of the operating system. Basically, these abilities signify what the operating system is planned to achieve.
In the OCBD, the following OCs were defined(Figure 3):

- Provide State
- Remotely controlled
- Save captured data
- Provide live video footage
- Export captured data
- Analyse data
- Connect the drone to the controller
- Disconnect the drone from the controller
- Connect the controller to the laptop
- Disconnect the controller from the laptop

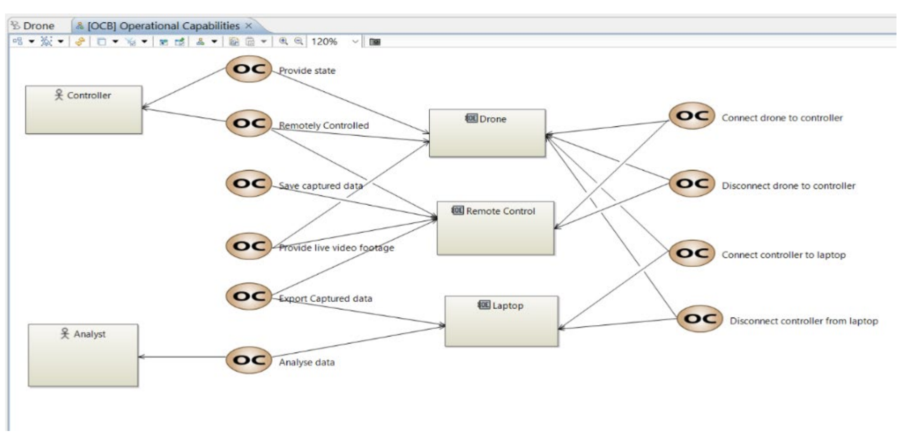

Figure 3. OCBD Diagram (Capella Software)

The next task is to define the interactions between operational activities by the operational activities interaction blank (OAIB) diagram. This step involves identifying how the various operational activities interact with each other. These interactions could include exchanging data or having one activity trigger another. The following are different operational processes (OP) and how different operational actors serve them.

Provide State of Drone: This OP is concerned with the operations associated with checking the health and condition of the drone prior to flying the drone. It contains OA:

- Display State of Drone (Drone)
- Comprehend the State of Drone (Controller)

Connect Controller to Laptop: This OP illustrates how to establish a connection between the controller and a laptop. Users can utilise this connection to upgrade the drone's software, download the recorded flight data, and much more. It includes OA:

- Receive connection (Remote Controller)
- Receive connection (Laptop)
- Connect the controller to laptop (Analyst)

Detach Drone from Laptop: This OP illustrates how to stop the connection between the drone and the laptop, which contains the following OAs:

- Disconnect (Remote Controller)
- Disconnect (Laptop)
- Disconnect the controller from laptop (Analyst)

Control Remotely: This OP comprises actions for managing the drone's flight route, such as take-off and landing, as well as manual flight control. It contains OA:

- Navigate the drone using the controller (Controller)
- Send navigation signals (Remote Control)
- Receive navigation signals (Drone)
- Operate motors (Drone)

Provide live video footage: This OP incorporates actions for capturing, transfering, receiving, and displaying the video tape from the drone to the remote control. It contains OAs:

- Capture image & video data (Drone)
- Send live video footage (Drone)
- Capture thermal images & videos (Drone)
- Receive live footage (Remote Control)

Transfer captured records: This OP encompasses activities of transferring the video captured by the drone on the remote control to the laptop for further examination. This contains OAs:

- Export captured data (Remote Control)
- Receive Captured Data (Laptop)

The next crucial step is allocating operational activities, often referred to allocate operational activities (OAB). Each activity in this step is assigned to a unique operational body or actor which ensures that there is accountability and clarity within the operational framework. In general, OAB determines who has responsibility in which part of the operational system. The final draft represents the operational capability based on operational procedures and scenarios. This involves providing a detailed description of how each of the operational capabilities will be achieved.

3.2 Functional Analysis

SA breaks down the drone's mission into a hierarchy of functions, clarifying its operational processes to meet the requirements of stakeholders. This approach helps in identifying potential misunderstandings. The first step in SA is to determine the actors that interact with the system of interest (SOI) (in this case, a drone) using the context system actors (CSA) diagram. These actors can be external systems, human operators, or any entity that exchanges information with the SOI. Capella is a tool used for modelling systems that can automatically transition OEs and OAs identified earlier during operational analysis into system actors. In this case, the controller, drone, operator, analyst, and laptop are automatically transitioned from OEs and OAs along with their linkages. In this case the following actors were idetifined:

- Controller
- Analyst
- Drone
- Remote control
- Laptop

The next step involves defining the purpose and functionalities of the SOI using the mission and system capabilities (MCB) diagram. This entails identifying the high-level missions (goals) that SOI should achieve and the capabilities (services) that SOI provides to meet those missions(Figure 4).

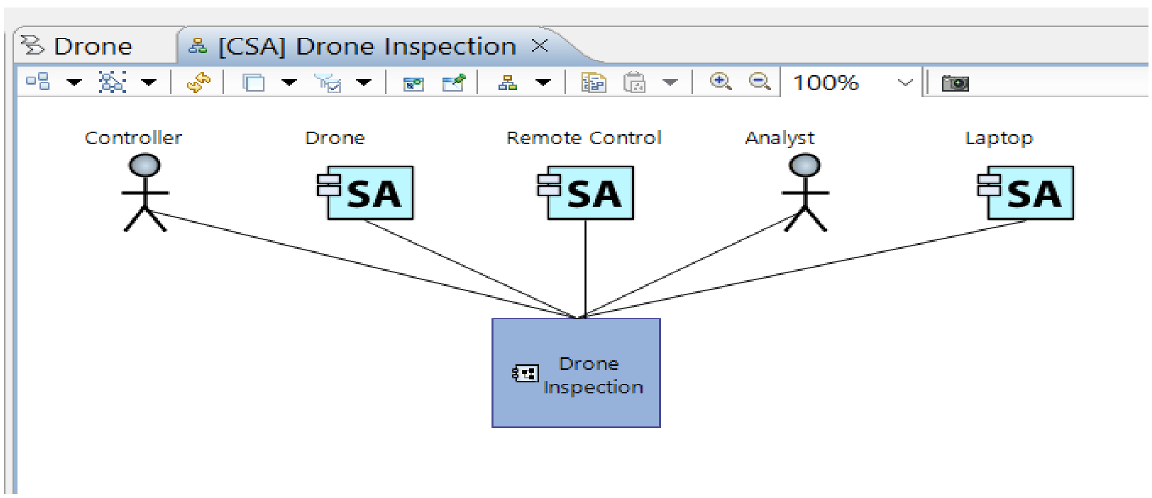

Figure 4 CSA diagram (Capella Software)

Capella facilitates the process by enabling the transition of operational capabilities, as defined earlier, into system capabilities. All the capabilities described in the operational analysis phase were transitioned into the MCB diagram, and all the actors from the CSA diagram were also included. Then, the capabilities were assigned to the respective actors according to their involvement in performing the capability. The next step is to make a system functions breakdown diagram (SFBD). SFBD focuses on refining the operational activities identified during operational analysis into system functions. These system functions represent the specific actions that the SOI performs to fulfil its capabilities. The interactions between system functions are illustrated in the system data flow blank (SDFB) diagram. These interactions, known as functional exchanges, depict the flow of data and control information that enables the system to function as a whole.

In this case, the SDFB diagram describes the interactions that will occur to deliver the required capabilities. Control remotely is a functional chain that enables the control of the drone from a remote location. To perform this function, the system must interact to navigate the drone using the controller, send and receive navigation signals, and operate the motors ((Figure 5).

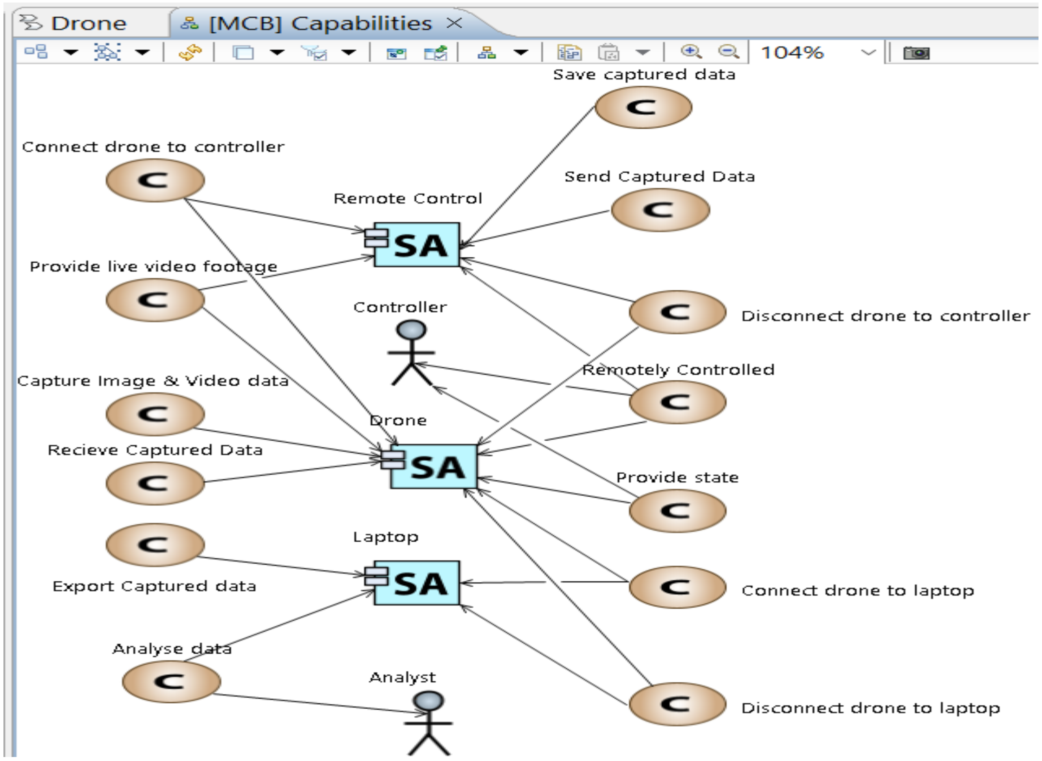

Figure 5. MCB diagram (Capella Software)

In the next step, system components and actors are used to allocate system functions. The system architecture blank diagram (SABD) deals with allocating system functions to system components and actors. This step determines whether an operational activity will be entirely realized by the system, partially shared between the system and actors, or not performed by the system at all. All the system functions (SF) are allocated to the respective systems and actors. The controller was assigned to navigate the drone using the controller, comprehend the drone's state, connect the drone to the controller, and disconnect the drone from the controller. Similarly, all other actors and system components were assigned the respective SFs.

After allocating the SF to respective actors and systems, the interaction links are defined. This link describes the flows between two SFs. Between sending live video footage and receiving live footage, the captured data will flow. Similarly, all required interactions are mapped. Then, the functional chains are described in the SAB diagram. A functional chain refers to a sequential order of interrelated functions or activities that contribute to achieving a specific goal or outcome within a system. The SAB diagram serves as the basis of MBSE for interpreting how a system achieves its intended capabilities. It utilizes two key elements to achieve this, functional chains and scenarios. Functional chains depict the sequential order of functions that work together in a domino effect, ultimately delivering a specific system capability. Scenarios, on the other hand, delve into how the system behaves under particular conditions to realize the same capability. Capella streamlines this process by allowing us to define functional chains first. Then it offers an automated transition to create scenarios based on these established chains. This eliminates unnecessary work by leveraging the existing functional structure to define system behaviour in various situations. Essentially, the SAB diagram, which includes functional chains and scenarios, provides a comprehensive picture of how a system delivers its capabilities under multiple circumstances. Capella's automation helps create these scenarios efficiently, saving valuable time and effort within the MBSE process. Traceability is maintained throughout system analysis to ensure consistency between model elements and their corresponding diagrams. The order of creating diagrams is flexible and should be guided by the modeler's choices and project requirements. The steps and diagrams presented here provide a recommended approach, but Capella allows for customization based on specific needs.

3.3 Logical Analysis

Logical analysis breaks down the drone into logical elements, creating a clear system architecture. The functional and logical diagrams are interconnected, with each function associated with one or more high-level logical elements. The logical architecture of the drone defines logical components or subsystems. The system analysis layer treats the system as a black box, concentrating on specifying the expected behavior of the system and identifying the essential external exchanges with the actors. In contrast, the logical architecture layer begins to reveal the system's internal workings by describing the high-level architectural decisions that will be used to fulfill the stakeholders' expectations. Capella automatically generates a realization link among each logical element (i.e., function, functional exchange and functional chain) and the source system element. It also highlights the transitions that are iterative and incremental; if a new system function is identified while working at the logical level, it must be added to the system level, and the transition must be reapplied.

3.4 Physical Analysis

Physical architecture focuses on translating those functionalities into the tangible components that will constitute the real-world system. This stage introduces technological considerations and implementation choices. The initial step involves realizing physical capability. This step bridges the gap between LA and PA. Capella seamlessly transitions logical capabilities realizations into the physical architecture's contextual realization blank (CRB) diagram. This ensures that all actors identified earlier remain involved in at least one capability realization within the physical architecture. Physical functions are derived and can further refine the previously established logical functions. These functions represent the specific actions that the system's physical components will execute to fulfil the functionalities defined in the logical architecture. The physical functions breakdown diagram (PFBD) visually depicts these functions and any hierarchical relationships between them. Once the physical functions are identified, the physical data flow blank (PDFB) diagrams come into play. These diagrams illustrate the interactions (functional exchanges) that occur between the physical functions.

The next step involves allocating functions to behavioural nodes and deploying physical node components. With both functions and components clearly defined, the physical architecture blank (PAB) diagram becomes the central hub for allocating functions to the relevant behavioural components. This diagram also facilitates the creation of exchanges between behavioural components, representing the concrete implementation of the functional exchanges identified earlier. Additionally, physical links and paths are also defined to specify how components communicate and interact within the system. Like LA, capability realizations in the physical layer are also described using functional chains (sequences of functions) and scenarios (temporal sequences of interactions). Capella offers functionalities to automate the transition and creation of these scenarios from established functional chains (Figure 6).

.

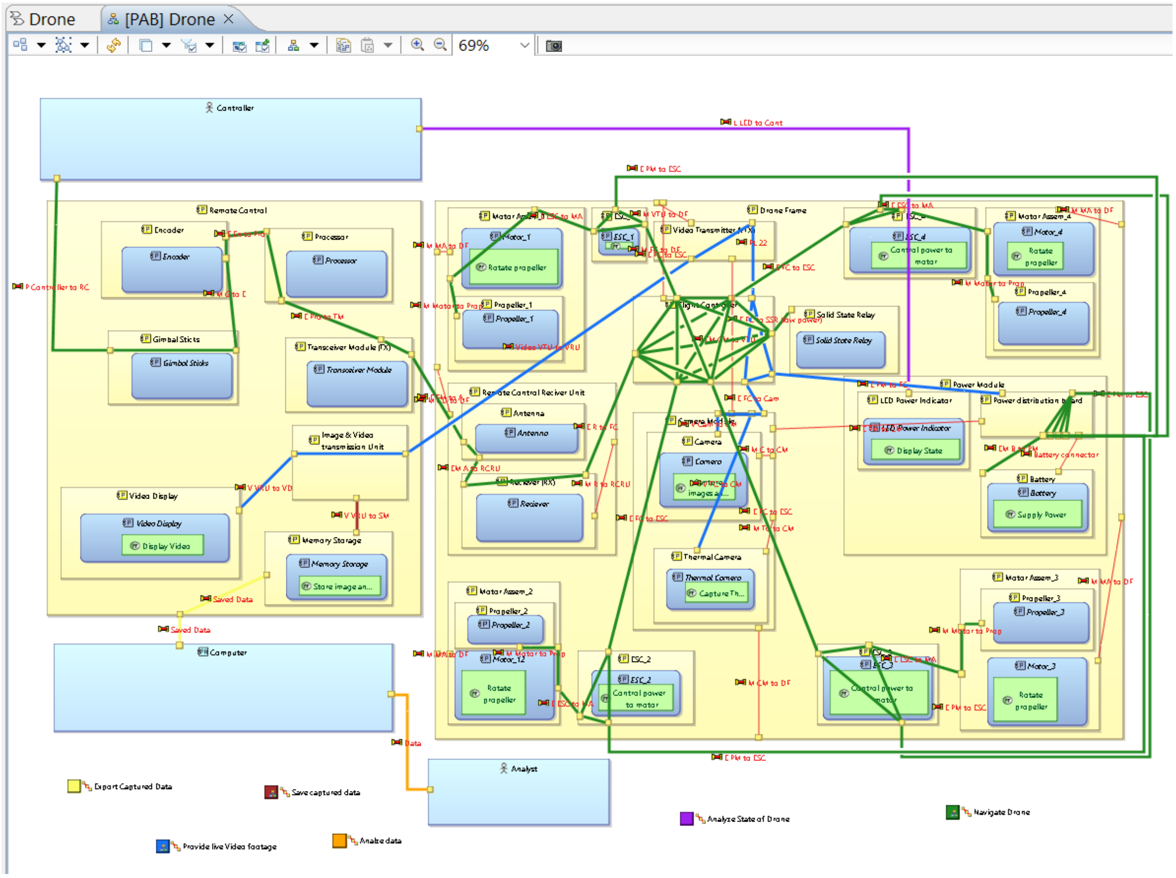

Figure 6. Drone (PAB) (Capella Software)

4. Results and Discussion

This paper examines how the ARCADIA approach can be used to design a drone system, using the Capella software. The methodology of this research work presented a systematic approach, breaking down the design process into four major steps, which included OA, SA, LA, and PA. The paper identifies several important advantages of ARCADIA methodology implementation in a formalized MBSE to design complex systems, especially the drone systems. The early detection of possible problems is possible due to the model-based nature of ARCADIA enabled by Capella. During the design stage, inconsistencies/absence of functionality in the model will be easily identified. Traceability among models and diagrams facilitates good troubleshooting and fixing of detected issues before purchasing or deploying the physical components. Related stakeholders involved in the design of the drone system will be able to communicate and collaborate better through the support of the standardised modelling language and visual representations in ARCADIA. This enables engineers of different domains to informed their decisions during the design phase easily because the system is easy to understand with its various levels of perception. The models at each point of time (OA, SA, LA, PA) can be re-examined and improved when the necessity arises. This flexibility is highly appreciated at the initial stages of drone system design, when requirements and functionalities might shift due to additional research or feedback of stakeholder.

5. Implications of the study

This study emphasizes the concepts and implementation of MBSE approach by presenting the ability of MBSE in designing complicated systems. The research work fills the significant gap to the literature in the area of MBSE and requirement engineering regarding the design of the drone systems. The future academician can use the findings of this research work to understand and improve the knowledge of MBSE in the field of system design. Further, this research work offers an outline by highlighting the value of application based, real world model software research in the field of MBSE. Hence, this research work not only explores the difficulties in drone system design but also offers practical solutions and techniques that may be used to develop complex systems in different domains.

In addition to this, numerous industries that are dealing with complex system design may be substantially affected by this research work. This research work also offers a way for industries to streamline their design process by providing the effectiveness of MBSE methodology. In the modelling phase, early problem detection and design flexibility can greatly reduce development costs and time-to-market for complex objects, ranging from automobiles, aeroplanes to medical equipment and modern machinery. This study demonstrates the potential of MBSE to simplify complex system design processes across multiple domains. This feature is especially helpful in fields where smooth cooperation is essential, such as; those that who create complex products that need input from multiple domain specialist (e.g., software, mechanical, electrical). This research work also highlights how MBSE may be used to verify virtual systems. This results in a lower risk for organizations who prioritise safety, primarily in the aviation sector. Identifying and fixing design flaws using the MBSE method lowers the likelihood of serious accidents and malfunctions while building a physical aircraft.

6. Conclusion and Future Scope

The visual system representation capability of Capella enabled the multidimensional modelling process that presented the real value to the project team. The system diagrams of OCBs and SABs allow non-technical groups to understand the functionality and interaction of the system effortlessly. The improved functional insight resulted in the development of communication patterns and more active participation of all people in the process of design development, as well as in the visual design tools provided by Capella that made the design review faster. The major defects in diagrams could easily be observed through cross-stage inspections (OA, SA, LA, PA). This technique helps the designers to detect and correct the design problems at the initial stages hence enhancing the general quality of the system design. Graphical models developed by Capella served as self-documenting system architecture.

Therefore, ARCADIA approach implemented by Capella is a valuable method to design a complex system such as a drone. The model-based methodology, along with the graphical representation features of Capella, improves communication and allows easy detection of faults at each stage and simplifies design documentation. This study may serve as a foundation and be applied by future researchers to design application-specific drones with the help of the ARCADIA approach and Capella software. The key functionalities of a drone system are also considered in this research. This research work can be extended in future by combining particular functionalities that may be based on the proposed use of the drone. The ARCADIA and Capella software of project design will have the features of problem recognition and path planning. In order to test the system in various contexts through system performance evaluations, the models created in this research work can be tested to offer different scenarios. Capella allows model simulation capability prior to constructing physical prototypes and thus the behaviors of the drone system can be virtually tested. This simulation expertise may be extremely useful in the testing of the flight dynamics, control stability, and sensor performance in a non-normal environment. The development of 3D representations of the drone with references to the predetermined physical parts and their mutual connection can be facilitated through the combination of computer-aided design (CAD) software.

Moreover, the construction of control and power electronic circuit design will be easy by integrating software that supports electrical circuit design (ECAD). The capabilities of Capella are highly useful in designing the complex systems, particularly in projects with changing requirements across different disciplines. This research work may be applied by the future researchers to investigate the alternative design choices, to add individual elements into a specific application, and to integrate with other design tools to produce a premier drone system.

References

Aigner, A., and Khelil, A. Assessment of model-based methodologies to architect cyber physical systems, Proceedings of the International Conference on Omni-Layer Intelligent Systems - COINS '19, Unknown, Ed., New York, New York, USA: ACM Press, pp. 146–151, 2019,ISBN: 9781450366403. DOI: 10.1145/3312614.3313779.

Baker Jr, L., and Long, J. E. Role of System Engineering Across the System Life Cycle. Vitech white paper, Vitech Corporation, Vienna, VA. Vienna, Virginia. 2020, Retrieved from http://www.vitechcorp.com/resources/technical_papers/200701031632040.baker_long.pdf.

Bao, Y., Ming, X., Chen, Z., Zhou, T., and Zhang, X. How to enable human-centric collaboration in social product development paradigm: A practical and theoretical exploration. Computers & Industrial Engineering, vol. 198, no. 1, pp. 110632, 2024.

Carolan, M., Omara, M., and Nikou, A. A model-based engineering approach for designing a tethered UAV for precision agriculture applications. Computers and Electronics in Agriculture, vol. 199, np. 1, pp. 107062, 2022.

Dassault Systèmes. What is Model-Based Systems Engineering (MBSE)? Retrieved 10th April, 2024, from https://www.3ds.com/insights/model-based-systems-engineering,2023.

Feng, Y., Gao, G., Wang, P., Fu, M., and Zhang, Z. Towards model-based requirements engineering: Construction method of stakeholder value networks model based on MBSE. Computers & Industrial Engineering, vol. 195, no. 1, pp. 110398, 2024.

García, Á., Bregon, A., and Martínez-Prieto, M. A. Towards a connected Digital Twin Learning Ecosystem in manufacturing: Enablers and challenges. Computers & Industrial Engineering, vol. 171, no.1, pp. 108463, 2022.

Grieves, M., and Weilkiens, T. Model-Based Systems Engineering: A Guide for Engineers and Program Managers. CRC Press, 2022.

Harkat, H., Camarinha-Matos, L. M., Goes, J., and Ahmed, H. F. Cyber-physical systems security: A systematic review. Computers & Industrial Engineering, vol. 188, no. 1, pp. 109891, 2024.

IBM. Model-based systems engineering. Retrieved on 4th April 2024 from https://www.ibm.com/topics/model based-systems, 2023.

INCOSE (Ed.). INCOSE systems engineering handbook. John Wiley & Sons, 2023.

INCOSE. International Council on Systems Engineering (INCOSE) MBSE Initiative. Retrieved 5th April 2024, from https://www.incose.org/communities/working-groupsinitiatives/mbse-initiative, 2020.

Jiang, L., Su, S., Pei, X., Chu, C., Yuan, Y., and Wang, K. Product-part level digital twin modeling method for digital thread framework. Computers & Industrial Engineering, vo. 179, no. 1, pp. 109168, 2023.

Jing, Z., Li, L., Lyu, Y., Wang, R., Wang, Y., Wang, D., and Wang, F. Y. Autonomous services: The evolution of services through intelligent vehicles. IEEE Transactions on Intelligent Vehicles, vol. 8, no. 11, pp. 4468-4473, 2023.

Leroux, R., Pantel, M., Ober, I., and Bruel, J. M. Model-based systems engineering for systems simulation. In Leveraging Applications of Formal Methods, 8th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2018), pp. 429-448, 2018.

Lins, T., and Oliveira, R. A. R. Cyber-physical production systems retrofitting in context of industry 4.0. Computers & industrial engineering, vol. 139, no. 1, pp. 106193, 2020.

Lu, J., Tsinarakis, G., Sarantinoudis, N., Arampatzis, G., Zheng, X., and Kiritsis, D. A semantic model-based systems engineering approach for assessing the operational performance of metal forming process. Computers & Industrial Engineering, vol. 190, no. 1, pp. 110042, 2024.

Madni, A. M., and Sievers, M. Model-based systems engineering: Motivation, current status, and research opportunities. Systems Engineering, vol. 21, no. 3, pp. 172-190, 2018.

Markets and Markets. Drone Services Market by Type, Application (Aerial Photography, Data Acquisition and Analytics), Industry, and Geography - Global Forecast to 2025. Retrieved from https://www.marketsandmarkets.com/Market-Reports/drone servicesmarket-80726041.html on 12th April 2024, 2023.

Navas, J., Bonnet, S., Tannery, P., and Voirin, J. L. Architecture Design of Nuclear Power Plants Systems through Viewpoints-based Systems Analysis. In INCOSE International Symposium, vol. 28, no. 1, pp. 1093-1108, 2018.

Noguchi, R. A. Architecting Descriptive Models for MBSE. In INCOSE International Symposium vol. 33, no. 1, pp. 1057-1074, 2023.

Plazanet, S., and Navas, J. Model-Based Systems Engineering as an Enabler of Agility. INSIGHT, vol. 26, no. 2, pp. 26-30, 2023.

Ramirez, C. A., Agrawal, P., and Thompson, A. E. An approach integrating model-based systems engineering, iot, and digital twin for the design of electric unmanned autonomous vehicles. Systems, vol. 13, no. 2, pp. 73, 2025.

Roques, P. Systems architecture modeling with the Arcadia method: a practical guide to Capella. Elsevier, 2017.

Thales. Datasheet_Arcadia. Retrieved from https://www.eclipse.org/capella/resources/Datasheet_Arcadia.pdf. on 17th March 2024.

Vitech Corporation. Genesys 6.0: System definition guide. Retrieved from http://www.vitechcorp.com/support/documentation/genesys/600/systemdefinitionguide.pdf, . 2018.

Voirin, J. L. Model-based system and architecture engineering with the arcadia method. Elsevier, 2017.

Weilkiens, T., Scheithauer, A., Di Maio, M., and Klusmann, N. Evaluating and comparing MBSE methodologies for practitioners. IEEE International Symposium on Systems Engineering (ISSE), pp. 1-8, 2016.
Xu, L., Yang, Q., Qin, M., Wu, W., and Kwak, K. (2023). Collaborative human recognition with lightweight models in drone-based search and rescue operations. IEEE Transactions on Vehicular Technology, vol. 73, no. 2, pp. 1765-1776, 2023.
Zhang, X., Wu, B., Zhang, X., Duan, J., Wan, C., and Hu, Y. An effective MBSE approach for constructing industrial robot digital twin system. Robotics and Computer Integrated Manufacturing, vol. 80, no. 1, pp. 102455, 2023.
Zhang, L., Chen, Z., Laili, Y., Ren, L., Deen, M. J., Cai, W., ... and Gu, P. (2025). MBSE 2.0: Toward more integrated, comprehensive, and intelligent MBSE. Systems, vol. 13, no. 7, pp. 584, 2025.

Biographies

Ram Dayal Pandey is pursuing his Doctor of Philosophy (Ph.D.) degree in the Department of Management Studies and Industrial Engineering at IIT (ISM) Dhanbad, India. He holds Master of Technology degree with Hons. in Industrial Engineering from NIT Jalandhar, Punjab, India and Bachelor of Technology degree in Mechanical Engineering with Hons. from Dr. A. P. J. AKTU Lucknow. He is also a reviewer at Benchmarking: An International Journal. His research interests include Multi criteria decision making (MCDM), Supply Chain Management, Sustainability, Industry 4.0, Digitalization, Operations Management and System Dynamics Modelling.

Prajwal Kayalkar is currently working in Siemens Pune, India. He holds his Master of Technology degree in Industrial Engineering and Management from IIT (ISM) Dhanbad, India. His research interest includes Model based systems engineering (MBSE) and Discrete Event Simulation.

Dr. Shikha Singh received the Ph.D. degree in industrial and management engineering from Indian Institute of Technology (IIT) Kanpur, Kanpur, India, in 2019. She is currently an Assistant Professor in the Industrial Engineering and operations area with the IIT (ISM), Dhanbad, India. Before joining IIT, she was with Hindustan Aeronautics, Ltd., for more than a decade. Her research interest includes institutionalizing various management concepts and their supported information systems in manufacturing organizations, project management, business process management, product lifecycle management, systems engineering, operations management, and management of different technologies related to digitalization and Industry 4.0.