



Data architecture design method for exchange-collaboration platforms via data coordination

Lifeng Wang^{1,2,*}, Liguang Yang^{1,2}

¹PowerChina Huadong Engineering Corporation Limited, Hangzhou 311122, China;

²Zhejiang Huadong Engineering Digital Technology Co., Ltd., Hangzhou 311122, China;

*wang_lf3@hdec.com

Abstract. Engineering projects are often plagued by data silos: stakeholders' systems operate in isolation, impeding interaction, coordination, and response. To break these barriers, we propose a data-coordination-driven architecture for an exchange-collaboration platform that integrates BIM and digital-engineering technologies. A multi-tier processing and service layer ingests, cleanses, transforms, and fuses multi-source heterogeneous data and exposes a unified data-service interface. The platform improves data interoperability and business collaboration across the entire project life-cycle and provides an efficient, reliable data foundation for digital construction management.

Keywords: data coordination; data architecture; data hub; BIM;

1 Introduction

Traditional information-management paradigms in construction are characterised by pervasive data silos: disparate stakeholders deploy heterogeneous systems that are locked to specific project phases, resulting in poor cross-system data exchange, inefficient process coordination and sluggish response times. These frictions degrade both construction quality and collaborative performance.

Although incumbent foreign platforms such as Autodesk BIM 360 and Bentley^{[1][2]} ProjectWise provide mature file-based workflows, their closed architectures and limited extensibility prevent domestic owners, designers and contractors from accumulating self-controlled, granular data assets. This constraint not only introduces information-security risks but also stifles indigenous innovation in engineering-software ecosystems.

Recent domestic prototypes have attempted to replace foreign solutions with self-built collaboration portals; however, most remain at the file-transfer level and fail to deliver structured, real-time data-level collaboration. Consequently, data-level coordination—i.e., the continuous, loss-free and semantics-preserving flow of heterogeneous data—has become the critical path toward genuine interoperability.

Motivated by this gap, we propose a data-coordination-oriented architecture for an exchange-collaboration platform. The design follows The Open Group Architecture

Framework (TOGAF)^{[3][4]} and explicitly integrates building-information-modelling (BIM) and digital-engineering technologies. A multi-tier data-processing and service fabric ingests, cleanses, transforms and fuses multi-source heterogeneous data, and exposes a unified, standards-based service interface. The architecture aims to dismantle information silos, enhance cross-organisational and whole-life-cycle collaboration, and provide a theoretically sound and technically reproducible foundation for autonomous, controllable BIM platforms in China.

To more clearly demonstrate the differentiated advantages of our platform compared to incumbent foreign platforms (e.g., Autodesk BIM 360), a comparative analysis is conducted from dimensions such as core architecture and data granularity. The specific comparison is presented in Table 1.

Table 1. Comparative Analysis of Platform Advantages.

Comparison Aspect	Proposed Platform	Autodesk BIM 360
Core Architecture	Open, Extensible	Closed, Restricted
Data Interaction Granularity	Component-level, Real-time Data Services	Primarily File-based Version Management
Semantic Interoperability	Resolves semantic conflicts via BES intermediate model	Relies on native software semantics, significant semantic loss across software
Data Autonomy & Control	High, supports on-premises deployment	Low, dependent on cloud vendor
Customization Cost	Low, supports hot-pluggable plugins and microservices	High, limited by platform-provided APIs

2 Project Overview

The research is undertaken under the National Major R&D Programme “Systematic Development and Application of Independently Controllable BIM Platforms for Complex Engineering”. Guided by three principles—(i) demand-driven, collaborative development and application; (ii) focus on common, core technologies; and (iii) pooled resources with clear division of labour—the project delivers a BIM-data exchange and collaboration platform.

The platform’s primary objective is to guarantee interoperability of multi-source heterogeneous BIM data by establishing a universal, reliable and secure data-exchange environment. A canonical, structured data-exchange paradigm has been devised that defines normative requirements for BIM data during design, transfer and storage phases, thereby significantly improving the accuracy and efficiency of cross-system and cross-platform data sharing.

Figure 1 illustrates the platform’s layered architecture^[5]. The domain layer encapsulates core business capabilities and data models and constitutes the architectural kernel. The present paper concentrates on the internal data architecture of this layer,

including domain-object partitioning, data-aggregation design, data-lifecycle governance and persistence strategy. This data architecture is the cornerstone for achieving high business cohesion, low coupling and data consistency in complex collaborative scenarios.

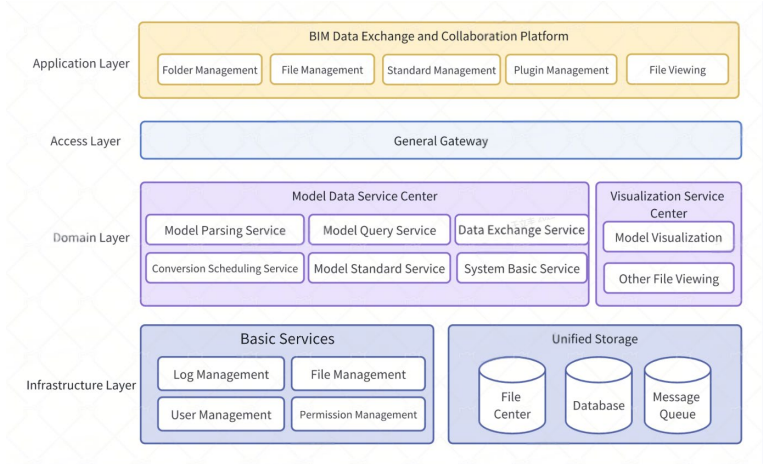


Fig. 1. Overall Architecture of the Collaboration Platform.

3 Architecture Design

The BIM-data exchange and collaboration platform adopts TOGAF as its planning methodology and is decomposed into four conceptual tiers: data sources, data conversion, data services and data storage (Figure 2). The following sections elaborate on each tier.

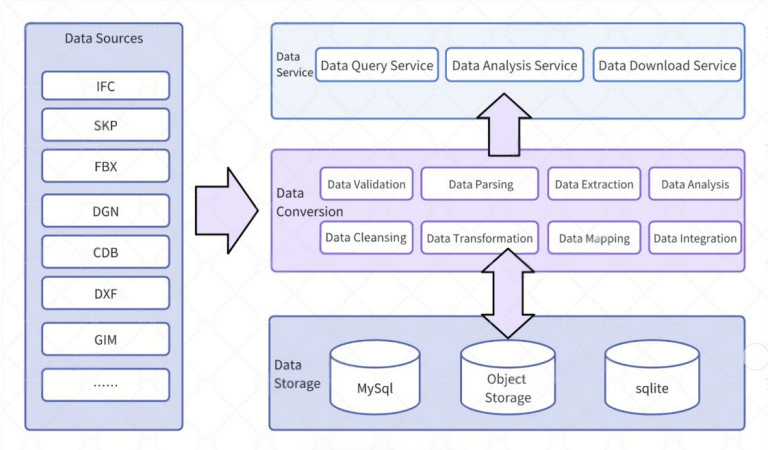


Fig. 2. Data Architecture Design.

3.1 Data Sources

Model data span the entire life cycle—survey, design, construction, operation and maintenance. The platform ingests mainstream BIM formats (IFC, SKP, FBX, DGN, DXF, CDB, GIM) covering architecture, structure, MEP, municipal, railway and hydraulic engineering. Both structured (properties, sensors) and unstructured (3-D meshes, drawings, documents) data are admitted, yielding a multidimensional, multi-scale and multi-disciplinary data corpus. As shown in Figure 3, this approach integrates various data types to support comprehensive analysis and application.

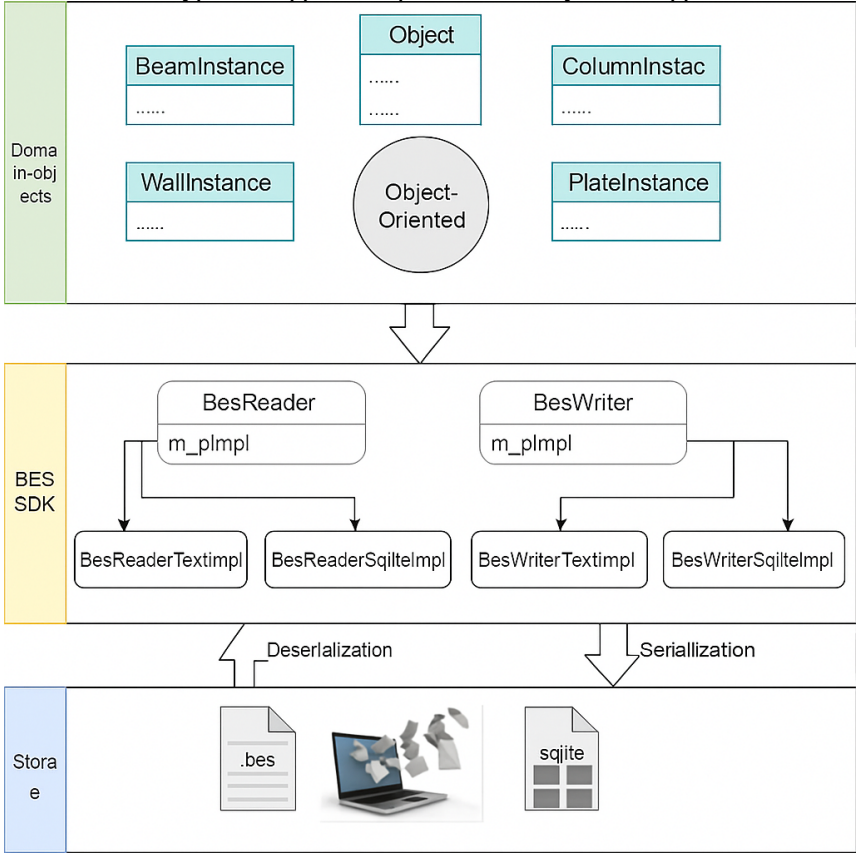


Fig. 3. Overall Functional Architecture.

3.2 Data Conversion

3.2.1 Data Standards. Heterogeneous differences in syntax, semantics, coordinate reference systems (CRS) and modelling conventions are resolved through the BIM Exchange Standard (BES) framework^[6]. An intermediate semantic model maps external schemas to a platform-internal canonical schema: (i)Syntax layer: BES text and database formats with compressed payload storage; (ii)Semantic layer: BES entity,

property-set and relation definitions; (iii)Geometry layer: BES primitives (BesCoordinates3d, BesVector3d, BesShape, BesAppearanceMaterial, BesCircleSection) with unified CRS.

The core of the intermediate semantic model is the establishment of a platform-internal Canonical Semantic Dictionary. This dictionary assigns a unique, unambiguous internal identifier and property set to each core engineering concept (e.g., "Structural Column"). Upon importing BIM data from diverse sources, configurable mapping rules are applied to translate heterogeneous terminologies from external models (e.g., "Column" from Discipline A and "StructuralColumn" from Discipline B) into this unified concept. This process resolves semantic conflicts arising from differing naming conventions and disciplinary jargon, ensuring information remains consistent and interpretable during cross-system flow. For instance, during clash detection, all structural columns are correctly identified and analyzed regardless of their original naming in the source models.

Relational constraints and mandatory-attribute validation guarantee integrity and identifiability during cross-system flow. Pre-processing pipelines perform deduplication, missing-value imputation, error correction, semantic alignment and CRS unification. A data-quality metrics suite automatically verifies geometric accuracy, attribute completeness and logical consistency.

During the data conversion process, the platform employs technical methods such as data cleansing, format conversion, semantic alignment, and coordinate unification to preprocess the incoming data. This specifically includes the removal of redundant data, imputation of missing values, correction of erroneous data, and recognition and parsing of multiple data formats. Semantic alignment involves establishing a common semantic dictionary and mapping rules to uniformly map terms and attributes from different sources to the platform's standard definitions, ensuring that information is not distorted during cross-system transmission. Coordinate unification transforms all spatial data into the coordinate system specified by the platform through coordinate reference system transformation and geometric calibration, ensuring accurate positioning and seamless integration of spatial data during integration and display.

Concurrently, in conjunction with industry standards and project requirements, a data quality assessment system is established to automatically verify the geometric accuracy, attribute completeness, and logical consistency of models, ensuring the converted data possesses high usability and credibility. The engineering project instances in the files can contain corresponding sites, buildings, floors, and component elements based on the spatial composition of the model and its association with model elements. The spatial composition of the model can adopt a spatially decomposed manner. The following uses a building model as an example, as shown in Figure 4.

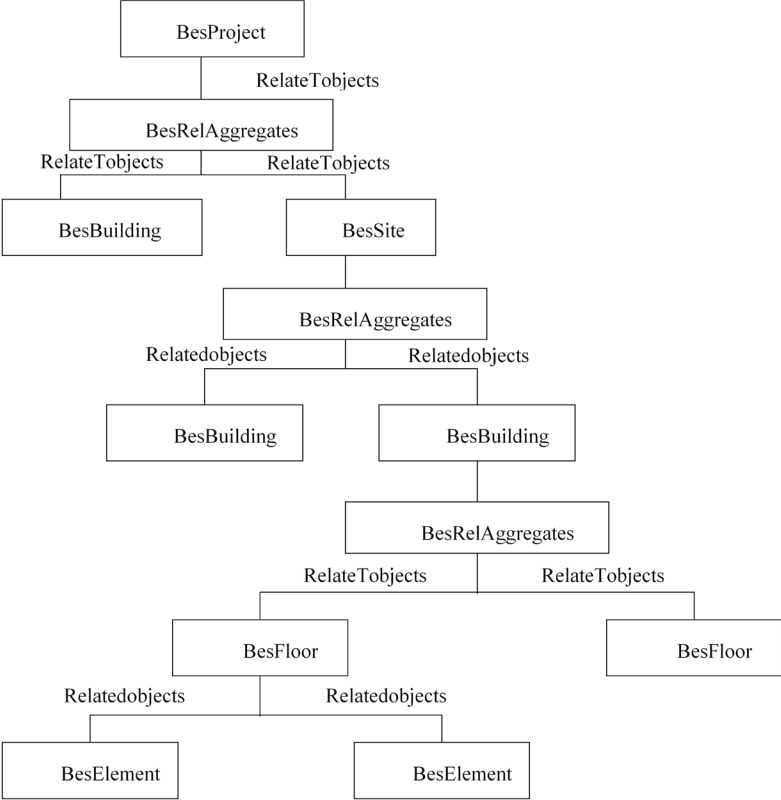


Fig. 4. Spatial Composition of the Building Information Model.

3.2.2 Conversion-Plug-in Framework. Each external format is handled by a light-weight plug-in that performs read-map-write operations: parse the native file into memory-resident intermediate objects, then emit BES-conformant structures. Plug-ins are mutually independent, hot-swappable and upgradable. Table 2 summarises the IFC→BES plug-in workflow; remaining plug-ins (DGN, FBX, DXF, GIM) follow an identical pattern.

Table 2. Internal Workflow and Interface Design of the Plug-in.

Function Signature	Description
const char* getExt()	Returns the file extension handled by the plugin, such as "ifc".
ICConverter* create()	Creates an instance of the converter, which is managed and released by the platform.
bool ICConverter::convert(const char* in, const char* out, ICallback* cb)	Executes the conversion; during the process, it sends a unified memory object to the platform via cb->send(obj).
void ICConverter::setOption(const char* key, const char* val)	Dynamically receives platform parameters, such as -BesLengthUnit=METRE.

3.3 Data Services

The "real-time data-level collaboration" achieved by this platform focuses on providing fine-grained, instantly accessible and operable data service interfaces, rather than requiring all users to edit the same model entity simultaneously. The architecture supports efficient near-real-time collaboration through the following mechanisms: 1) High-performance Data Service Layer: Offers data query and update interfaces with millisecond-level response, ensuring swift feedback for user operations. 2) Optimistic Locking and Conflict Detection: When multiple users attempt to modify the same component, an optimistic locking mechanism is employed. Version conflicts are detected upon data submission. If a conflict occurs, the system alerts the latter submitting user and provides options for diff comparison and manual merge. 3) Event-driven Notification: A real-time notification service, based on technologies like WebSocket, actively pushes update events to all online clients when model data changes, enabling their interfaces to refresh in near-real-time.

All interfaces are cross-platform and support HTTPS/TLS encryption and OAuth 2.0 authorisation, as shown in Figure 5.

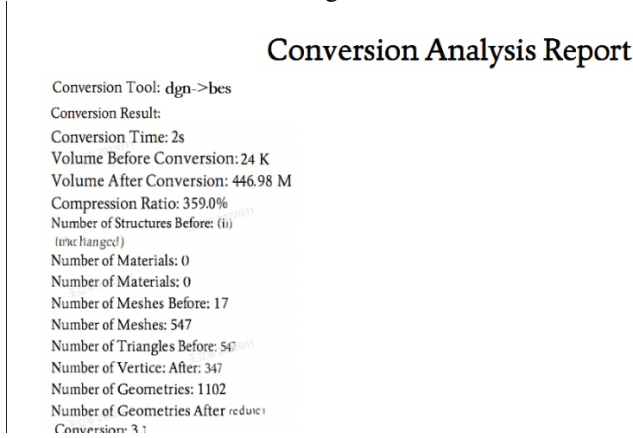


Fig. 5. Preview of the Model Conversion Analysis Report.

3.4 Data Storage

A logical–physical two-tier strategy that supports high-frequency querying, version tracing and cross-platform migration is organised as follows:

(i) Logical model—116 entities are grouped into three strata:

- Definition classes (Entity*, Domain*) store metadata for geometry, attributes and classification;
- Instance classes hold concrete objects, relations and property values whose names are strictly aligned with the BES for cross-system comparability;
- File classes (File*) keep source-file metadata (version, author, schema, encoding) and enable life-cycle traceability.

Every table starts with `_refid`, mirroring the “#Number” field in the BES text segment to allow bidirectional text–database positioning. Foreign keys `ParentId`, `ClassId` and `RelatingObject` encode inheritance, aggregation and attribution semantics; cascade updates and integrity checks are enforced by the engine. Table 3 provides an overview of the structured data fields used in the BIM data exchange platform, detailing each field's properties such as name, data type, unit, value range, and its significance in engineering contexts. This standardization ensures data consistency and interoperability across the platform.

Table 3. Example of Data Entity Attributes.

Field Name	Data Type	Unit	Value Range	Engineering Meaning
id	long			Primary Key of the Database
beam_id	long			ID of the Department it Belongs to
slot_height	double	m	>0	Height of the Slot
slot_width	double	m	>0	Width of the Slot
...	...	...	...	...
rotate_angle	double	rad	$(-\pi/2, \pi/2)$	The angle of the diagonally placed transverse diaphragm is defined as positive when measured counterclockwise from the smaller mileage to the larger mileage along the top surface of the beam.

(ii) Physical model—SQLite serves as the default engine: server-less, single-file, transactional and foreign-key-aware. A one-million-component model returns sub-second queries on an embedded workstation. When concurrency scales to multi-user on-line collaboration, zero-logic migration to MySQL or PostgreSQL is achieved by updating a data-type mapping table.

(iii) Type system and indexing—The logical layer adopts the least-common subset {NULL, INTEGER, REAL, TEXT, BLOB}. Multi-valued or nested attributes are stored as “optional flag + JSON blob”^[7] to prevent wide-table bloat while preserving extensibility. B-tree indexes cover `_refid`, `ClassId`, `ParentId` and high-frequency business filters; full-text search is enabled through SQLite FTS5.

(iv) Extension and archiving—New domain entities are hot-plugged by inserting rows into `DomainClass` and `DomainAttribute` without touching the core schema. Historical data are partitioned by project $\times$ time; cold partitions are automatically migrated to compressed tables, cutting disk usage by >50 %. Large BLOBs (geometry, textures) are off-loaded to object storage via external linkage, achieving compute–storage decoupling and cloud-scale elastic growth.

The platform implements a fine-grained access control strategy based on project roles (RBAC). Each data entity is associated with metadata that explicitly records its owner (creator or owning organization), access permissions (read, write, administer), and data state (e.g., Draft, Under Review, Published, Archived). A data lifecycle state machine governs the flow of data. For instance, a component created by a designer is

in the "Draft" state, visible only to the project team; upon submission for review, it changes to "Under Review"; after approval, it is "Published" as a baseline version and cannot be arbitrarily modified; upon project completion, the entire dataset can be "Archived" to save storage space while preserving an audit trail. All these governance details are stored as metadata within the Definition classes and File classes, ensuring full life-cycle traceability, security, and compliance.

4 Project Validation

Systematic tests were conducted on five mainstream formats (IFC, DXF, DGN, SKP, FBX) covering architectural, structural and MEP samples. The private-cloud testbed comprised 40-core CPU, 256 GB RAM nodes isolated within an enterprise intranet. Metrics: upload time, conversion time, conversion-success rate and data-integrity rate. Each sample was repeated three times and averaged.

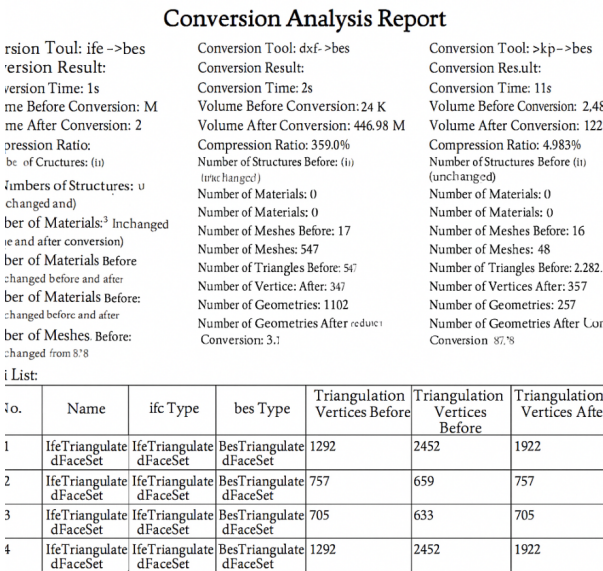


Fig. 6. Model Analysis Report.

Table 4. Performance Comparison of Conversion Plugins.

Format Type	Average Upload Time (s)	Average Conversion Time (s)	Average File Size (MB)	Average Number of Components	Conversion Success Rate (%)	Data Integrity (%)
IFC	1	1	2	13	98.7	99.2
DXF	4	2	8.4	/	97.5	98.1
DGN	3	16	6.4	486	99.3	99.5
SKP	22	60	44.7	12827	98.9	98.7
FBX	37	112	75.4	4924	99.5	99.0

Results (Table 4) show conversion-success $\geq 97\%$ and data-integrity $\geq 98\%$ for every format, corroborating plug-in robustness. A real-time analytics report (Figure 6) lists component counts, geometric detail and conversion logs, allowing users to verify quality instantly. The architecture therefore meets the efficiency and fidelity demands of large-scale engineering collaboration.

5 Conclusion

We have presented a four-tier, TOGAF-aligned data architecture that systematically coordinates multi-source, cross-phase and cross-disciplinary BIM data. By mapping heterogeneous formats into a canonical, traceable BES schema, the platform delivers: 1 000 components converted in <30 s with complete geometry and attribute retention; 20-million-polygon models rendered at ≥ 30 fps on desktop clients; logical–physical decoupling supporting SQLite, MySQL and PostgreSQL with project- and time-based partitioning; elastic cloud scaling and micro-service deployment.

National-level project deployment confirms that the architecture furnishes a reusable, extensible and autonomous data backbone for domestic BIM platforms and accelerates the industry’s digital transformation.

References

1. Song P, Li Y, Yao F, et al. Digital collaborative design of the DC yard for the ± 1100 kV Guquan converter station based on ProjectWise[J]. *Electric Power Survey & Design*, 2018(5): 66-70.
2. Meng M, Du Z, Yang Y. Research and application of BIM collaborative design for ecological river channels based on data shortcuts[J]. *Water Conservancy Construction and Management*, 2023, 43(8): 1-9, 29.
3. Ma X, Su C, Zhang X. Practice of enterprise architecture method based on TOGAF[J]. *Digital Communication World*, 2017, No. 148(04): 442-447.
4. Liu S, Shi J, Dong F. Research on the maturity evaluation framework of enterprise digital transformation based on the TOGAF standard[J]. *The Journal of New Industrialization*, 2023, 13(7): 36-45.
5. Guo S, Zeng Z, Li H. *Enterprise Architecture and Digital-Transformation Planning & Design Tutorial*[M]. Beijing: Tsinghua University Press, 2022.
6. Cheng K, Dai C, Liang B, et al. Data interaction and application of building information models based on the Industry Foundation Classes standard[J]. *Science Technology & Engineering*, 2024, 24(9): 3797-3803.
7. He N. Research on database backup and recovery strategies in the big-data environment[J]. *Information & Computer (Theoretical Edition)*, 2023, 35(8): 179-181.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

