Annex A

This annex describes the elements in the CPT.

A.1 Digital Twin CPT Detail Description

	What it Is		What it Does	What it Means
No	**Digital Twin Capability**	**Category**	**Ability**	**Purpose**
1	Data Acquisition & Ingestion	1. Data Services	The ability to configure and acquire data from different data sources including control system, historians, IoT sensors, smart devices, engineering system, enterprise systems, etc.	The purpose is to acquire data from the physical world, engineering technology systems, and information technology systems to support subsequent processing and insight generation.
2	Data Streaming	1. Data Services	The ability to transfer of large volumes of data continuously and incrementally between a source and a destination without having to access all data at the same time.	The purpose is to acquire fast continuous packets of information that are changing at high speed to be able to get near real-time insights.
3	Data Transformation & Wrangling	1. Data Services	The ability to convert data types and properties through cleaning, structuring and enriching raw data to make if suitable for further processing and analytics.	The purpose is to make data useable in digital twins.

4	Data Contextualization	1. Data Services	The ability to add language or metadata to enrich real time or transactional data.	The purpose is to combine data from different sources (such as real-time data and slower-changing contextual data) to make it suitable for subsequent processing by the digital twin.
5	Batch Processing	1. Data Services	The ability to execute against previously collected data in bulk form.	The purpose is to provide is an efficient way of processing high volumes of data in batches or groups.
6	Real-time Processing	1. Data Services	The ability to manage and act on the captured data with minimal latency.	The purpose is to support immediate insights from the data.
7	Data PubSub Push	1. Data Services	The ability to package filtered data to different services based on a publish/subscribe model.	The purpose is to provide information to subscribed digital twin consumers.
8	Data Aggregation	1. Data Services	The ability to gather raw data and express in a summary form.	The purpose is to gather data from multiple sources with the intent of combining these data sources into a summary for data analysis.
9	Synthetic Data Generation	1. Data Services	The ability to generate synthetic data based on patterns and rules in existing sources.	The purpose is to create representative synthetic data that can used by the digital twin to train and score predictive models.
10	Ontology Management	1. Data Services	The ability to manage knowledge graphs and ontologies.	The purpose is to enable a digital twin to interpret data directly from knowledge graphs and ontologies.

11	Digital Twin Model Repository	1. Data Services	The ability to store, manage and retrieve the metadata that describe the digital twin model. The model can include formal data names, comprehensive data definitions, proper data structures and precise data integrity rules.	The purpose is to register and manage a portfolio of digital twin models in a central repository to improve configuration management and model governance.
12	Digital Twin Instance Repository	1. Data Services	The ability to store, manage and retrieve digital twin instance data that conforms to the requirements of the digital twin model.	The purpose is to store, manage and retrieve digital twin instance state data.
13	Temporal (Time Series) Data Store	1. Data Services	The ability to store, organize and retrieve data relating to time instances through temporal data types, and store information relating to past, present and potentially future time.	The purpose is to store, manage and retrieve temporal (timeseries) data.
14	Data Storage & Archive Services	1. Data Services	The ability to store, organize and retrieve data based on how frequently it will be accessed and how long it will be retained.	The purpose is to reduce the cost and effort of managing digital twin data by using hot, cold and archival data services
15	Simulation Model Repository	1. Data Services	The ability to store, manage and retrieve the algorithmic codebase, business rules and metadata that describe a simulation model.	The purpose is to register and manage a portfolio of simulation models in a central repository to improve configuration management and model governance.

16	AI Model Repository	1. Data Services	The ability to store, manage, search and retrieve the algorithmic codebase that describe an artificial intelligence (AI) model or machine learning (ML) model.	The purpose is to register and manage a portfolio of AI and machine learning models in a central repository to improve configuration management and model governance.
17	Enterprise System Integration	2. Integration	The ability to integrate the digital twin with existing enterprise such as ERP, EAM, CRM and CMMS.	The purpose is to integrate business applications, such as ERP, to enable data to flow between digital twin systems with ease.
18	Engineering Systems Integration	2. Integration	The ability to integrate the digital twin with existing engineering systems such as CAD, CAM, BIM and Historians.	The purpose is to integrate business applications, such as CAD, to enable data to flow between digital twin systems with ease.
19	OT/IoT System Integration	2. Integration	The ability to integrate directly with control systems and IOT devices/sensors and SCADA.	The purpose is to integrate business applications, such as SCADA, to enable data to flow between digital twin systems with ease.
20	Digital Twin Integration	2. Integration	The ability to integrate or access information from existing digital twin instances.	The purpose is to integrate digital twin applications with one another to enable interoperable digital twins.
21	Collaboration Platform Integration	2. Integration	The ability for the digital twin to interface with platforms like Yammer, Jabber, Teams and Slack.	The purpose is to integrate collaboration platforms to provide digital twin users with a conversational user interface.

22	API Services	2. Integration	The ability for the digital twin to publish APIs to external, partner and internal developers to access data and services.	The purpose is to simplify digital twin development by allowing data to integrate data with the endpoint.
23	Edge AI & Intelligence	3. Intelligence	The ability to make decisions at the device level based on real -time data, distribution, and federation of analytics at the edge instead of transporting the data to the cloud to perform analytics.	The purpose is to make real-time decisions at the edge.
24	Command & Control	3. Intelligence	The ability to execute upon work instructions without human interaction. Control would be limited to IoT devices and non-plant controls.	The purpose is to support future smart IoT devices with centralized management.
25	Orchestration	3. Intelligence	The ability to coordinate the automated configuration, management and coordination of systems, applications, digital twins and services.	The purpose it to manage complex tasks and workflows between different systems, applications, digital twins or systems of digital twins easily.
26	Alerts & Notification	3. Intelligence	The ability to display and manage alerts, messages, message queues, triggers and notifications.	The purpose is to trigger actions that may require intervention to the ongoing processes.
27	Reporting	3. Intelligence	The ability to generate configurable and customizable reports to get insights into the data.	The purpose is to get insights into the data that can be useful for various stakeholders in the system as well as for regulatory compliance.

28	Data Analysis & Analytics	3. Intelligence	The ability to analyze data through charts, tables and dashboards; to fetch data between dates and filter data based on various criteria. The analysis of data, typically large sets of business data, using mathematics, statistics and software with an objective to draw conclusions.	The purpose is to understand past trends from historical data.
29	Prediction	3. Intelligence	The ability to estimate that a specified event will happen in the future or will be a consequence of other events.	The purpose is to use historical data, engineering, and analytical models to predict events before they occur.
30	Machine Learning (ML)	3. Intelligence	The ability of computer algorithms to improve a digital twin automatically through experience. The algorithms build a mathematical model based on training data, to make predictions or decisions without being explicitly programmed to do so. It is seen as a subset of artificial intelligence.	The purpose is to enable the digital twin and digital twin systems to learn from data, identify patterns and make decisions with minimal human intervention.

31	Artificial Intelligence	3. Intelligence	The ability for a system to perform actions and take decisions like humans. AI would include machine learning, natural language processing, knowledge modelling and representation, reasoning, inferencing etc. It is based on the capacity of a computer to perform operations analogous to learning and decision making in humans, as by an expert system, a program for CAD or CAM or a program for the perception and recognition of shapes in computer vision systems.	The purpose is to enable a digital twin or a digital twin system to take actions and decisions similar to humans.
32	Federated Learning	3. Intelligence	The ability to train an algorithm across multiple decentralized digital twin edge devices or servers holding local data samples, without exchanging their data samples.	The purpose is to enable multiple actors to build a common, robust machine learning model without sharing data, thus addressing critical issues such as data privacy, data security, data access rights and access to heterogeneous data.
33	Simulation	3. Intelligence	The ability to create an approximate imitation of a process or a system using past historical information, physical models, video, audio and animation. What-if-scenarios.	The purpose is to imitate the behavior of a physical system in the digital twin before applying to the physical world. Training operations and maintenance teams on simulated digital twins is another purpose of simulation.

34	Mathematical Analytics (Engineering Calculations)	3. Intelligence	The ability to perform mathematical and statistical calculations to enable physics-based and other mathematical models.	The purpose is to enable the use of physics models and mathematics calculations in digital twin analytics.
35	Prescriptive Recommendations	3. Intelligence	The ability to create prescriptive recommendations based on business rules and AI logic to suggest the best next actions to take when a pre-determined event happens.	The purpose is to enable digital twins to provide guidance based on a combination of analytics, business rules and workflow to create actions and deliver business outcomes.
36	Business Rules	3. Intelligence	The ability to create, manage and use business rules that influence the digital twin behavior throughout its lifecycle.	The purpose is to enable digital twins to provide and manage business rules that influence a digital twin’s behavior.
37	Distributed Ledger & Smart Contracts	3. Intelligence	The ability to use distributed ledgers for digital twin applications that require immutable data for digital twin instances, transactions and automation (smart contracts).	The purpose is to enable digital twins to interact in an automated, trustworthy and responsible manner with systems that support smart contracts and provide a full, immutable transaction record.
38	Composition	3. Intelligence	The ability to use a modular digital twin application development approach to compose and recompose digital twin services that deliver use case specific outcomes rapidly.	The purpose is to compose or recompose digital twins from a set of packaged, reusable business capabilities (PBCs) to reduce time to value, duplication and support citizen development of digital twins.

39	Basic Visualization	4. UX	The ability to visualize data through simple charts, graphs, simple dashboards, tables, hierarchical and basic 3D views of the assets graphically or parametrically (that is, through parameters and values).	The purpose is to help people understand the significance of data by placing it in a visual context.
40	Advanced Visualization	4. UX	The ability to visualize data through complex charts and graphs, dashboards fetching raw and process data from multiple systems, complex 3D models and animations, visualizations with overlayed data from different systems graphically or parametrically (that is, through parameters and values).	The purpose is to help people understand the significance of data by placing it in a visual context.
41	Real-time Monitoring	4. UX	The ability to present and interact with continuously updated information streaming at zero or low latency.	The purpose is to help make decisions that are of consequence to real-time.
42	Entity Relationship Visualization	4. UX	The ability to present digital twin entities and their hierarchical or graph-based relationships interactively.	The purpose is to help business users navigate and interact with complex entity (asset) hierarchies in a user-friendly manner.

43	Augmented Reality (AR)	4. UX	The ability to provide an interactive experience of a real-world environment where the objects that reside in the real world are enhanced by computer-generated perceptual information such as visual, auditory or haptic environments.	The purpose is to realize an improved, immersive and interactive experience.
44	Virtual Reality (VR)	4. UX	The ability to provide a simulated experience that can be similar to, or completely different from, the real world.	The purpose is to realize an improved, immersive, and interactive experience.
45	Dashboards	4. UX	The ability to provide a graphical user interface that provides at-a-glance views of key performance indicators relevant to a particular objective or business process.	The purpose is to enable various personas in operations, technology, and business to understand the current or past state of a system visually.
46	Continuous Intelligence	4. UX	The ability to analyze data in flight (signals) to derive insights and actions in a business user-focused visual interface.	The purpose is to have various personas in operations, technology, and business to make informed real-time decisions.
47	Business Intelligence	4. UX	The ability to analyze stored data (records) to derive insights and actions in a business user focused visual interface.	The purpose is to have various personas in operations, technology, and business to make informed real-time decisions.
48	Business Process Management & Workflow	4. UX	The ability to execute a sequence of actions as a process flow to achieve specific business outcomes.	The purpose is to have effective, repeatable actions that deliver the business outcomes of the digital twin.

49	Gaming Engine Visualization	4. UX	The ability to create immersive virtual worlds and interactive experiences with gaming engine technology.	The purpose it to enable digital twins in a digital metaverse where users interact with the digital twin highly interactively.
50	3D rendering	4. UX	The ability to render 3D visualizations from point cloud data sets generated by LiDAR and other scanning technologies.	The purpose is to interact with large point cloud and 3D datasets in a user-friendly manner.
51	Gamification	4. UX	The ability to enable game playing in digital twin interactions.	The purpose is to facilitate gamification elements such as points scoring, badges and competition in the user experience and interactive engagement of a digital twin.
52	Device Management	5. Management	The ability to provision and authenticate, configure, maintain, monitor and diagnose connected IoT devices operating as part of digital twin environment.	The purpose of (IoT) device management is to provide and support the spectrum of functional capabilities of the devices and sensors.
53	System Monitoring & Alerting	5. Management	The ability to observe digital twin systems, applications, and services by collecting, analyzing and acting on their health data to maximize their availability and performance.	The purpose of system monitoring is to provide and support the whole spectrum of digital twin systems, applications and services.

54	Logging	5. Management	The ability to record events and transactions; to access user data and their transactions to understand and trace the activities occurring in a digital twin system.	The purpose of event logging is to provide records that enable event activities to be traced within a digital twin system.
55	Data Governance	5. Management	The ability to manage the availability, usability, integrity and security of the data in digital twin systems, based on internal data standards and policies that control data usage.	The purpose is to ensure that data is consistent and trustworthy and doesn't get misused.
56	Data Encryption	6.Trustworthiness	The ability to convert digital twin data from a readable format into an encoded format that can be used to transfer data securely. It includes the ability to decrypt the data to read or process the data once it reaches its destination.	The purpose of data encryption is to protect digital data confidentiality as it is stored in a digital twin system, accessed and transmitted.
57	Device Security	6.Trustworthiness	The ability to enforce authenticated and authorized access to IoT device data through identity management, role-based access, encryption and policies.	The purpose is to control access to device data by having the appropriate privileges and enforcement framework for users and programs.

58	Security	6.Trustworthiness	The ability to protected digital twins from unintended or unauthorized access, change or destruction. Security concerns equipment, systems and information, ensuring availability, integrity and confidentiality of information.	The purpose is to ensure a digital twin is protected from unintended or authorized access, change or destruction.
59	Privacy	6.Trustworthiness	The ability to enable the rights of individuals that interact with digital twins to control or influence what information related to them may be collected and stored and by whom and to whom that information may be disclosed.	The purpose of privacy is to ensure the rights of individuals with regards to data collection, storage and use is respected and enforced.
60	Safety	6.Trustworthiness	The ability to operate digital twins without causing unacceptable risk of physical injury or damage to the health of people, either directly, or indirectly as a result of damage to property or to the environment.	The purpose is to ensure a digital twin is operating safely without causing an unacceptable risk to safety.
61	Reliability	6.Trustworthiness	The ability of a digital twin system or component to perform its required functions under stated conditions for a specified period. This includes expected levels of performance, QoS, functional availability and accuracy.	The purpose is to ensure a digital twin is able to operate and maintain an acceptable level of service as continuously as possible.

62	Resilience	6.Trustworthiness	The ability of a digital twin system or component to maintain an acceptable level of service in the face of disruption. This includes the ability to recover lost capacity in a timely manner (using a more-or-less automated procedure) or to reassign workloads and functions.	The purpose is to ensure a digital twin is able to operate and maintain an acceptable level of service when disrupted.

form are analyzed in Section II, and the current development status of these tools are reviewed. Section III gives the design concept of the makeTwin, and introduces its ten core functional modules. Section IV presents the workflow of the makeTwin as well as its interaction mechanisms. Section V introduces the digital twin system for a chemical fiber textile shop-floor (CFTS) which is developed according to the proposed makeTwin reference architecture. Section VI concludes this work and points out the future works.

2. Demands and definitions of digital twin software platform

Industrial software is an important carrier for the implementation of the digital twin. Researchers, developers, and enterprise users all rely on digital twin software platforms to support the development of digital twin application services. On the one hand, digital twin software platforms are expected to have features such as configurability, callability, modifiability, and extendibility for digital twin models, data, algorithms, IoT connectivity, interaction, simulation, and visualization. On the other hand, digital twin software platforms should be comprehensive, universal, and developmental to meet the different needs of researchers, developers, and enterprise users. Therefore, a detailed survey is conducted, and the different users' demands for digital twin software platforms are summarized, as shown in Fig. 1 and Table 1.

(1) Digital twin model

Various users expect digital twin industrial software that can meet different functions when conducting research and development on digital twin models. Such functions are model import, model creation, model conversion, model lightweight, model assembly, model fusion, model validation, and model correction. The specific demands are as follows.

- **Model import**. It should support a diverse range of geometry models in multiple formats. Software with this capability includes, but is not limited to, TwinBuilder,[40] AWS IoT TwinMaker,[41] Azure Digital Twins,[42] and 3DEXPERIENCE.[43] Additionally, it should have the ability of importing various types of mechanism models, including physics, behavior, and rule models, as exemplified by 3DEXPERIENCE[43 and TwinBuilder[40].
- **Model creation.** Comprehensive modeling capabilities with multiple physics, programming languages, and high-fidelity simulations are essential. For instance, TwinBuilder[40] and MindSphere[44] enable the creation of **multidisciplinary** domain models, including structure, fluid, and electromagnetic.
- **Model conversion.** It must possess the capability to perform format conversion from various input formats to the designated target format.
- **Model lightweight.** Fast lightweighting of large-scale models should be supported.
- **Model assembly.** The ability to construct complex models from basic models is essential, such as using component models to build complex equipment model with multi-layer spatio-temporal correlations. For instance, Azure Digital Twins[42] and 3DEXPERIENCE[43] support model assembly to a certain degree.
- **Model fusion.** Establishing coupling relationships between models in different disciplines and fields must be supported. For example, TwinBuilder[40] realizes the coupling of multi-physics field models.

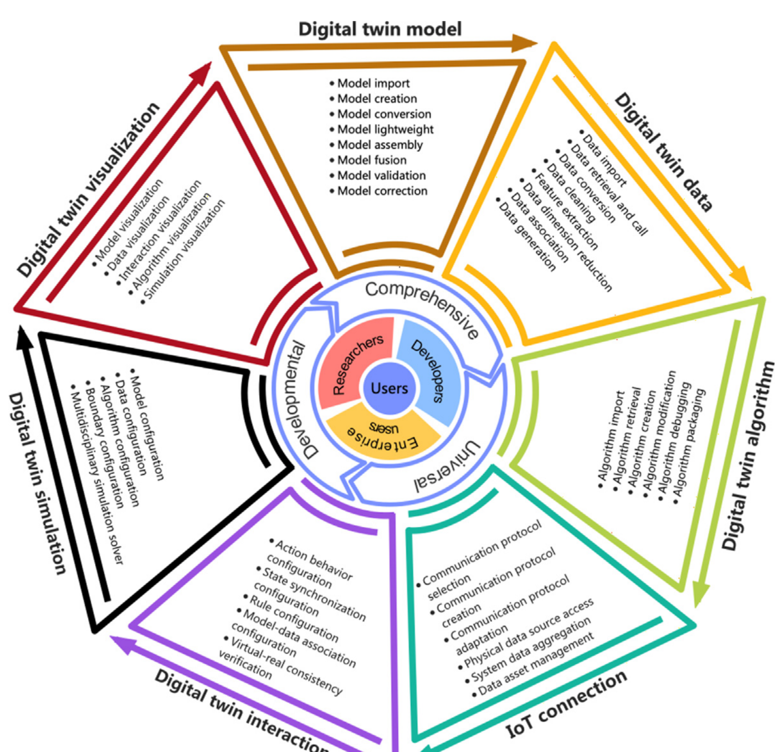

Fig. 1 Demands of users for digital twin software platform.

Table 1 Functional demands of users for digital twin software platform.

Core functions	Functional demands	Core functions	Functional demands
$\mathbf{F_1}$: Digital twin model	$\mathbf{F_{1.1}}$: Model import $\mathbf{F_{1.2}}$: Model creation $\mathbf{F_{1.3}}$: Model conversion $\mathbf{F_{1.4}}$: Model lightweight $\mathbf{F_{1.5}}$: Model assembly $\mathbf{F_{1.6}}$: Model fusion $\mathbf{F_{1.7}}$: Model validation $\mathbf{F_{1.8}}$: Model correction	$\mathbf{F_5}$: Digital twin interaction	$\mathbf{F_{5.1}}$: Action behavior configuration $\mathbf{F_{5.2}}$: State synchronization configuration $\mathbf{F_{5.3}}$: Rule configuration $\mathbf{F_{5.4}}$: Model-data association configuration $\mathbf{F_{5.5}}$: Virtual-real consistency verification
$\mathbf{F_2}$: Digital twin data	$\mathbf{F_{2.1}}$: Data import $\mathbf{F_{2.2}}$: Data retrieval and call $\mathbf{F_{2.3}}$: Data conversion $\mathbf{F_{2.4}}$: Data cleaning $\mathbf{F_{2.5}}$: Feature extraction $\mathbf{F_{2.6}}$: Data dimension reduction $\mathbf{F_{2.7}}$: Data association $\mathbf{F_{2.8}}$: Data generation	$\mathbf{F_6}$: Digital twin Simulation	$\mathbf{F_{6.1}}$: Model configuration $\mathbf{F_{6.2}}$: Data configuration $\mathbf{F_{6.3}}$: Algorithm configuration $\mathbf{F_{6.4}}$: Boundary configuration $\mathbf{F_{6.5}}$: Multidisciplinary simulation solver
$\mathbf{F_3}$: Digital twin algorithm	$\mathbf{F_{3.1}}$: Algorithm import $\mathbf{F_{3.2}}$: Algorithm retrieval $\mathbf{F_{3.3}}$: Algorithm creation $\mathbf{F_{3.4}}$: Algorithm modification $\mathbf{F_{3.5}}$: Algorithm debugging $\mathbf{F_{3.6}}$: Algorithm packaging	$\mathbf{F_7}$: Digital twin visualization	$\mathbf{F_{7.1}}$: Model visualization $\mathbf{F_{7.2}}$: Data visualization $\mathbf{F_{7.3}}$: Interaction visualization $\mathbf{F_{7.4}}$: Algorithm visualization $\mathbf{F_{7.5}}$: Simulation visualization
$\mathbf{F_4}$: IoT Connection	$\mathbf{F_{4.1}}$: Communication protocol selection $\mathbf{F_{4.2}}$: Communication protocol creation $\mathbf{F_{4.3}}$: Communication protocol adaptation $\mathbf{F_{4.4}}$: Physical data source access $\mathbf{F_{4.5}}$: System data aggregation $\mathbf{F_{4.6}}$: Data asset management.		

- **Model validation.** The ability to proficiently validate the precision of the developed digital twin model is essential. For example, TwinBuilder[40] and Azure Digital Twins[42] provide the ability to verify and optimize twin models.
- **Model correction.** It must possess the capability to execute model corrections. For example, the Core Simulation Live[45] enables users to update analysis results while modifying geometric entities dynamically.

(2) Digital twin data

When carrying out digital twin data research and application development, users expect to be able to quickly and adeptly process heterogeneous data from multiple sources while meeting the following requirements.

- **Data import.** First, users hope to import various types and formats of data and achieve format compatibility with heterogeneous data. For instance, 3DEXPERIENCE[43] is capable of directly importing up to 82 data formats, while AWS IoT TwinMaker[41] also support the importation of multi-source and heterogeneous data.
- **Data retrieval and call.** Ensuring data integrity is essential while retrieving and calling data. For example, 3DEXPERIENCE[43] solves various problems such as data standardization, low efficiency of data design tools, and poor accuracy of manual data, enabling the automation of processing complex and diverse full-space 3D geographic information spatial data in a simple and fast way.
- **Data cleaning.** Integrating, filtering, correlating, and synthesizing data is necessary to facilitate judgment, planning, validation, and diagnosis. For example, Beacon[46] and Predix[47] support the fusion of multi-source and heterogeneous data and virtual data collected. Meanwhile, with the help of data fusion software Pycharm, the efficiency of debugging, syntax highlighting, project management, code jumping, intelligent prompts, automatic completion, unit testing, and version control can be improved.
- **Data conversion.** It should be able to transmit and exchange data through one or more data links.
- **Feature extraction.** The function should identify the most effective features in the original features and convert them into features with clear physical meaning, geometric features, or statistical significance.
- **Data association.** It should be possible to associate uncertain multi-source and heterogeneous data observations with trajectories.
- **Data dimension reduction.** It should be capable of performing dimensionality reduction on both linear and non-linear data.
- **Data generation.** It refers to the ability to generate virtual data with a similar distribution to the actual data in specific operating conditions with simulation. For example, Unity3D Computer Vision Datasets[48] solves the issue of insufficient training data when developing computer vision applications.

(3) Digital twin algorithm

Users expect the digital twin algorithm to implement built-in or custom algorithm models with the following capabilities.

- **Algorithm import.** It requires to support algorithm models to be developed in multiple languages, such as Python, C + +, Java, etc. For example, Azure Digital Twins[42] supports running Kusto queries and advanced algorithms using Python plugin.
- **Algorithm retrieval.** It needs to retrieve and call multiple algorithm models from the algorithm library.
- **Algorithm creation.** Typically, users aspire to generate customized algorithms utilizing the development environment's assistance. Software with this capability includes Unity Manufacturing Toolkits,[49] Predix,[47] etc.
- **Algorithm modification.** With an editable interface of the algorithm, users are allowed to modify the algorithm structure and parameters.
- **Algorithm debugging.** It requires to support the compilation and debugging of the algorithm model.
- **Algorithm packaging.** It refers to combining algorithm models' properties, structures, and operations into a standalone entity.

(4) IoT connection

When carrying out an IoT connection, that is, realizing the virtual-real interconnection of digital twin, users anticipate the following requirements to facilitate the establishment of a data bridge between the physical space and virtual space.

- **Communication protocol selection.** A rich set of encapsulated communication protocols are required to be called upon. For example, Unity Manufacturing Toolkits[49] and Predix[47] incorporate standard module libraries for mainstream communication protocols.
- **Communication protocol creation.** It should be possible to create communication protocols for different physical devices.
- **Communication protocol adaptation.** For instance, Predix[47] supports the connection of physical devices through multiple industrial protocols via configuration. AWS IoT TwinMaker[41] enables IoT connectivity for different devices through communication protocol configuration. Emulate3D[50] supports multiple communication protocols such as PCC, Profibus, OPC, Modbus and CANopen, and can configure communication protocols for various PLC brands.
- **Physical data source access.** Real-time data collection can be achieved by connecting physical products/equipment through IoT. For example, Thingworx[45] can connect various smart devices to the IoT ecosystem for data collection. Software tools with similar functions include MindSphere[44] and Azure Digital Twins[42] built-in connector supports connection to physical space devices through IIoT, and AWS IoT TwinMaker[41] can be used to collect and analyze data on large-scale industrial equipment.
- **System data aggregation.** It's crucial to enable the digital twin system to obtain data from multiple information systems. For instance, the AWS IoT TwinMaker[41] Data Connector connects to AWS IoT SiteWise for asset model data and to Amazon S3 for resources like visual asset files and device documentation. TwinBuilder[40] incorporates a built-in connector to integrate the packaging SDK with the IIoT and send and receive operational data.
- **Data asset management.** It is necessary to support planning, controlling, developing, executing, and supervising data and information assets for the digital twin system.

(5) Virtual-reality interaction

When conducting virtual-reality interaction for digital twin, it is necessary to adhere to the following requirements to achieve synchronous interaction and feedback control between the virtual and physical actions.

- **Action behavior configuration.** The ability to synchronize the action of physical products/equipment with the virtual model is inevitable. For instance, the Unity Manufacturing Toolkit[49] realizes synchronization between virtual and real actions through the physical state data collected in real-time after configuring action constraints related to equipment/products. Azure Digital Twins[42] also has similar functionality.
- **State synchronization configuration.** The states consistency between virtual and real space should be guaranteed. For example, 3DEXPERIENCE[43] can maintain consistency between real and virtual assembly actions and states by connecting physical space data, and TwinBuilder[40] can simulate synchronization between real and virtual by collecting physical product state data in real-time.
- **Rule configuration.** It requires providing action and state rule configuration.
- **Model-data association configuration.** It aims to realize the interactive relationship between the model and the data in virtual-reality interaction.
- **Virtual-real consistency verification.** It needs to verify the consistency of the actions and states of the digital twin models and the physical entities.

(6) Virtual simulation

Users anticipate being able to conduct virtual simulations using the developed digital twin models, data, algorithms, and interactions among other components, to accurately replicate the properties and parameters of the physical space. The capability of the virtual simulation function should meet the following requirements.

- **Model configuration.** Before the virtual simulation, the relevant parameters of the digital twin models can be configured.
- **Algorithm configuration.** Similarly, it needs to support algorithm parameters configuration.
- **Data configuration.** It needs to support configuration of real-time and historical data.
- **Boundary configuration.** It is necessary to define the parameters that establish the boundaries of the simulation process.
- **Multidisciplinary simulation solver.** It is important that the model from different disciplines and domains should be solvable. For example, MindSphere[44] provides the function of modeling and simulating complex systems in a multidisciplinary domain for digital twin. TwinBuilder[40] supports

the integration and co-simulation of multi-physical domain system models in various ways. 3DEXPERIENCE[43] facilitates electromagnetic, material science, fluid, structure, and multi-body simulation. Emulate3D[50] offers support for the process simulation of complex automation systems. MIoT.VC[51] is employed for digital industrial simulation, such as 3D process simulation, assembly simulation, logistics simulation, robot simulation, and virtual commissioning. Altair Activate[52] combines physical and data-driven technologies to help users understand the overall performance of the system and identify the interrelationships between subsystems in the early stages of product development. Emulate3D[50] can achieve virtual simulation and debugging of the entire factory automation control system.

(7) Visualization

When conducting digital twin research and application, users anticipate a digital twin software platform that offers enhanced visual representations for digital twin models, data, algorithms, connectivity, interactions, and simulations.

- **Model visualization.** It provides visualization for the movement, position, and state of a 3D geometry model. For example, ANSYS supports embedded software integration with SCADE Display[53] for model visualization. ThingWorx[45] enables augmented reality bird's eye view to identify problems in a plant.
- **Data visualization.** Visualization for various data should be utilized to gain insight into problems and aid decision-making. For instance, AWS IoT TwinMaker[41] provides alerts based on data visualization to determine when data or forecasts are out of expectations. Predix[47] can be employed for data analysis and monitoring and offers asset performance management and operational optimization services. ThingWorx[45] monitors and alerts on equipment status with real-time data collection.
- **Interaction visualization.** Visualization for virtual-reality interaction is required. For example, 3DEXPERIENCE,[43] TwinBuilder,[40] and Simcenter[44] all support virtual-real interaction visualization.
- **Algorithm visualization.** It aims to provide visualization of the solving process and results of the algorithm.
- **Simulation visualization.** It needs to offer a visual representation of the process and results of diverse simulation models. For instance, 3DEXPERIENCE[43] uses the SIMULA software set to build a 3D simulation ecosystem and visualize the simulation. TwinBuilder[40] supports the visualization of multidisciplinary and multi-physics simulation processes.

As a fusion of fundamental scientific and engineering principles, industrial software integrates industrial technology with information technology. The successful execution of digital twin application is contingent upon the utilization of industrial software. Nowadays, numerous renowned software suppliers have developed digital twin software platforms based on their extensive technical expertise and business requirements. Nevertheless, due to factors such as digital twin insights, business domain scope, and their own technology, the digital twin software platforms in the market have significant limitations in terms of functionality.

(1) For researchers, the existing digital twin industrial software platforms are deemed inadequate as they are mainly designed for conventional algorithm testing, multi-physical finite element simulation, or general data storage and analysis. These platforms are often inclined towards specific domains, such as geometry modeling, mechanism analysis, or IoT platforms. Furthermore, there is a lack of a system tool platform that caters specifically to digital twin theory research, including the fusion of geometry-physics-behavior-rule for digital twin models, acquisition-transmission-processing-use for digital twin data, and complementary-iteration-coevolution for both digital twin model and data.

(2) For developers, in the realm of software development for digital twin, existing industrial platforms typically offer only a subset of the core functionalities, such as 3D visualization and interaction, finite element analysis, multi-physics field calculations, or machine learning algorithm design. To cater to distinct and customized digital twin functional specifications, it is often necessary to manually integrate multiple development and compatible tools for individual projects. At present, there's still a lack of a universal software development platform that is equipped with robust open-source, all-encompassing features and superior compatibility.

(3) For enterprise users, the current digital twin industrial software platforms are highly specialized, providing professional-grade multi-physical coupling analysis, diverse algorithms, and advanced 3D visualization development techniques. However, there is a significant challenge for common enterprise users who struggle to customize digital twin application service systems to align with their specific requirements. Consequently, a configurable, modular, and universally applicable digital twin development platform that can support the creation and implementation of digital twin application services is urgently required.

3. makeTwin: Platform architecture and core functions

Based on the deep exploration of digital twin theory and application, a software platform, makeTwin, is designed to meet the urgent demand for a universal digital twin software platform and to promote the implementation of digital twin theory. This section will focus on its design concept and ten core functions.

3.1. From the five-dimensional digital twin model to makeTwin

In the past few years, digital twin theory has gained significant attention from academia and industry, and the exploration of digital twin applications in various fields has entered a critical stage. Based on Grieves' three-dimensional digital twin model[3,] a five-dimensional model was proposed in 2017, which includes physical entity, virtual model, digital twin data, connection, and service.[22] Subsequently, the theory system of digital twin model,[11] data,[12] connection and interaction[13–14] have been successively proposed based on the five-dimensional model. Enabling technologies and tools for digital twin construction have also been explored to provide clear guidance for the digital twin construction method.[15] Moreover, a set of digital twin standard systems has been jointly established with standardization administration and application enterprises, focusing on digital twin understanding and communication, the research and implementation of key technologies, and industry applications.[24] With the increasing improvement of digital twin theory, scholars and engineers from various industries realize that digital twin applications should be based on industrial software to fully realize their industry value.

Under the guidance of the research on digital twin basic theories and common key technologies mentioned above, and inspired by the concept of "Make More Digital Twins",[1] a general, comprehensive, open, and scalable industrial software platform reference architecture, makeTwin, has been developed. The original design architecture and implementation have been worked out.[54] This platform aims to provide a research and development reference architecture for digital twin holistic construction and service deployment for different users and application scenarios.

3.2. *makeTwin: Platform architecture*

The architecture of makeTwin platform is comprised of ten fundamental functional modules, namely *twinModelBuilder*, *twinDataProcessor*, *twinAlgBuilder*, *twinIoTConnector*, *twinInteractor*, *twinSimulator*, *twinLibrary*, *twinVisualization*, *twinSceneTemplate*, and *twinAppDeployer*, as shown in Fig. 2. Each module provides interoperability with features such as configurability, callability, modifiability, and extendibility. The eight core functions of makeTwin (excluding *twinAppDeployer* and *twinLibrary*) can interact through customizable API. makeTwin aims to provide a public platform for digital twin theoretical innovation research and verification for academic researchers, offer a general digital twin configurable and reusable development platform for project developers, and supply convenient and reliable modular digital twin services for enterprise users.

3.3. *makeTwin: Ten core functions*

3.3.1. *twinModelBuilder*

The primary function of twinModelBuilder is to manage digital twin models. To achieve multidimensional, multiscale, and multidomain descriptions and characterizations of physical entities or complex systems, twinModelBuilder adheres to the theoretical system of digital twin model construction[11] and consists of two functional sub-modules, which are ModelContainer and ModelIntegration.

The ModelContainer comprises four essential functions, which are model import, model creation, model conversion, and model lightweight. Their specific features are described as follows.

- **Model import**. The model import function enables the importation of geometry models in diverse formats and is compatible **with** mechanism models (physics, behavior, rule models) developed by different languages.
- **Model creation.** The model creation function supports the creation of multidomain (thermodynamics, mechanics, electromagnetics, etc.) and multidimensional (physics, behavior, rule, etc.) mechanism models.
- **Model conversion.** The model conversion function facilitates the format conversion of various geometry models and is compatible with physics, behavior, and rule models developed by different languages.
- **Model lightweight.** The model lightweight function allows for removing non-essential features of 3D geometry models at varying levels of granularity.

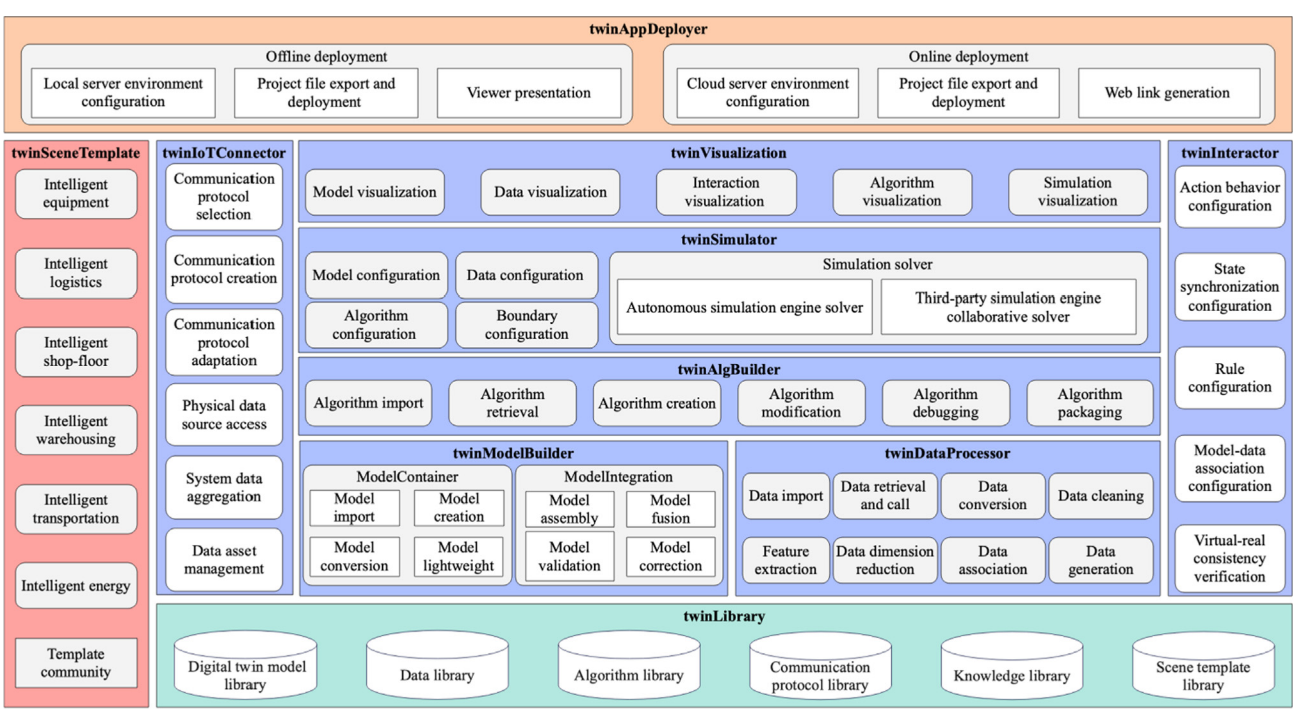

Fig. 2 Architecture for makeTwin software platform.

The ModelIntegration contains four additional functions, which are model assembly, model fusion, model validation, and model correction. Their specific features are described as follows.

- **Model assembly.** The model assembly function involves the process of assembling individual models, unit-level models, system-level models, and complex-system-level models in the spatial dimension. This process includes the creation of topological and constraint relationships.
- **Model fusion.** The model fusion function establishes coupling relationships between different disciplinary and domain models, mainly involving coupling relationship selection and model fusion.
- **Model validation.** The model validation function entails verifying the consistency between digital model output and physical object output based on various requirements, including individual model validation and fusion model validation.
- **Model correction.** The model correction function denotes the process of adjusting model parameters to rectify deviations. This primarily involves selecting correction parameters, rectifying deviations, and adjusting parameters.

3.3.2. twinDataProcessor

twinDataProcessor has been designed to equip makeTwin with advanced digital twin data management capabilities. To meet the criteria of digital twin data construction and processing, including complementary, standardization, timeliness, association, fusion, information growth, and service,[12] the twinDataProcessor module provides various features as follows.

- **Data import.** The data import function enables makeTwin to import various types of historical data of users, with compatibility for different mainstream data formats.
- **Data retrieval and call.** The function of data retrieval and call allows users to search and access specific data quickly.
- **Data cleaning.** The data cleaning function comprises data review and verification, deduplication, and error correction.
- **Data conversion.** The data conversion function supports the conversion of data from various sources into uniform formats.
- **Feature extraction.** The feature extraction function provides resources for users to extract key features from digital twin data, including causality, similarity, complementarity, etc.
- **Data association.** The data association function supports integrating data from different sources or databases into a unified database through shared fields or variables, enabling cross-database querying and data usage across different databases.
- **Data dimension reduction.** Data dimension reduction supports dimensionality reduction for high-dimensional data.
- **Data generation.** Data generation allows data augmentation through digital twin mechanism model simulation, digital twin data deduction, machine learning generation, etc.

3.3.3. twinAlgBuilder

The primary objective of twinAlgBuilder is to equip makeTwin with algorithm model development and management capabilities. To meet algorithm models' configurability, verifiability, and optimization requirements, twinAlgBuilder offers various features, including multi-language algorithm import, algorithm call, algorithm creation, algorithm modification, and algorithm debugging. The specific functionalities of these features are described as follows.

- **Algorithm import.** The algorithm import function allows for the import of algorithm models developed by mainstream third-party languages such as Python, MATLAB, C++, etc.
- **Algorithm retrieval.** The algorithm retrieval function supports the compatibility and retrieval of existing algorithm models in the algorithm library.
- **Algorithm creation.** The algorithm creation function empowers users to develop custom algorithm functions using the built-in Python editor.
- **Algorithm modification.** The algorithm modification function enables users to call existing algorithms from the library and modify algorithm structures and parameters using the algorithm function editor interface.
- **Algorithm debugging.** The algorithm debugging function is used for algorithm running and debugging, providing performance optimization related to model correctness, robustness, time complexity, space complexity, and other factors.
- **Algorithm packaging.** The algorithm packaging function is responsible for encapsulating the algorithm code, dependencies, and necessary data into a deployable unit that can be accessed through a set of API keys. Once encapsulated, other applications or systems can use the algorithm model without requiring knowledge of the underlying code or dependencies.

3.3.4. twinIoTConnector

twinIoTConnector plays a critical role in connecting the virtual and physical worlds. To meet the digital twin guidelines of accuracy, real-time, consistency, security, and reliability,[13] twinIoTConnector sets up the bridge to achieve virtual-reality interaction, with features such as communication protocol selection, communication protocol creation, communication protocol configuration, physical data source access, system data aggregation, and data asset management. Their specific functionalities are described as follows.

- **Communication protocol selection.** The communication protocol call function allows users to call and use existing protocols from the communication protocols library.
- **Communication protocol creation.** The communication protocol creation function authorizes users to customize communication protocols based on protocol statutes.
- **Communication protocol adaptation.** The communication protocol configuration function is responsible for configuring communication protocols for different physical devices.
- **Physical data source access.** The physical data source access function collects and aggregates heterogeneous physical data from multiple sources of various physical resources.
- **System data aggregation.** The system data aggregation function can access and collect enterprise information system data such as MES, ERP, WMS, etc.

- **Data asset management.** The data asset management function can assist the equipment management and visual statistical analysis of data assets.

3.3.5. twinInteractor

The core function of the twinInteractor module is to provide makeTwin with the capability of virtual-real interaction. The twinIoTConnector and twinInteractor collaborate to establish a bridge for virtual-real information exchange and sharing. They also deliver "data nutrients" for dynamic updates of digital twin models, real-time control of physical entities, and online optimization of decision-making solutions to accelerate information sharing and integration. Its critical features include action behavior configuration, state synchronization configuration, rule configuration, digital model association configuration, virtual-real consistency verification, etc. Their specific functionalities are described as follows.

- **Action behavior configuration.** The action behavior configuration function allows the configuration of action form, motion path, and motion coupling relationship for the 3D geometry model in the virtual space so as to realize the motion synchronization between the virtual and physical spaces.
- **State synchronization configuration.** The state synchronization configuration enables the configuration and updating of the operation state of physical resources into the virtual space, aiming to achieve state synchronization of the virtual and physical spaces.
- **Rule configuration.** This function module is designed to configure relevant rules and constraints for the geometry models' motion range, safety boundary, and interruption conditions in the virtual space.
- **Model-data association configuration.** The model-data association configuration function facilitates the association configuration between the digital twin model and enabling signals in virtual space.
- **Virtual-real consistency verification.** The virtual-real consistency verification function can be used to verify whether the virtual and reality actions or states are consistent with each other.

3.3.6. twinSimulator

The primary function of twinSimulator is to equip makeTwin with high-precision computing and simulation capabilities. The twinSimulator module offers two types of simulation engines, which are third-party and customized. They can perform multidisciplinary, multi-physics, multi-scale, and multi-probability simulations tailored to different application scenarios. The key features of the twinSimulator include model configuration, algorithm configuration, data configuration, and boundary configuration. Once these configurations are set up, a simulation can be conducted. Customized simulation engines are available for virtual simulations, such as fault diagnosis and prediction, scheduling and optimization, and path planning. For complex simulation problems that the existing simulation engine cannot solve, makeTwin incorporates third-party simulation engines to provide joint simulation solutions. In the case of multidisciplinary coupling simulation calculations, the ANSYS simulation interface can be employed for joint solution calculation. Additionally, Python development and debugging tools can be integrated and invoked by the makeTwin platform to run and solve Python-related algorithms.

3.3.7. twinVisualization

The core function of twinVisualization is to provide makeTwin 2D/3D visualization capabilities. The twinVisualization module includes five functions: model visualization, data visualization, interaction visualization, algorithm visualization, and simulation visualization.

- **Model visualization.** The model visualization function demonstrates the imported 3D models in the form of a graphical display. It is mainly used for model rendering, model static visualization, model information visualization, and model action visualization.
- **Data visualization.** The data visualization function converts the provided information or data into a visual context to make it easier and faster to understand. The visual context includes but is not limited to bar charts, line charts, scatter charts, pie charts, area charts, dashboard charts, Gantt charts, and tables.
- **Interaction visualization.** The interactive visualization function manipulates and understands data/models through dialogue and interaction between the human operator and the system. The interaction includes but is not limited to dialog boxes, checkboxes, buttons, sliders, knobs, voice, and gestures.
- **Algorithm visualization.** The algorithm visualization function describes complex algorithm models using graphical visualization. It includes but is not limited to visualization of algorithm structure, parameter debugging, execution process, and execution results.
- **Simulation visualization.** The simulation visualization function develops simulations for different scenarios by calling 3D model visualization, data visualization, interaction visualization, and algorithm visualization.

3.3.8. twinLibrary

twinLibrary is responsible for providing a comprehensive support system for makeTwin. It consists of a digital twin model library, an algorithm library, a data library, a knowledge library, a communication protocol library, and a scene template library. These functions are described as follows.

- **Digital twin model library.** The digital twin model library stores the digital twin models imported and developed by users, comprising 3D geometry, behavior, physics, and rule models.
- **Algorithm library.** The algorithm library contains algorithm models developed by users. It provides access to built-in general-purpose encapsulated algorithm models, such as data processing class models and intelligent optimization class models.
- **Data library.** The data library preserves physical data collected from physical resources, virtual data generated from simulation, processed feature data sets, and deduced data.

- **Knowledge library.** The knowledge library keeps knowledge documents, such as process documents, fault maintenance documents, and expert experience documents.
- **Communication protocol library.** The communication protocol library houses standard communication protocols for equipment virtual-real connection in different application scenarios.
- **Scene template library.** The scene template library stores and provides access to customized application scenario templates developed by users, shortening the development cycle when creating new digital twin application services.

3.3.9. twinSceneTemplate

The core function of twinSceneTemplate is to provide a collection of common scenario templates across various fields. These templates can be imported at the beginning of project development to assist users in quickly navigating digital twin application services. Once development is completed, the developed scenario can also be saved to the scene template library for future use. Users are able to combine the functions of monitoring, simulation, prediction, evaluation, optimization, and management services freely with twinSceneTemplate in different application scenarios. Currently, makeTwin offers a range of service scenarios, such as smart equipment, smart shop-floor logistics, stereo warehouse, and smart energy. Additionally, users have the ability to upload both general-purpose and specialized scenario templates to the scenario template community for broader usage and accessibility. For instance, the smart equipment application scenario includes the model of general-purpose equipment, communication protocols, and various service components required by the scenario. Users can choose the relevant functions based on their specific needs.

3.3.10. twinAppDeployer

twinAppDeployer is mainly designed to facilitate flexible deployment of makeTwin application services through multiple modes, including offline and online deployment. Offline deployment is geared towards devices which is lack Internet connectivity, whereby the digital twin application system is installed locally after configuring the environment on a designated computer or server. Conversely, online deployment is intended for devices lacking Internet connectivity, allowing users to upload digital twin application service files to a cloud server environment and obtain web links to access the application as web pages.

4. makeTwin: Workflow and interaction mechanism

The operation and application of the makeTwin software platform are particularly crucial. Therefore, this section gives the workflow and interaction mechanism of makeTwin.

4.1. makeTwin: Workflow

In the makeTwin software platform, researchers, developers, and enterprise users can choose two main development modes for the digital twin service program based on their application needs. The first mode is application service development based on scenario templates, and the second is customized construction, as shown in Fig. 3. The specific workflow is as follows.

Step 1: New project creation.

Step 2: Determination of using existing scene template or not. In this step, users can determine whether to use the scene templates or not based on the availability and actual application needs.

Step 3: Scene configuration. If scene templates are available and applicable, users can configure digital twin models, data, communication protocols, and algorithm models by accessing these scene template library. Some specialized models, data, communication protocols, and algorithm models can be customized and constructed. If not, users need to create and configure the scene with the help of various libraries. Then, the simulation solver and visualization are configured.

Step 4: Service App generation and deployment. In this step, users generate a service App through the makeTwin software platform and deploy the App.

Step 5: End.

4.2. makeTwin: Interaction mechanism

The interaction mechanism of makeTwin is the input and output correlation between various core functions. In the makeTwin platform, the interaction mechanism between twinModelBuilder, twinAlgBuilder, twinDataProcessor, twinIoTConnector, twinInteractor, twinSimulator, twinVisualization, twinLibrary, and twinSceneTemplate is mainly considered, as shown in Fig. 4.

(1) The main carrier for twinModelBuilder to interact with other modules is the digital twin model. twinModelBuilder inputs the generated digital twin models from the third-party software with various formats, the existing model from twinLibrary, the algorithms generated by twinAlgBuilder, and the verification information from twinSimulator. It outputs the newly generated digital twin models for twinInteractor, twinSimulator, twinVisualization, twinSceneTemplate and twinLibrary.

(2) The main carrier for twinDataProcessor to interact with other modules is data. twinDataProcessor inputs the collected data from twinIoTConnector, the simulation data from twinSimulator, and the historical data available in twinLibrary. It outputs the processed data to twinAlgBuilder, twinInteractor, twinSimulator, twinVisualization, and twinSceneTemplate.

(3) The main carrier for twinAlgBuilder to interact with other modules is the algorithm. twinAlgBuilder inputs the algorithm models available from twinLibrary, the real-time data offered by twinIoTConnector, the processed data from twinDataProcessor, and the algorithm validation information from twinSimulator. It outputs the new algorithm models for twinModelBuilder, twinSimulator, twinVisualization, twinSceneTemplate, and twinLibrary.

(4) The main carriers for twinIoTConnector to interact with other modules are communication protocols and data. twinIoTConnector inputs physical device data and enterprise system data obtained through communication protocols. It outputs real-time data for twinAlgBuilder, twinDataProcessor, twinSimulator, twinVisualization, twinLibrary, and twinSceneTemplate.

TABLE 1. Definitions of digital twin element technolgies.

Layer	Technology	Definition
Digital Virtualization	Real world data preprocessing technology	Preprocessing technology based on data classification for integrating heterogeneous complex data of different characteristics collected in the real world
	Multidimensional data causal relation analysis and integration technology	Database management and correlation analysis technology for integrating preprocessed heterogeneous complex data and transferring information between digital twin components
	Digital object distributed storage, processing, and analysis framework technology	Intelligence framework technology for distributed storage, high-speed processing, and analysis efficiency of large data about digital twin objects
	Data collection and processing technology	Technology that converts multi-format (e.g., text, table, photo, and video) data acquired in the real world into an unified form
	Multidimensional information and object visualization	Technology that derives a relationship between various data and displays various information through multidimensional visualization/multidimensional data visualization/multi-scale transformation
	Object identification technology	Technology to classify, measure, and track the objects constituting a physical space using RFID·USN·IoT technology
	Sensor placement optimization technology	Technology to reduce redundant information, improve sensor utilization, and maintain stability of collected data
	Virtual sensor technology	Technology to calculate physical areas that are difficult to measure with hardware using knowledge and algorithms (SW)
Digital Twin Synchronization	Space-time synchronization technology	Synchronization technology between the real world and virtual world through the connection between real objects and virtual objects (Physical to Virtual)
	Information update technology	Technology that minimizes the error between the real world and the virtual world by updating the changed information of the real world within a predetermined (or permitted) time through minimal resources in the already built virtual world
	Sensor-based actuator technology	High-speed and high-precision control technology for an output device (actuator) that reflects optimization information inferred from the virtual world
	Object cleaning technology	Technology that minimizes defective elements, such as redundancy, errors, and incomplete and delayed data
	Data and information effectiveness verification technology	Technology that detects and corrects errors in information collected in the real world and delivers the event to the management system
	High-speed/low-latency data transmission technology	Technology that delivers various heterogeneous complex data collected in the real world to the virtual world to minimize the temporal difference between the real and virtual world and synchronize them
	Data transmission management and load reduction technology	Network resource management and load reduction technology to stably deliver information collected in the real world to the virtual world

TABLE 2. Definitions of digital twin element technolgies.

Modeling & Simulation	Physics modeling technology	Multiple modeling and physics simulation technology including simulations of fluids, structures, air, acoustics, thermodynamics, etc.
	Behavior modeling technology	Technology to model and simulate virtual objects according to actual physical phenomena, motions, and behaviors through images, graphs, networks, etc.
	System rule technology	Technology that automatically models and simulates rules using large amounts of historical data and knowledge through big data and machine learning algorithms
	Automatic scenario generation and tailoring technology	Scenario automatic generation for modeling and simulation and model tuning technology for simulation by mission based on basic model sharing
	Digital twin simulation technology	Technology that reduces the probability of error occurrence, uncertainty, cost, etc. through simulation that reflects the real world such as multiphysics, multiscale, and probability
	Modeling verification and certification technology	Modeling and simulation verification and certification technology to maintain precision, sensitivity, and robustness
Federated Digital Twin	Digital twin identification system management technology	Classification, identification and management system operation technology of moving objects for joint collaboration between digital twin models
	Federation metadata creation and management technology	Technology that creates and manages metadata about the functions of various federators (elements) that make up the digital twin model
	Federation intelligence technology	Federation intelligence technology for efficient digital twin federation (e.g., data generation, management, learning, and matching)
	Digital twin mutual information exchange technology	Technology for exchanging and sharing information based on reliability/convenience of digital twins using information sharing technologies such as blockchain
	Federation governance technology	Purposeful context Life cycle management technology (such as creating, sharing, maintaining, renewing, and decommissioning) for federating with each other; and digital twin multi-role management technology for multi-role/role-specific federations
Intelligent Digital Twin Service	High-speed visualization and service information presentation technology (service display tech.)	High-speed visualization (including AR/VR/XR) and service information presentation technology to present digital twin model information including optimal information in the real world in various forms (e.g., number, diagram, and 2D/3D)
	Intelligent service resource management technology	Life cycle management technologies such as reliability and history of various resources (sensors, actuators, devices, systems, etc.) that make up the real world; and technology to efficiently collect, classify and organize digital twin resources (model/algorithm/data tools, etc.)
	Service search technology	Appropriate service search technology based on matching algorithms according to user needs
	Service evaluation technology	Objective evaluation techniques, such as service time, cost, reliability, feature similarity, security, maintainability, and satisfaction
	Fault detection technology	Fault elimination technology that tests faults that can be damaged due to service, demand, network, application, etc. and measures necessary in the event of a fault
	Service maintenance technology	Technology to provide service mobility (continuity) over time and space movement

Evaluating the need for real-time data processing and analytics capabilities within a digital twin system is crucial to ensure that the system can meet operational requirements and deliver timely insights. Here is the process to guide this evaluation.

1) Clarify the specific use cases and objectives of the digital twin where real-time capabilities are essential. Identify scenarios where timely data processing and analytics can enhance decision-making or operational efficiency.

2) Determine which processes, systems, or aspects of the physical environment require real-time monitoring, control, or predictive capabilities.

3) Evaluate the temporal requirements for data processing and analytics. Determine if there are critical time windows within which data insights must be generated and acted upon.

4) Assess how quickly decisions need to be made based on data insights derived from the digital twin. Consider operational impacts and consequences of delayed or real-time decision-making.

5) Identify the sources of data that will feed into the digital twin. Evaluate the frequency of data updates and the need for real-time data acquisition from sensors, IoT devices, or other sources.

6) Assess the complexity of integrating real-time data streams into the digital twin platform. Consider interoperability with existing systems, protocols for data transmission, and scalability of data ingestion.

7) Determine the computational speed required to process incoming data streams in real-time. Assess the processing power and capabilities needed to analyze data and generate insights promptly.

8) Evaluate the availability of computational resources (e.g., CPU, memory, storage) to support real-time data processing and analytics within the digital twin environment.

9) Assess the complexity of analytics required for real-time insights. Determine if simple threshold-based alerts are sufficient or if advanced analytics such as machine learning models are needed.

10) Evaluate whether the digital twin requires predictive analytics capabilities to forecast future outcomes or behaviors based on real-time data inputs.

11) Consult with stakeholders, including domain experts, operations managers, IT personnel, and end-users. Validate assumptions and gather insights into the criticality of real-time capabilities for their operational workflows.

12) Gather specific requirements and use case scenarios from stakeholders that illustrate the need for real-time data processing and analytics.

17) Research and evaluate technologies and platforms that support real-time data processing and analytics. Consider cloud-based solutions, edge computing capabilities, and scalable analytics frameworks.

18) Conduct a cost-benefit analysis to assess the economic feasibility of implementing real-time capabilities. Evaluate potential cost savings, operational efficiencies, and competitive advantages gained from real-time insights.

19) Based on the assessments and stakeholder inputs, make a decision on whether real-time data processing and analytics capabilities are necessary for the digital twin system.

20) Document the rationale behind the decision, including key considerations such as operational requirements, stakeholder feedback, technological feasibility, and cost implications.

21) Continuously monitor the performance of real-time data processing and analytics capabilities within the digital twin system. Measure adherence to operational requirements and user satisfaction.

22) Be prepared to adapt the digital twin system's capabilities based on evolving operational needs, technological advancements, and feedback from stakeholders.

Addressing security concerns related to data access, transmission, storage, and integrity within the digital twin environment, while ensuring compliance with relevant privacy regulations, is crucial for protecting sensitive information and maintaining trust. Here is a process to address these concerns.

1) Create a comprehensive inventory of all data sources and types that will be used in the digital twin environment.

2) Classify data based on sensitivity and criticality, such as public, internal, confidential, and restricted data.

3) Identify potential threats to data security, including unauthorized access, data breaches, data corruption, and insider threats.

4) Evaluate the system for vulnerabilities in hardware, software, network, and processes.

5) Assess the potential impact of identified threats on data confidentiality, integrity, and availability.

6) Implement RBAC to ensure that users have access only to the data necessary for their roles.

7) Use MFA to add an additional layer of security for accessing the digital twin environment.

8) Maintain detailed logs of all access and actions performed on the system to monitor and review suspicious activities.

9) Encrypt data in transit using protocols such as TLS/SSL to protect data from interception and tampering.

10) Implement secure API gateways and use authentication tokens to secure data transmission between systems.

11) Use Virtual Private Networks (VPNs) to create secure communication channels for remote access.

12) Encrypt data at rest using robust encryption standards like AES-256 to protect against unauthorized access.

13) Implement strict access controls to storage systems, including physical security measures for on-premises storage.

14) Ensure data redundancy and backup strategies to protect against data loss and ensure data availability.

15) Use hashing algorithms to verify the integrity of data. Store and compare hash values to detect any changes to the data.

16) Implement digital signatures to ensure data authenticity and integrity, especially for sensitive transactions.

17) Continuously monitor data integrity using automated tools to detect and respond to integrity violations.

18) Identify and understand relevant privacy regulations such as GDPR, CCPA, HIPAA, etc., applicable to the organization and data.

19) Collect and process only the minimum amount of personal data necessary for the digital twin's purpose.

20) Implement techniques to anonymize or pseudonymize personal data to protect individual identities.

21) Develop and enforce security policies that define how data should be handled, accessed, and protected.

22) Conduct regular security training and awareness programs for employees to understand and adhere to security best practices.

23) Establish an incident response plan to quickly and effectively respond to security breaches and data incidents.

24) Implement continuous monitoring tools to detect and respond to security threats in real-time.

25) Conduct regular security audits and assessments to ensure compliance with policies and identify areas for improvement.

26) Perform regular penetration testing to identify and address security vulnerabilities before they can be exploited.

27) Maintain comprehensive documentation of all security measures, protocols, and procedures implemented.

28) Generate and maintain reports required for regulatory compliance, including data protection impact assessments (DPIAs) and audit reports.

29) Establish a feedback loop to continuously improve security measures based on audit findings, incident reports, and technological advancements.

Considering scalability requirements to accommodate growth in data volume and complexity, while evaluating performance expectations for real-time responsiveness, is essential for the successful implementation and longevity of a digital twin system. Here is a process to address these aspects.

1) Estimate the expected growth in data volume and complexity over time. Consider factors such as the addition of new data sources, increased data granularity, and the expansion of the digital twin's scope.

2) Define key performance metrics for the digital twin, such as data processing speed, response time, throughput, and latency. Establish acceptable thresholds for these metrics.

3) Evaluate the current data volume, data processing requirements, and system performance. Identify bottlenecks and areas where performance is currently suboptimal.

4) Project future workloads based on anticipated growth in data volume and complexity. Consider scenarios such as increased sensor deployments, more frequent data updates, and the introduction of new analytical models.

5) Assess the feasibility of horizontal scaling, which involves adding more machines or nodes to distribute the load. Consider cloud-based solutions and distributed computing architectures.

6) Evaluate vertical scaling options, such as upgrading existing hardware with more powerful processors, additional memory, and faster storage solutions.

7) Explore data partitioning and sharding techniques to divide large datasets into smaller, more manageable pieces that can be processed in parallel.

8) Evaluate scalable storage solutions such as distributed file systems, cloud storage services, and data lakes. Ensure these systems can handle the projected data volume growth.

9) Choose databases that support scalability features like replication, sharding, and clustering. Consider NoSQL databases for unstructured data and time-series databases for sensor data.

10) Use caching mechanisms to store frequently accessed data in memory, reducing the need for repeated data retrieval from slower storage.

11) Implement load balancing to distribute incoming data and processing tasks evenly across available resources, preventing any single resource from becoming a bottleneck.

12) Optimize data processing algorithms and analytical models to improve efficiency and reduce computation time.

13) Implement stream processing frameworks such as Apache Kafka, Apache Flink, or AWS Kinesis to handle real-time data streams efficiently.

14) Consider edge computing solutions to process data closer to its source, reducing latency and offloading work from central servers.

15) Ensure that the network infrastructure supports low-latency communication between data sources, processing nodes, and end-users.

16) Conduct load testing by simulating expected future workloads. Use tools like Apache JMeter, Gatling, or Locust to generate synthetic data and measure system performance.

17) Benchmark the performance of different system components under varying load conditions. Identify performance bottlenecks and areas for optimization.

18) Implement real-time monitoring tools to track system performance, resource utilization, and data processing metrics. Use platforms like Prometheus, Grafana, or Datadog.

19) Configure auto-scaling policies to automatically adjust resources based on real-time demand. This is especially useful in cloud environments where resources can be dynamically allocated.

20) Regularly review the scalability plan to ensure it aligns with the evolving needs of the digital twin system. Update projections and strategies based on actual system performance and growth trends.

21) Gather feedback from stakeholders, including IT personnel, data engineers, and end-users, to identify additional scalability requirements and performance expectations.

22) Document the scalability and performance evaluation process, including identified requirements, chosen strategies, test results, and implemented solutions.

23) Communicate the findings and plans to relevant stakeholders to ensure alignment and support for scalability initiatives.

Assessing the interoperability requirements with existing systems, both within the organization and potentially with external partners or suppliers, is essential for the successful integration and functionality of a digital twin system. Here is the process to address these interoperability requirements.

1) Create a comprehensive inventory of existing systems, both internal and external, that the digital twin needs to interact with. This includes enterprise resource planning (ERP) systems, manufacturing execution systems (MES), customer relationship management (CRM) systems, IoT platforms, and external partner systems.

2) Identify all stakeholders involved in the interoperability process, including IT personnel, system administrators, business managers, and external partners or suppliers.

3) Clearly define the objectives for integrating the digital twin with existing systems. Determine the desired outcomes, such as data sharing, real-time updates, process automation, or enhanced decision-making capabilities.

4) Identify specific use cases that require interoperability, such as supply chain coordination, predictive maintenance, operational efficiency improvements, or customer support enhancements.

5) Review the existing interfaces and APIs of the systems that need to be integrated. Understand the communication protocols, data formats, and standards used by these systems.

6) Map out the current data flows between systems, identifying data sources, destinations, and the frequency of data exchanges. Highlight any data transformation or processing that occurs during these exchanges.

7) Assess the compatibility of data formats, communication protocols, and standards between the digital twin system and existing systems. Identify any mismatches or areas where data conversion may be necessary.

8) Ensure adherence to relevant industry standards and protocols (e.g., OPC UA, MQTT, RESTful APIs) to facilitate seamless integration and interoperability.

9) Identify technical challenges that may hinder interoperability, such as differences in data schemas, inconsistent data quality, varying communication protocols, or legacy system constraints.

10) Recognize organizational challenges, such as resistance to change, lack of skilled personnel, or insufficient documentation of existing systems.

11) Design an integration architecture that outlines how the digital twin will interact with existing systems. Consider using middleware, data hubs, or integration platforms to facilitate data exchanges.

12) Develop or adapt APIs to ensure seamless communication between the digital twin and other systems. Ensure APIs are secure, scalable, and well-documented.

13) Implement data mapping and transformation processes to align data formats and structures between systems. Use ETL (Extract, Transform, Load) tools if necessary.

14) Ensure that data exchanges between systems are secure. Implement encryption, authentication, and authorization mechanisms to protect sensitive data.

15) Verify that interoperability solutions comply with relevant regulations and standards, such as GDPR, CCPA, or industry-specific regulations. Ensure data privacy and protection measures are in place.

16) Perform thorough integration testing to ensure that the digital twin system communicates effectively with existing systems. Test for data accuracy, timeliness, and consistency.

17) Involve stakeholders in UAT to validate that the integrated systems meet business requirements and user expectations. Address any issues or discrepancies identified during testing.

18) Implement monitoring tools to track the performance and reliability of the integrated systems. Monitor data exchanges, system performance, and user interactions.

19) Continuously optimize the interoperability solutions based on monitoring insights and feedback. Address any emerging issues promptly and make necessary adjustments to improve performance and scalability.

20) Document the interoperability solutions, including system interfaces, APIs, data mappings, security measures, and testing procedures. Ensure documentation is accessible to all relevant stakeholders.

21) Provide training to IT personnel, system administrators, and end-users on the interoperability solutions. Ensure they understand how to use and maintain the integrated systems effectively.

Planning for the lifecycle management of a digital twin system, including updates, maintenance, and version control of models and software, is crucial to ensure its long-term effectiveness and reliability. Here is a process to manage the lifecycle of a digital twin system.

1) Clearly define the objectives of lifecycle management for the digital twin system. This includes ensuring system reliability, maintaining accuracy, and incorporating continuous improvements.

2) Identify all stakeholders involved in the lifecycle management process, including IT personnel, data scientists, domain experts, and end-users.

3) Define the key phases of the digital twin lifecycle, such as development, deployment, operation, maintenance, and retirement.

4) Assign roles and responsibilities for each phase, ensuring that all tasks have clear ownership.

5) Implement a version control system (e.g., Git) to manage changes to models, software, and configurations. Ensure that all updates and modifications are tracked.

6) Develop a branching strategy to manage parallel development, testing, and deployment. This includes main branches for stable releases and feature branches for ongoing developments.

7) Use configuration management tools (e.g., Ansible, Puppet, Chef) to automate and standardize the configuration of the digital twin environment.

8) Schedule regular updates for the digital twin system, including software patches, model recalibrations, and data integrations. Develop a calendar for these updates.

9) Establish maintenance windows to perform updates and maintenance tasks with minimal disruption to operations.

10) Implement automated testing frameworks to validate updates and ensure they do not introduce new issues. This includes unit tests, integration tests, and performance tests.

11) Establish a formal process for submitting and reviewing change requests. This includes evaluating the impact, risk, and benefits of proposed changes.

12) Create an approval workflow for changes, involving relevant stakeholders to ensure that all perspectives are considered before implementation.

13) Plan and execute changes in a controlled manner, ensuring that all modifications are documented and communicated to relevant parties.

14) Define key performance metrics to monitor the health and performance of the digital twin system. This includes accuracy, response time, and resource utilization.

15) Implement monitoring tools to continuously track system performance. Use dashboards and alerts to identify and address issues proactively.

16) Conduct regular performance reviews to evaluate the effectiveness of the digital twin and identify areas for improvement.

17) Implement data validation procedures to ensure the accuracy and consistency of data inputs and outputs within the digital twin system.

18) Develop and maintain a backup and recovery plan to protect against data loss and ensure business continuity.

19) Develop a scalability strategy to accommodate growth in data volume, complexity, and user demand. This includes both horizontal and vertical scaling approaches.

20) Ensure that the digital twin system is designed with future expansion in mind, allowing for the integration of new technologies, data sources, and use cases.

21) Maintain comprehensive documentation for all aspects of the digital twin system, including architecture, models, configurations, and operational procedures.

22) Provide ongoing training for stakeholders to ensure they are familiar with the digital twin system, its capabilities, and lifecycle management practices.

23) Establish mechanisms for collecting feedback from users and stakeholders. Use this feedback to inform updates and improvements.

24) Foster a culture of continuous improvement, regularly reviewing and refining lifecycle management practices to enhance the digital twin system.

Conducting a thorough cost-benefit analysis is essential to justify the investment in developing and maintaining a digital twin system. This process involves identifying and quantifying the costs and benefits associated with the system, then comparing them to assess the overall value. Here is a process to conducting a cost-benefit analysis.

1) Clearly define the scope of the digital twin project, including the systems, processes, and business units it will impact.

2) Outline the specific objectives and expected outcomes of the digital twin system, such as improved operational efficiency, predictive maintenance, enhanced decision-making, or cost savings.

3) Costs related to developing or purchasing the digital twin software, including licensing fees.

4) Expenses for servers, storage, sensors, IoT devices, and other hardware.

5) Costs for integrating the digital twin with existing systems and data sources.

6) Fees for consultants, vendors, and third-party services.

7) Salaries and benefits for the team managing and maintaining the digital twin.

8) Costs for training employees on the new system.

9) Ongoing costs for software licenses and cloud services.

10) Costs for maintaining and upgrading hardware.

11) Expenses related to data storage, processing, and management.

12) Costs for implementing and maintaining security measures.

13) Potential downtime during implementation and integration phases.

14) Temporary reduction in productivity as users adapt to the new system.

15) Reduction in operational costs due to improved efficiency, reduced downtime, and optimized resource usage.

16) Increased revenue from enhanced product quality, faster time-to-market, or new business opportunities.

17) Cost savings from reduced equipment failures and maintenance costs.

18) Enhanced decision-making capabilities through real-time data insights and predictive analytics.

19) Improved customer satisfaction and loyalty due to better product quality and reliability.

20) Gaining a competitive edge through innovation and improved operational capabilities.

21) Fostering a culture of innovation and continuous improvement.

22) Increased employee satisfaction and retention due to improved tools and processes.

23) Enhanced compliance with regulations and better risk management.

24) Assign monetary values to each identified cost and benefit. Use historical data, market research, and expert estimates to determine these values.

25) Define the time frame for the cost-benefit analysis, typically ranging from a few years to the expected lifespan of the digital twin system.

26) Calculate the NPV of the project by discounting future costs and benefits to their present value using an appropriate discount rate. NPV = (Present Value of Benefits) - (Present Value of Costs).

27) Calculate the ROI to assess the profitability of the investment. ROI = (Net Benefits / Total Costs) * 100.

28) Perform sensitivity analysis to understand how changes in key assumptions (e.g., cost estimates, benefit projections, discount rate) impact the NPV and ROI. Identify the factors that have the most significant effect on the results.

29) Identify potential risks associated with the digital twin project, such as technological challenges, data security issues, or resistance to change.

30) Develop strategies to mitigate identified risks and include contingency plans in the cost-benefit analysis.

31) Compile the findings into a comprehensive report that includes the scope, objectives, identified costs and benefits, NPV, ROI, sensitivity analysis, and risk assessment.

32) Prepare a presentation to communicate the results to stakeholders, including executives, business managers, and technical teams. Highlight the key benefits, justify the investment, and address any concerns.

33) Engage stakeholders in reviewing the cost-benefit analysis findings and incorporate their feedback.

34) Based on the analysis and stakeholder input, make an informed decision on whether to proceed with the development and maintenance of the digital twin system.

Identifying and mitigating potential risks associated with the implementation and operation of a digital twin system involves a comprehensive risk management process. This process encompasses technical, business, and organizational risks. Here is the process to manage these risks.

1) Risks associated with integrating the digital twin with existing systems and data sources.

2) Issues related to the accuracy, completeness, and timeliness of the data feeding the digital twin.

3) Challenges in scaling the digital twin to handle increasing data volume and complexity.

4) Risks of the digital twin not meeting performance expectations, such as response times and processing speeds.

5) Vulnerabilities in data access, transmission, storage, and overall system security.

6) Risks of the project exceeding the budget due to unforeseen expenses.

7) Uncertainty about achieving the projected benefits and ROI.

8) Changes in market conditions that may impact the relevance or value of the digital twin.

9) Risks of non-compliance with industry regulations and standards.

10) Resistance from employees or stakeholders to adopting the digital twin system.

11) Lack of necessary skills and expertise among the staff to develop, operate, and maintain the digital twin.

12) Risks associated with project planning, execution, and coordination.

13) Misalignment of goals and expectations among different stakeholders.

14) Assess the likelihood of each identified risk occurring and its potential impact on the project and organization.

15) Prioritize risks based on their assessed probability and impact. Use a risk matrix to categorize risks as high, medium, or low priority.

16) Conduct thorough integration testing to ensure compatibility with existing systems and data sources.

17) Implement robust data governance practices to ensure data quality and reliability.

18) Design the digital twin with scalability in mind, using modular architecture and scalable infrastructure.

19) Continuously monitor and optimize the performance of the digital twin system.

20) Implement comprehensive cybersecurity measures, including encryption, access controls, and regular security audits.

21) Establish a detailed budget with contingency funds to handle unforeseen expenses.

22) Regularly track and measure the benefits realized from the digital twin to ensure alignment with projected outcomes.

23) Conduct regular market analysis to stay informed about changes and adjust strategies accordingly.

24) Stay updated on relevant regulations and ensure the digital twin system adheres to all compliance requirements.

25) Develop and implement a comprehensive change management strategy to facilitate smooth adoption.

26) Provide training programs to equip staff with the necessary skills and knowledge to use and maintain the digital twin.

27) Apply project management best practices, including regular progress reviews, risk assessments, and stakeholder communication.

28) Engage stakeholders throughout the project to ensure alignment and address concerns proactively.

29) Develop a detailed action plan for implementing the risk mitigation strategies. Assign responsibilities and timelines for each action.

30) Allocate necessary resources, including budget, personnel, and technology, to support the implementation of mitigation strategies.

31) Continuously monitor the effectiveness of mitigation strategies and adjust them as needed based on changing conditions and new insights.

32) Conduct regular risk reviews to assess the current risk landscape and the effectiveness of mitigation strategies.

33) Prepare risk reports and communicate them to stakeholders, highlighting key risks, mitigation actions, and any changes in risk status.

34) Collect feedback from stakeholders on the risk management process and incorporate it into future planning.

35) Document lessons learned from the risk management process and apply them to future projects.

36) Continuously refine and improve the risk management framework to enhance its effectiveness and adaptability.

Assessing the readiness of an organization to adopt and leverage a digital twin system involves evaluating various factors such as skills, resources, and cultural acceptance. Here is the process to conduct this assessment.

1) Identify the technical and domain-specific skills required to develop, implement, and maintain the digital twin system.

2) Determine the necessary resources, including technology, infrastructure, budget, and personnel.

3) Evaluate the organization's culture and its openness to adopting new technologies and processes.

4) Create an inventory of existing skills and expertise within the organization. This includes technical skills (e.g., software development, data analytics, IoT) and domain-specific knowledge.

5) Compare the required skills for the digital twin project with the current skills inventory. Identify any gaps that need to be addressed.

6) Determine the training and development programs needed to bridge identified skills gaps.

7) Assess the current state of technology and infrastructure. Identify any upgrades or additional investments needed to support the digital twin system.

8) Evaluate the budget available for the digital twin project, including development, implementation, and ongoing maintenance costs.

9) Assess the availability of personnel required for the project, including project managers, developers, data scientists, and domain experts.

10) Evaluate the level of support and commitment from leadership and key stakeholders for the digital twin initiative.

11) Conduct surveys or focus groups to gauge employee attitudes toward adopting new technologies and processes. Identify any resistance or concerns.

12) Assess the organization's experience and capability in managing change. Determine if there are established change management processes and resources.

13) Organize workshops with key stakeholders to discuss the digital twin project, its objectives, and its potential impact on the organization.

14) Collect feedback on perceived challenges, opportunities, and readiness from different departments and teams.

15) Use workshops to identify specific indicators of readiness or lack thereof in skills, resources, and cultural acceptance.

16) Compile the findings from the skills assessment, resource evaluation, and cultural readiness assessment into a comprehensive report.

17) Assign a readiness score to each area (skills, resources, cultural acceptance) based on the assessment findings.

18) Provide recommendations for addressing gaps and enhancing readiness. This includes training programs, resource allocation, and change management strategies.

19) Develop and implement training programs to address skills gaps. Consider partnerships with educational institutions or hiring new talent.

20) Allocate necessary resources to address any deficiencies in technology, infrastructure, or budget.

21) Develop a comprehensive change management plan to ensure smooth adoption. This includes communication strategies, leadership engagement, and employee support initiatives.

22) Continuously monitor the progress of readiness improvement initiatives. Use key performance indicators (KPIs) to track success.

23) Conduct periodic reassessments to ensure that the organization remains ready as the project evolves and new challenges arise.

24) Establish a feedback loop to continuously gather insights from stakeholders and adjust strategies as needed.

25) Provide regular updates to stakeholders on the readiness assessment findings and progress of mitigation strategies.

26) Involve stakeholders in decision-making processes to ensure their buy-in and support throughout the project lifecycle.

27) Celebrate milestones and successes to build momentum and reinforce positive cultural change.

Planning for change management activities is crucial to ensure the smooth adoption of a digital twin system across an organization. This process involves preparing, managing, and reinforcing changes related to the new system. Here is the process to plan and execute change management activities.

1) Clearly outline the objectives and goals of the change management process, aligning them with the overall goals of the digital twin project.

2) Identify all key stakeholders affected by the digital twin system, including executives, managers, employees, and external partners.

3) Analyze the impact of the digital twin system on different parts of the organization, identifying areas that will undergo significant changes.

4) Form a dedicated change management team consisting of change managers, communication specialists, HR representatives, and project leaders.

5) Clearly define the roles and responsibilities of each team member in the change management process.

6) Create a governance structure to oversee change management activities, including decision-making processes and escalation paths.

7) Conduct surveys and interviews with employees and stakeholders to assess their readiness for change and identify potential resistance points.

8) Develop metrics to measure change readiness, such as awareness levels, perceived benefits, and willingness to adopt the new system.

9) Analyze the results of the readiness assessment to identify strengths, weaknesses, and areas that need additional support.

10) Define the objectives of the communication plan, such as informing, engaging, and motivating stakeholders.

11) Develop clear and consistent key messages about the digital twin system, its benefits, and its impact on the organization.

12) Identify the most effective communication channels for different stakeholder groups, such as emails, newsletters, meetings, intranet, and social media.

13) Create a detailed communication schedule, outlining when and how key messages will be delivered.

14) Conduct a training needs assessment to identify the specific skills and knowledge required for using the digital twin system.

15) Develop a comprehensive training curriculum that includes hands-on workshops, e-learning modules, user manuals, and FAQs.

16) Plan the delivery of training programs, ensuring that they are accessible to all employees and tailored to different learning styles.

17) Establish ongoing support mechanisms, such as help desks, user forums, and regular Q&A sessions, to assist employees during the transition.

18) Run pilot programs or small-scale implementations to test the digital twin system and change management strategies. Gather feedback and make necessary adjustments.

19) Engage stakeholders through regular meetings, workshops, and feedback sessions to ensure their involvement and address any concerns.

20) Continuously monitor the progress of change management activities, using KPIs and feedback to assess effectiveness and make improvements.

21) Celebrate key milestones and achievements to recognize the efforts of employees and reinforce positive behavior.

22) Establish a feedback loop to continuously gather input from stakeholders and make iterative improvements to the change management process.

23) Develop a sustainment plan to ensure that the changes are embedded in the organization’s culture and processes. This includes regular reviews, refresher training, and ongoing communication.

24) Evaluate the outcomes of the change management process against the defined objectives and goals. Identify successes and areas for improvement.

25) Document lessons learned throughout the change management process to inform future initiatives and improve organizational change capability.

26) Promote a culture of continuous improvement by regularly reviewing and updating change management practices based on feedback and new insights.

Defining metrics and KPIs to measure the performance and effectiveness of a digital twin system involves a systematic approach to ensure alignment with the system's intended objectives. Here is the process.

1) Clearly define the intended objectives of the digital twin system, such as improving operational efficiency, enhancing predictive maintenance, or reducing costs.

2) Ensure alignment with key stakeholders, including business managers, domain experts, and end-users, to understand their expectations and requirements.

3) Metrics that measure improvements in process efficiency, equipment uptime, and resource utilization.

4) Metrics that track the accuracy and effectiveness of predictive maintenance activities, such as reduction in unplanned downtime.

5) Metrics that quantify cost savings achieved through the digital twin system.

6) Metrics related to improvements in product quality and consistency.

7) Metrics that measure the impact of data-driven decision-making enabled by the digital twin.

8) Reduction in the time required to complete a specific process.

9) Improvement in the utilization rates of equipment and labor.

10) Increase in the number of units produced or processed within a given timeframe.

11) Increase in the average time between equipment failures.

12) Decrease in unplanned downtime due to predictive maintenance.

13) Reduction in maintenance costs resulting from predictive maintenance activities.

14) Reduction in overall operational costs, including energy, materials, and labor.

15) Decrease in inventory holding costs through optimized inventory management.

16) Reduction in the rate of product defects or rework.

17) Increase in the yield of high-quality products.

18) Improvement in the accuracy of decisions based on insights from the digital twin.

19) Reduction in the time required to make informed decisions.

20) Use historical performance data as a baseline to set realistic targets for each metric.

21) Compare with industry benchmarks to ensure targets are competitive and achievable.

22) Collaborate with stakeholders to set mutually agreed-upon targets.

23) Identify the data sources required to measure each metric, including sensors, IoT devices, and enterprise systems.

24) Implement systems and processes for accurate and timely data collection.

25) Develop analytical models and tools to process and analyze the collected data.

26) Create dashboards to visualize metrics and KPIs in real-time.

27) Establish a reporting cadence (e.g., weekly, monthly) to review performance against targets.

28) Implement alert mechanisms to notify stakeholders of any significant deviations from targets.

29) Conduct regular performance reviews with key stakeholders to evaluate the effectiveness of the digital twin system.

30) Establish a feedback loop to gather insights and suggestions for improvement from end-users and stakeholders.

31) Continuously refine metrics, KPIs, and processes based on feedback and changing business needs.

32) Document the defined metrics, KPIs, data sources, collection methods, and analysis techniques.

33) Regularly communicate performance results and insights to all relevant stakeholders to ensure transparency and alignment.

34) Provide training and support to stakeholders to help them understand and utilize the metrics and KPIs effectively.

A number of activities have to be performed before an organisation undergoes the development of a digital twin (DT) project. DT systems exist to help organisations with the lifecycle management of physical entities, non-physical entities and/or processes. The power of DTs is their ability to model virtually every aspect of an organisation, because every organisation has either physical or non-physical assets and processes.

1) The organisation can review existing reports, performance metrics, operational procedures, documentation that may provide insights into issues faced relating to any aspect that can be modelled by DTs. Recurring problems or areas where performance targets are consistently not being met are a useful starting point.

2) Of course, direct communication with people involved with the organisation is as valuable as report/documentation analysis. People can answer direct questions about their experiences, pain points, frustrations, and areas where they believe improvements are needed.

a) Organize workshops or focus groups with people from different departments or functional areas. Facilitate discussions around specific topics or processes to uncover issues and challenges faced by participants. Encourage brainstorming and idea sharing to gain diverse perspectives.

b) Distribute surveys to gather feedback on challenges they encounter in their roles. Ensure that survey questions are specific and relevant to the areas of interest, such as operational efficiency, quality control, maintenance issues, customer satisfaction, etc.

c) Utilize data analytics tools to analyze quantitative data related to performance metrics, production outputs, maintenance records, customer complaints, downtime incidents, etc. Look for trends, anomalies, or areas where performance falls below expectations.

d) Conduct on-site observations or job shadowing to observe firsthand how tasks and processes are carried out. Look for inefficiencies, bottlenecks, workflow interruptions, or areas where manual processes could be automated or improved.

e) Compare the organization’s performance and operational practices against industry benchmarks or best practices. Identify areas where the organization lags behind or where there is potential for improvement based on industry norms.

f) Review customer feedback, complaints, and suggestions to identify pain points in product or service delivery. Customer insights can provide valuable perspectives on areas that need attention to enhance satisfaction and loyalty.

3) Once the improvement areas have been identified, the corresponding stakeholders can also be identified. These stakeholders could include operational managers, scientists, design engineers, production engineers, frontline workers, IT personnel, customer service representatives, and others directly involved in daily operations. Stakeholders will be involved (in different capacities and degrees) in the DT project.

4) Once challenges and pain points have been identified through the above methods, prioritize them based on severity, impact on business outcomes, feasibility of addressing them, and alignment with strategic goals. Validate findings through discussions with stakeholders and data-driven analysis where possible.

5) Document the finalised business objectives clearly and comprehensively. Communicate these objectives to all relevant stakeholders to ensure everyone understands the purpose and expected outcomes of the digital twin initiative.

6) Ensure that the objectives are SMART:

a) Specific* Clearly defined and focused.

b) Measurable: Able to be quantitatively or qualitatively measured.

c) Achievable: Realistic and feasible given the resources and constraints.

d) Relevant: Aligned with the strategic goals of the organization.

e) Time-bound: Have a timeframe or deadline for achievement.

Developing a long-term roadmap for the evolution and expansion of a digital twin system involves strategic planning to align with future business needs and technological advancements. Here is the process to create such a roadmap.

1) Evaluate the current capabilities, performance, and usage of the existing digital twin system.

2) Define the desired future state of the digital twin system in alignment with business objectives and emerging technologies.

3) Identify key business drivers that will influence the evolution of the digital twin system, such as growth strategies, market demands, and operational efficiencies.

4) Engage stakeholders to gather requirements for future functionalities, enhancements, and capabilities of the digital twin system.

5) Research and assess emerging technologies that could enhance the capabilities of the digital twin system, such as AI/ML, IoT advancements, and cloud computing.

6) Evaluate scalability requirements and potential integration with other systems and technologies within the organization.

7) Define clear strategic objectives for the digital twin system roadmap, such as improving decision-making, enabling new business models, or enhancing customer experience.

8) Establish SMART (Specific, Measurable, Achievable, Relevant, Time-bound) goals to guide the roadmap development and implementation.

9) Divide the roadmap into phases or milestones that outline the evolution of the digital twin system over time.

10) Define specific initiatives and projects needed to achieve each phase or milestone, including timelines, resources, and dependencies.

11) Prioritize initiatives based on strategic alignment, business impact, and feasibility.

12) Develop a timeline for the implementation of each phase or milestone in the roadmap.

13) Allocate resources, including budget, personnel, and technology investments, required for each initiative.

14) Identify potential risks and develop mitigation strategies to minimize disruptions during implementation.

15) Define a governance structure to oversee the implementation of the roadmap, including roles, responsibilities, and decision-making processes.

16) Engage key stakeholders throughout the development and implementation process to ensure alignment and support.

17) Establish metrics and KPIs to monitor the progress and success of each initiative and milestone.

18) Conduct regular reviews and evaluations of the roadmap's implementation progress, adjusting strategies and priorities as needed.

19) Implement a feedback mechanism to gather input from stakeholders and end-users to inform continuous improvement.

20) Develop a communication plan to regularly update stakeholders on the progress of the roadmap implementation, achievements, and upcoming milestones.

21) Continuously iterate on the roadmap based on feedback, changing business needs, and technological advancements to ensure its relevance and effectiveness.

22) Document all aspects of the roadmap, including objectives, goals, initiatives, timelines, and governance structure.

23) Archive the roadmap documentation for future reference and as a historical record of decisions and actions taken.

Gathering and analyzing requirements from all stakeholders is critical for the successful development of a new digital twin project. There is a number of activities required to accomplish this.

1) Create a stakeholder analysis matrix to identify all relevant stakeholders for the digital twin project. This includes business managers, domain experts (e.g., engineers, scientists), end-users (e.g., operators, maintenance personnel), and IT personnel (e.g., data architects, system administrators).

2) Schedule and conduct one-on-one interviews with key stakeholders to understand their roles, responsibilities, and perspectives regarding the digital twin project. Use open-ended questions to explore their requirements, expectations, and challenges.

3) Organize workshops or focus groups with cross-functional teams of stakeholders to facilitate discussions on specific topics related to the digital twin. Encourage collaboration, brainstorming, and idea generation.

4) Design and distribute surveys to stakeholders to gather structured feedback on their requirements and preferences for the digital twin project. Ensure that survey questions are clear, relevant, and aligned with project objectives.

5) Analyze existing documentation such as business process manuals, technical specifications, system architecture diagrams, and user manuals. Identify relevant information that can provide insights into stakeholder requirements.

6) Conduct observational studies to observe stakeholders performing their daily tasks related to the processes that will be modeled or monitored by the digital twin. Note pain points, inefficiencies, and areas for improvement.

7) Research industry standards, best practices, and benchmarks related to digital twins and similar technologies. Use this information to validate stakeholder requirements and propose recommendations.

8) Develop use case scenarios in collaboration with stakeholders to illustrate how the digital twin will be used in real-world situations. Use these scenarios to validate and refine stakeholder requirements.

9) Create prototypes or demonstrations of the digital twin concept to showcase to stakeholders. Gather feedback on usability, functionality, and alignment with their requirements.

10) Document all gathered requirements in a structured manner using tools such as requirement management software, spreadsheets, or dedicated templates. Clearly capture stakeholder needs, priorities, dependencies, and any constraints.

11) Collaboratively prioritize requirements with stakeholders based on their criticality, impact on business objectives, feasibility, and technical complexity. Use techniques like MoSCoW (Must have, Should have, Could have, Won't have) to prioritize.

12) Conduct validation sessions with stakeholders to ensure that the documented requirements accurately reflect their expectations and needs. Address any discrepancies or misunderstandings promptly.

13) Continuously iterate and refine requirements based on feedback from stakeholders and evolving project understanding. Maintain open communication channels to accommodate changes and updates as necessary.

14) Seek consensus among stakeholders on the finalized set of requirements before moving forward with design and implementation phases. Ensure that all parties are aligned and committed to the project goals.

15) Establish traceability between requirements and project deliverables to ensure that each requirement is addressed and validated throughout the project lifecycle. Use traceability matrices or tools to manage this process effectively.

16) Clearly communicate the finalized requirements to all project team members, including developers, designers, testers, and project managers. Ensure everyone has a shared understanding of what needs to be delivered.

17) Implement a change management process to handle any requested changes or updates to requirements throughout the project lifecycle. Evaluate impacts on schedule, budget, and scope before approving changes.

18) Obtain formal review and approval of requirements documentation from stakeholders and project sponsors. Document sign-offs to confirm agreement on the scope and expectations of the digital twin project.

Creating a stakeholder analysis matrix is a structured process that helps identify and prioritize stakeholders based on their influence, interest, and involvement in a project. Here are the steps to create a stakeholder analysis matrix.

1) List all potential stakeholders who may be affected by or have an impact on the digital twin project. This includes individuals, groups, departments, or organizations both within and outside the organization.

2) Determine Stakeholder Criteria

a) Assess how much influence each stakeholder has over the project. Influence can be determined by their authority, position, or ability to allocate resources.

b) Evaluate how interested each stakeholder is in the project's outcome. Stakeholders with high interest may be directly affected by the project or have a vested interest in its success.

c) Consider the level of involvement each stakeholder will have throughout the project lifecycle. This includes their participation in decision-making, contribution of expertise, or execution of project tasks.

3) Gather information through interviews, discussions, documentation review, and other forms of communication to understand each stakeholder's perspective, concerns, and expectations regarding the digital twin project.

4) Create a table with stakeholders as rows and the criteria (influence, interest, involvement) as columns.

5) Assess and score each stakeholder against each criterion on a scale (e.g., high, medium, low or numerical values like 1 to 5).

6) Identify stakeholders who fall into the high influence, high interest, or high involvement categories. These stakeholders are typically the most critical and should be closely engaged throughout the project.

7) Based on the analysis, develop tailored strategies for engaging with each stakeholder group. This may include regular communication, consultation on decisions, involvement in testing or validation, etc.

8) Regularly review and update the stakeholder analysis matrix throughout the project lifecycle. Stakeholder priorities and relationships may evolve, requiring adjustments to engagement strategies.

Scheduling and conducting one-on-one interviews with key stakeholders is essential for gaining insights into their roles, responsibilities, and perspectives regarding the digital twin project. Here's a structured process to effectively schedule and conduct these interviews.

1) Determine which stakeholders are crucial to interview based on their roles, expertise, and involvement in the digital twin project. These may include project sponsors, business managers, domain experts, IT personnel, end-users, etc.

2) Create an interview guide that outlines the topics and questions you want to cover during the interviews. The guide should be structured to ensure consistency across interviews while allowing flexibility to explore specific areas based on the stakeholder's responses.

3) Contact stakeholders to schedule the interviews. Provide clear information about the purpose of the interview, its duration, and any preparations they need to make (e.g., reviewing project documentation, preparing examples).

4) Decide whether interviews will be conducted in person, over the phone, or via video conferencing. Choose a location that is conducive to a focused discussion and minimizes distractions.

5) Ensure that all necessary equipment and tools (e.g., recording devices, note-taking materials) are ready before the interviews. Test technology if conducting remote interviews to avoid technical issues.

6) Start each interview by introducing yourself and explaining the purpose of the interview.

7) Provide context about the digital twin project, its objectives, and how their input is valuable to its success.

8) Ask stakeholders to describe their roles and responsibilities within the organization.

9) Probe deeper to understand how their roles intersect with the digital twin project and what their specific contributions or expectations are.

10) Explore stakeholders' perspectives on the digital twin project. Ask about their understanding of its goals, potential benefits, and any concerns or challenges they foresee.

11) Encourage stakeholders to share examples or anecdotes that illustrate their perspectives.

12) Use the interview guide to systematically ask about stakeholders' requirements, expectations, and desired outcomes from the digital twin project.

13) Clarify any ambiguities and seek specifics regarding functionality, usability, performance, and other relevant aspects.

14) Allow time for stakeholders to ask questions or raise issues that may not have been covered earlier in the interview.

15) Listen actively and take notes to capture important insights and details.

16) Summarize key points discussed during the interview to ensure mutual understanding.

17) Thank stakeholders for their time and participation.

18) Inform stakeholders about any follow-up actions, such as sharing a summary of findings or incorporating their feedback into project plans.

19) Review your notes or recordings from the interviews to identify common themes, priorities, and areas of consensus or divergence among stakeholders.

20) Document detailed interview notes, including direct quotes and paraphrased insights, while maintaining confidentiality and respecting stakeholder privacy.

21) Synthesize stakeholder requirements and expectations into a comprehensive document or report. Clearly outline priorities, dependencies, and any conflicting perspectives that need further resolution.

22) Share the synthesized findings with stakeholders for validation and feedback. Ensure that stakeholders have an opportunity to review and clarify their inputs.

23) Incorporate validated stakeholder insights into project plans, requirements documentation, and ongoing communication strategies to ensure alignment and stakeholder buy-in.

Designing and distributing surveys to stakeholders is an effective way to gather structured feedback on their requirements and preferences for a digital twin project. Here is the process to design and distribute surveys.

1) Clearly define the goals and objectives of the survey. Determine what specific information you want to gather from stakeholders regarding their requirements and preferences for the digital twin project.

2) Identify the target audience for the survey. This may include business managers, domain experts, IT personnel, end-users, and other relevant stakeholders who have insights into the project.

3) Choose a survey tool that suits your needs. Popular options include Google Forms, SurveyMonkey, Typeform, or internal survey tools provided by organizational software.

4) Create a set of clear, concise, and relevant questions that align with the objectives of the survey. Ensure that questions are structured to gather actionable insights. Types of questions can include:

a) Multiple-choice: Provide predefined options for stakeholders to choose from.

b) Rating scales: Ask stakeholders to rate their preferences or satisfaction on a scale (e.g., 1 to 5).

c) Open-ended: Allow stakeholders to provide detailed comments or suggestions in their own words.

5) Arrange questions in a logical sequence to maintain flow and coherence. Start with general questions before moving to more specific ones about requirements and preferences.

6) Optionally, include demographic questions (e.g., department, role, years of experience) to segment responses and identify patterns based on different stakeholder groups.

7) Review the survey for clarity, relevance, and coherence. Test the survey with a small group of stakeholders to identify any ambiguities or issues before finalizing it.

8) Determine how you will distribute the survey to stakeholders. Options include email distribution lists, internal communication platforms (e.g., Slack, Microsoft Teams), or embedding the survey link in project documentation.

9) Clearly communicate the purpose of the survey, its importance, and how stakeholder input will be used to inform decisions related to the digital twin project. Encourage participation by emphasizing the value of their feedback.

10) Specify a deadline for survey responses to ensure timely completion. Allow sufficient time for stakeholders to respond while maintaining project timelines.

11) Send reminder emails or notifications to stakeholders as the deadline approaches to encourage those who have not yet responded to complete the survey.

12) Monitor survey responses as they are submitted to ensure completeness and track response rates. Most survey tools provide real-time analytics and reporting capabilities.

13) Analyze survey data to identify trends, patterns, and common themes in stakeholder requirements and preferences. Use charts, graphs, and summary statistics to visualize and interpret the data effectively.

14) Segment responses based on demographic information or stakeholder roles to understand variations in perspectives across different groups.

15) Summarize key findings from the survey in a clear and concise format. Highlight insights related to stakeholder requirements, preferences, priorities, and any significant feedback received.

16) Prepare a comprehensive report or presentation that outlines survey findings, including actionable recommendations based on stakeholder feedback. Include quotes or verbatim responses to provide context and depth.

17) Share the survey findings with stakeholders, project sponsors, and relevant teams involved in the digital twin project. Ensure transparency and facilitate discussions around how the insights will influence project planning and decision-making.

Identifying the sources of data that will feed into the digital twin is a critical step in its development. Here is the process to identify these data sources.

1) Clarify the objectives and use cases of the digital twin. Determine what specific aspects of the physical system or process the digital twin will replicate, monitor, analyze, or simulate.

2) Identify the key variables, parameters, or metrics that the digital twin needs to monitor, simulate, or predict. These may include physical characteristics, performance indicators, operational parameters, etc.

3) Engage with stakeholders, including domain experts, engineers, operations managers, and IT personnel, through workshops or interviews. Discuss the types of data needed to support the digital twin's functionalities and objectives.

4) Analyze existing documentation such as system specifications, process diagrams, technical manuals, and data flow diagrams. Identify potential data sources that are already documented or integrated into current systems.

5) Map out the flow of data within the physical system or process that the digital twin will replicate. Identify points where data is generated, collected, stored, and used for decision-making or operational control.

6) Identify sensors and IoT devices that collect real-time data from the physical system. This includes sensors for temperature, pressure, flow rates, vibration, energy consumption, etc.

7) Consider data generated by control systems, SCADA (Supervisory Control and Data Acquisition) systems, PLCs (Programmable Logic Controllers), and other automation systems that manage and control operations.

8) Access historical data repositories such as databases, data warehouses, or data lakes that store past operational data, maintenance records, performance logs, and incident reports.

9) Explore external data sources that could enrich the digital twin's understanding or predictions, such as weather data, market trends, supplier data, or regulatory information.

10) Identify any manual data inputs or data entry points where operators or technicians input data that could be relevant to the digital twin.

11) Evaluate the quality, reliability, and accuracy of the identified data sources. Consider factors such as data completeness, consistency, timeliness, and potential biases.

12) Assess the accessibility of data from identified sources. Determine whether data can be accessed in real-time or near real-time, or if there are delays or limitations in data availability.

13) Prioritize data sources based on their relevance to the digital twin's objectives, the quality of data they provide, and their accessibility. Consider the impact of each data source on the accuracy and effectiveness of the digital twin.

14) Identify dependencies and interactions between different data sources. Ensure that data integration and synchronization across sources are feasible and well-managed.

15) Create a comprehensive inventory or catalog of identified data sources. Document details such as source location, type of data (e.g., sensor data, historical data), frequency of updates, and data ownership.

16) Develop a plan for data mapping and integration into the digital twin platform. Define protocols, APIs, or integration methods required to connect and synchronize data from various sources.

17) Validate the identified data sources and integration plan with stakeholders, including domain experts and IT personnel. Ensure alignment with project requirements and verify feasibility from technical and operational perspectives.

18) Continuously review and refine the list of data sources based on feedback, changes in project scope, or evolving requirements.

Assessing the quality, reliability, and completeness of data from various sources is crucial to ensure that the data used for a digital twin project is accurate and reliable. Here is the process to assess these aspects.

1) Determine the specific dimensions of data quality that are relevant to your digital twin project. Common dimensions include:

a) Accuracy: How closely data values reflect the true values or states of the measured entity.

b) Completeness: The extent to which all required data points are present within the dataset.

c) Consistency: The absence of contradictions between different data sources or over time.

d) Timeliness: The availability of data within an acceptable timeframe for use.

e) Relevance: The extent to which the data is pertinent and applicable to the digital twin's objectives.

f) Validity: The conformity of the data to defined formats, rules, and constraints.

g) Reliability: The dependability of data sources and the consistency of data collection methods.

2) Use data profiling tools or scripts to analyze data characteristics such as data types, value ranges, and distribution. This helps identify anomalies and potential issues.

3) Take samples of data from various sources and conduct statistical analysis to assess data variability, outliers, and patterns. This provides insights into data consistency and reliability.

4) Evaluate the completeness of datasets by comparing the expected data points against the actual available data. Identify missing values or gaps in data collection.

5) Validate data accuracy through cross-validation with independent sources or by comparing data against known benchmarks or ground truth measurements.

6) Check for inconsistencies by comparing data across different sources or over time. Identify discrepancies that may indicate data integration issues or errors.

7) Assess the timeliness of data updates and availability. Determine whether data is available in a timely manner to support real-time or near-real-time operations of the digital twin.

8) Evaluate the quality of sensor data by considering factors such as sensor calibration, measurement accuracy, and environmental conditions affecting sensor performance.

9) Review historical data for data entry errors, gaps in recording, or inconsistencies in data format. Ensure historical data is relevant and reliable for analysis.

10) Validate external data sources by assessing the credibility and reputation of data providers. Check for data quality assurances or certifications.

11) Document findings from data quality assessments, including identified issues, data quality scores, and recommendations for improvement.

12) Prepare a comprehensive report summarizing data quality assessments for each data source. Include actionable insights and prioritized recommendations for enhancing data quality.

13) Establish data quality controls and monitoring mechanisms to continuously assess and improve data quality over time.

14) Maintain a feedback loop with data providers and stakeholders to address identified data quality issues promptly and collaboratively.

15) Continuously refine data quality assessment processes based on feedback, new data sources, or changes in project requirements.

Determining the level of fidelity required for a digital twin model involves assessing the balance between accuracy, complexity, and computational resources needed to achieve the desired outcomes. Here is the process to guide this determination.

1) Clarify the specific objectives of the digital twin model. Determine what insights or predictions the model needs to provide to support decision-making or improve operations.

2) Identify and prioritize use cases where the digital twin will be applied. Consider whether the focus is on predictive maintenance, optimization of operations, performance monitoring, simulation of scenarios, etc.

3) Determine the required level of accuracy for the digital twin model outputs. Consider how closely the model predictions need to match real-world measurements or observations.

4) Assess the level of detail and precision needed in modeling various aspects of the physical system or process. This includes spatial and temporal resolutions, as well as the granularity of data inputs.

5) Evaluate the availability and accessibility of high-quality data sources needed to develop and validate the digital twin model. Consider the reliability, timeliness, and completeness of data.

6) Assess the variability and uncertainty in data inputs. Determine whether data-driven approaches alone can adequately capture the system's behavior or if additional modeling fidelity is necessary.

7) Physics-Based Modeling:

a) Assess the feasibility and appropriateness of physics-based models that simulate the underlying physical principles governing the system's behavior.

b) Determine if detailed knowledge of system dynamics and equations is available and if computational resources are sufficient to support complex simulations.

8) Data-Driven Approaches:

a) Evaluate the potential of data-driven models that leverage historical data and machine learning techniques to capture patterns and correlations within the data.

b) Consider the scalability and adaptability of data-driven models to evolving system conditions and data inputs.

9) Hybrid Models:

a) Explore hybrid modeling approaches that combine physics-based and data-driven methods to leverage the strengths of both approaches.

b) Determine if hybrid models can provide a balance between accuracy, computational efficiency, and data requirements.

10) Evaluate the computational resources available for developing, training, and running the digital twin model. Consider factors such as processing speed, memory requirements, and scalability.

11) Determine if real-time or near-real-time performance is necessary for the digital twin model. Assess whether the chosen modeling approach can meet these requirements.

12) Consult with domain experts, stakeholders, and end-users to validate assumptions and requirements regarding the level of fidelity needed.

13) Iterate on the determination of fidelity level based on feedback, evolving project requirements, and technological advancements.

14) Based on the assessments and considerations above, decide on the appropriate level of fidelity for the digital twin model.

15) Document the rationale behind the chosen modeling approach and fidelity level. Include considerations such as project goals, data availability, computational constraints, and stakeholder input.

16) Continuously monitor the performance of the digital twin model in practice. Assess whether the chosen fidelity level meets expectations and delivers actionable insights.

17) Be prepared to adapt the digital twin model’s fidelity level based on feedback, changes in operational conditions, and advancements in modeling techniques.