Towards a Digital Construction Platform at the National Research Council of Canada:

Blueprint for Digitalization of National Model Construction Codes, National Master Specification and the Canadian Construction Materials Center

December 2025

Final Report

—

Presented by

buildingSMART Canada

https://buildingsmartcanada.ca

Project Team

Project Lead:

- David Minicola, MBA, NRC

Research Lead:

- Erik Poirier, PhD, buildingSMART Canada

Research Team:

- Mehdi Fakour, M.Sc, buildingSMART Canada
- Esmaeil Naderi, M.Sc, buildingSMART Canada

Research Advisors:

- Farzad Jalaei, PhD, P.Eng., NRC
- Arash Hosseini, PhD, NRC

Contributors / Advisors:

- Jason Urquhart, BScF.Eng, P.Eng, NRC
- Steve Londry, NRC
- Keerthi Prashanth, NRC

The views, opinions, and conclusions expressed in this report are solely those of the authors and do not necessarily reflect the views or positions of any organizations, institutions, or entities that the authors may be associated with. The content is provided for informational purposes only and should not be considered as professional or legal advice.

Artificial intelligence was used in the development of parts of this report to support analysis of content as well as drafting and editorial work.

Contents

Table of Figures

List of Tables

Abbreviations

Abbreviation	Full title
ACCORD	Automated Compliance Checks for Construction, Renovation, or Demolition Works
AHJ	authority having jurisdiction
API	Application Programming Interface
AR	augmented reality
AWS	Amazon Web Services
BCA	Building and Construction Authority (Singapore)
BCF	BIM Collaboration Format
BIM	Building Information Modelling
BPMN	Business Process Model and Notation
bSC	buildingSMART Canada
bSDD	buildingSMART Data Dictionary
CBHCC	Canadian Board for Harmonized Construction Codes
CCBFC	Canadian Commission on Building and Fire Codes
CCMC	Canadian Construction Materials Centre
CCMS	Component Content Management System
CDBB	Centre for Digital Built Britain
CDE	common data environment
CDF	common data framework
CE	circular economy
CHEK	Change Digital Building Permit
CityGML	An open standardized data model and exchange format to store digital 3D models of cities and landscapes
CSC	Construction Specifications Canada
CSDP	Construction Sector Digitalization and Productivity Challenge program
CSI	Construction Specifications Institute
CSM	Contract Security Manual
DBP	digital building permit
DCP	Digital Construction Platform
D-COM	digital compliance
DITA	Darwin Information Typing Architecture
DTD	Document Type Definition
EPD	Environmental Product Declaration
ÉTS	École de Technologie Supérieure
EUnet4DBP	European network for Digital Building Permit
GDPR	General Data Protection Regulation
GHG	greenhouse gas
GIS	geographic information system

Abbreviation	Full title
GraphQL	A data query and manipulation language for APIs, that allows a client to specify what data it needs
ICC	International Code Council
IDEF	Integrated DEFinition Methods
IDM	Information Delivery Manual
IDS	Information Delivery Specification
IEC	International Electrotechnical Commission
IoT	Internet-of-Things
IFC	Industry Foundation Classes
IR	Information requirements
ISO	International Organization for Standardization
LCA	Life Cycle Assessment
LCCA	Life Cycle Cost Analysis
LCBE	Low Carbon Built Environment Challenge program
LoA	Level of Accuracy
LoC	Level of Completeness
LoC	Level of Coordination
LoD	Level of Development
LoD	Level of Detail
LoI	Level of Information
LOIN	Level of Information Need
LoX	Levels of X
MDA	Model Driven Architecture
NIST	National Institute of Science and Technology
NMCC	National Model Construction Codes
NMS	National Master Specifications
NRC	National Research Council of Canada
OGC	Open Geospatial Consortium
OMG	Object Management Group
openBIM	A collaborative process that is inclusive of all participants, promoting interoperability to benefit projects and assets throughout their lifecycle.
openCDE	a family of standards for providing connectivity and open communication between Common Data Environment platforms and BIM tools for the building and infrastructure sectors in the construction resources industry
OWL	Web Ontology Language
PDCSS	Platform to Decarbonize the Construction Sector at Scale
PDF	Portable Document Format
PIM	project information model
RDF	Resource Description Framework
PTNB	Plan Transition Numérique dans le Bâtiment
REST	Representational State Transfer

Abbreviation	Full title
RESTful API	An interface that two computer systems use to exchange information securely over the internet.
RIM	Regulatory Information Management Framework
RTF	Rich Text Format
SDO	standards development organization
SOAP	Simple Object Access Protocol
STEP	Standard for the Exchange of Product Model Data
SysML	System Modelling Language
TRL	Technology Readiness Level
UID	unique identifier
UCM	Use Case Management
UML	Unified Modelling Language
VR	virtual reality
VRML	Virtual Reality Modelling Language
WIP	work in progress
XML	Extensible Markup Language

Glossary

Term	Definition
Digital Construction Platform (DCP)	A digital ecosystem developed by the NRC to manage, integrate, and distribute construction codes, specifications, and product data in machine-interpretable formats.
Common Data Framework (CDF)	A structured semantic framework that enables consistent data organization, interoperability, and integration across NRC resources.
Digitalization	The process of transforming traditional document-based content into structured, machine-interpretable data.
Machine-interpretable	Data that can be processed and understood by computer systems without human intervention.
Interoperability	The ability of different systems and datasets to exchange and use information seamlessly.
NMCC (National Model Construction Codes)	Regulatory documents defining requirements for building safety, performance, and compliance in Canada administered by NRC.
NMS (National Master Specifications)	Standardized templates describing construction materials, systems, and installation methods administered by NRC.
CCMC (Canadian Construction Materials Centre)	A registry providing technical evaluations and compliance information for construction products administered by NRC.
Metamodel	A high-level model defining shared classes, attributes, and relationships used to structure and connect data across resources.
Data Model	A structured representation of entities, attributes, and relationships within a specific dataset or system.
Ontology	A formal representation of concepts, relationships, and rules within a domain to enable semantic understanding and reasoning.
Taxonomy	A classification system that organizes concepts into hierarchical categories.
Data Dictionary	A structured reference defining data elements, including their meaning, format, and relationships.
Metadata	Data that describes other data, such as source, author, version, and creation date.
Regulatory Clause	A unit of regulatory text from codes that defines requirements or conditions.
Requirement	A normalized statement describing what must be achieved, derived from regulatory clauses.
Constraint	A machine-executable rule derived from a requirement, used for automated validation.
Specification Section	A structured part of a specification document describing how to meet specific requirements.
Product Type	A category of construction products sharing common characteristics.
Product	A specific manufactured item with defined properties and supporting evidence.
Property	A measurable or definable characteristic of a product or system (e.g., fire resistance).
Property Value	The specific value assigned to a property for a given product or context.

Unit	A standardized measurement used to quantify a property (e.g., hours, mm).
Evidence	Documentation supporting a property value or compliance claim, such as test reports or certifications.
Test Method	A standardized procedure used to evaluate a property or verify compliance.
Reference Standard	An external standard defining requirements, methods, or criteria (e.g., ISO, ASTM).
XML (Extensible Markup Language)	A structured format used to encode documents in a machine-readable way.
JSON (JavaScript Object Notation)	A lightweight data format used for structured data exchange, commonly used in APIs.
RDF (Resource Description Framework)	A framework for representing data as relationships between entities.
OWL (Web Ontology Language)	A language used to define and share ontologies.
API (Application Programming Interface)	A mechanism that allows systems to communicate and exchange data.
Automated Compliance Checking	The use of machine-readable rules to verify whether a design or product meets regulatory requirements.
Traceability	The ability to track relationships between requirements, specifications, and supporting evidence.
MasterFormat	An industry standard classification system organizing construction information by work results.
UniFormat	A classification system organizing building elements by function.

Executive Summary

The National Research Council of Canada (NRC) is undertaking a transformative digitalization initiative as part of its Platform to Decarbonize the Construction Sector at Scale (PDCSS). This report presents a blueprint for digitalizing three key NRC resources: the National Model Construction Codes (NMCC), the National Master Specifications (NMS), and the Canadian Construction Materials Centre (CCMC) materials database. These resources form the foundation of the NRC Digital Construction Platform (NRC DCP), which aims to enhance productivity, sustainability, and regulatory compliance across the Canadian built asset industry.

The study presented in this report followed a systematic approach to evaluate and plan for the digitalization of NRC resources which included the following steps: (1) an audit of NMCC, NMS, and CCMC content to evaluate existing formats, structures, and machine-interpretability limitations was first conducted as part of an assessment of the current state; (2) international standards and best practices were identified and discussed to specify technical requirements for machine-interpretable formats, (3) key stakeholders and their functional interactions with each resource were mapped to further develop specific use cases; (4) activity diagrams and process models documenting end-to-end workflows from content creation through publication were created for each resource; (5) a foundational metamodel, resource-specific entity-relationship models, and a consolidated model unifying all three resources through shared concepts were developed; and (6) metadata requirements and data dictionary structures to enable provenance tracking, versioning, and governance, collectively forming the foundation for the Common Data Framework (CDF) were developed.

The audit of existing NRC resources reveals varying levels of digital maturity:

- **NMCC:** Authored as XML to generate PDF and HTML outputs, the codes are well-organized but remain document-centric with poor machine-interpretability. The complex hierarchical structure (five volumes with divisions, parts, sections, subsections, articles, and clauses) requires significant transformation to enable automated compliance checking and integration with digital tools.
- **NMS:** Authored as XML to generate PDF, RTF and DOCX formats, the specifications follow industry-standard classification systems (MasterFormat/UniFormat) but lack native machine-interpretability. With approximately 860 specification sections comprising over 7,000 pages per language, the NMS is in a transitional state, with NRC actively pursuing digitalization toward BIM-compatible formats.
- **CCMC:** Authored as HTML, the CCMC material database demonstrates the most advanced digitalization progress with a JSON API providing structured access to assessment metadata. However, detailed technical evaluation content, test results, and compliance opinions remain in unstructured PDF format, limiting their utility for automated processing.

All three resources share common challenges: data complexity and volume, lack of uniformity and harmonization across resources, limited semantic linkages, and absence of machine-interpretable metadata. These limitations significantly constrain the potential for automated compliance checking, BIM integration, and modern digital construction workflows.

The report proposes a digitalization strategy centered on the CDF, which will serve as the semantic backbone of the NRC DCP. The CDF establishes a unified ontology creating machine-interpretable relationships between building elements, code requirements, specifications, and approved materials while maintaining each resource's existing organizational logic. Key strategic elements include:

- **Metamodel Development:** A core ontology defining common classes and properties shared across all resources, ensuring interoperability and enabling automated reasoning about compliance and requirements.
- **Resource-Specific Data Models:** Detailed data models for NMCC, NMS, and CCMC, each with use cases, activity diagrams, and entity-relationship structures that capture domain-specific requirements while enabling cross-resource integration.
- **Consolidated Data Model:** A unified framework that links regulatory clauses to specification templates and product evaluations through shared concepts such as building elements, properties, requirements, and test methods.
- **NRC Data Dictionary:** A high-priority deliverable that will provide explicit definitions, persistent unique identifiers, and semantic annotations for every term, property, and relationship across all resources, aligned with international standards including buildingSMART Data Dictionary (bSDD) and IFC schemas.

To achieve true machine-interpretability of NRC key resources, the report presents format requirements based on principles of open data and international best practices:

- **Core Requirements:** Structured syntax following recognized standards (ISO, W3C), unambiguous semantics, robust schema validation (XML Schema, JSON Schema, SHACL), and explicit data type definitions.
- **Technical Characteristics:** Deterministic parsing, namespace support, version control, explicit character encoding (UTF-8), and well-defined error handling.
- **Semantic Considerations:** Ontology-based definitions, cross-schema mapping capability, persistent globally unique identifiers (URIs), and support for machine-executable logic enabling automated compliance checking.
- Recommended Formats: Open, standards-based formats including XML, JSON, RDF, JSON-LD, IFC (ISO 16739-1:2024), and domain-specific formats such as GeoJSON for spatial data and SHACL for validation rules.

To maintain momentum and ensure successful delivery of the NRC DCP, the report recommends immediate action on the following activities:

1. **Establish Governance Structure** (activities 3.1.01, 3.3.02, 3.3.03, and 3.3.04): If not already done, form a steering committee as well as technical and content working groups with representatives from NMCC, NMS, CCMC, and NRC IT services to oversee the project's implementation, ensuring alignment with the PDCSS objectives and industry needs. Develop project charter and risk management and mitigation strategies. As part of this work, the editorial board for the NRC Data Dictionary should be identified and mobilized as indicated in section 3.
2. **Continue Preparatory and Foundational Work**
    a. Digital Content Development (activities 1.1.03, 1.1.04, 1.1.05, 1.1.07, 1.1.08, 1.1.09, 1.1.10, 1.1.12, 1.1.13, 1.1.14): Continue developing the structure, standardized formats, and begin digitization of content. Focus on the development of the NRC data dictionary and metadata repositories.
    b. Supporting resources (activities 1.2.01 to 1.2.04): Confirm semantic and syntactic resources for the CDF in line with the NRC data dictionary and metadata repository
    c. Digital process formalization (activities 1.3.01, 1.3.02 and 1.3.03): Confirm and refine digital workflows and processes supporting content development, curation and distribution.
3. **Technical Infrastructure Development**
    a. Digital functions and services (activities 2.1.01, 2.1.07 and 2.1.08): Confirm/develop functional requirements for DCP for all three resources.
    b. Repositories and integrations (activities 2.2.03, 2.3.01, and 2.3.02): Initiate the design and implementation of the core digital infrastructure, including data repositories, API frameworks, and integration mechanisms.
4. **Stakeholder Engagement** (activity 3.3.01): Develop and implement a stakeholder engagement plan to ensure buy-in and collaboration from all relevant parties throughout the development process.
5. **Pilot Projects**: Initiate pilot projects for key components of the DCP, particularly in areas such as digital code compliance checking and e-permitting, to test concepts and gather early feedback.

The blueprint presented in this report provides a pathway from current static and document-centric resources to a fully realized NRC DCP, supporting dynamic and interconnected data assets, that will serve as a cornerstone of Canada's digital construction ecosystem.

1. Introduction

The National Research Council of Canada (NRC) has introduced the Platform to Decarbonize the Construction Sector at Scale (PDCSS) as a response to the Canadian built asset industry's significant challenges, which include stagnating productivity, poor performance and inefficient use of resources, among others. As part of the PDCSS, the Construction Sector Digitalization and Productivity Challenge program (CSDP), aims to accelerate digitalization in construction through research on performance-based codes, developing a digitalization roadmap, and exploring digital tools like portals and virtual inspections (Canada, 2023). One of the key initiatives within the CSDP is the NRC's Digital Construction Platform (NRC DCP), a digital platform which aims to promote digitalization in support of decarbonization and productivity increase in the Canadian built asset industry by digitalizing its three key resource areas for the construction sector: the National Model Construction Codes (NMCC), the National Master Specifications (NMS), and the evaluations and other resources produced by the Canadian Construction Materials Centre (CCMC). The NRC DCP, as well as a R&D roadmap guiding its development, was presented in a report published by NRC in October 2024 (Poirier et al., 2024). Once implemented, the NRC DCP will:

1. Provide a robust digital infrastructure that supports the management and dissemination of construction codes, standards, and specifications in digital and machine-interpretable formats.
2. Enhance interoperability among various digital tools and platforms used by industry professionals, regulatory bodies, and government agencies.
3. Promote sustainability through the integration of decarbonization strategies within the construction sector.
4. Streamline regulatory compliance processes using automated tools and enhanced collaboration features.
5. Promote and enable collaboration across NRC departments while centralizing and ensuring access to key resources.
6. Establish a foundational architecture and resource pool for the development of interoperable platforms and software tools within the Canadian built asset industry.

The digital ecosystem provided by the DCP will enhance the efficiency, transparency, and sustainability of the Canadian built asset industry. As a modular and scalable platform, the DCP leverages widely accepted resources, standards, and services to ensure a robust and interoperable ecosystem. Ultimately, the DCP will enable the successful integration of digital specifications, automated compliance checking, access to key resources and improved data management practices across the Canadian built asset industry and facilitate better stakeholder collaboration and support the decarbonization goals of the construction sector.

The NRC DCP R&D roadmap articulates 198 activities over a six-year period, from 2024 to 2030, into three main streams, including digital resources, which involves digitizing NRC's resources, developing supporting tools, and formalizing digital processes, digital platforms, which includes developing essential functions and services, establishing data exchange standards and integrations, and implementing repositories for data storage and management, and platform delivery, which covers project governance, capability development, and communication strategies. The roadmap is organized into phases, beginning with foundational preparatory work and progressing through subsequent stages that expand platform capabilities and services.

As part of this foundational work, this report presents two activities that aim to establish a detailed digitalization strategy, by performing an audit of existing NRC content (NMCC, NMS, CCMC) (Activity 1.1.01 in the NRC DCP R&D roadmap) and identifying machine-interpretable format requirements for each resource type (1.1.02 in the NRC DCP R&D roadmap). The first activity involved a thorough review and analysis of all existing content across the NMCC, the NMS, and the CCMC materials database. This audit assessed the current state of digitization, identified gaps, and determined priorities for the digitalization process. Key tasks performed included cataloging all content types, evaluating their current format and structure, and assessing their suitability for machine-interpretable conversion. Based on this audit, the second activity focuses on determining the most appropriate machine-interpretable formats for each type of content. This involved researching industry standards, evaluating compatibility with existing systems, and considering future interoperability needs.

This report begins with the results of a current state assessment of NMCC, NMS, and CCMC content to evaluate existing formats, structures, and machine-interpretability limitations. It then discusses international standards and best practices to specify technical requirements for machine-interpretable formats. Each resource is then developed around specific use cases that were initiated in the NRC DCP R&D roadmap report. This includes the identification of key stakeholders and their functional interactions with each resource, activity diagrams and process models documenting end-to-end workflows from content creation through publication. A foundational metamodel, resource-specific entity-relationship models, and a consolidated model unifying all three resources through shared concepts are then presented. Finally, metadata requirements and data dictionary structures to enable provenance tracking, versioning, and governance, collectively forming the foundation for the Common Data Framework (CDF) were developed.

2. Current state of digitalization of NRC resources

As laid out in the current state review in the NRC DCP report, the NRC's current digital resources are faced with several issues to be addressed in moving towards fully digital and machine-interpretable resources. These issues include "[...] data complexity and volume, lack of uniformity, siloed datasets, duplication, interoperability challenges, and technical infrastructure limitations. Additionally, concerns were identified regarding the accuracy and consistency of data, the allocation of resources, and the need for improved stakeholder engagement and regulatory compliance." (Poirier et al., 2024, p. 26). Of these challenges, two pertain to the digitalization of NRC resources: (1) data complexity and volume, and (2) uniformity, harmonization, and interoperability of content.

Regarding data complexity and volume, the issue that was highlighted pertained to the high level of detail and complexity of codes, specifications and product datasets as well as the large volume of information that requires digitalization across all three resources. In the NRC DCP report, it was stated that one of the key requirements for the DCP was for it to be "[...] able to contend with this complexity and large volume of data and information in a simplified and efficient manner. It must provide capabilities to link important data sets between the three resources in the cases where such links are necessary to support a specific purpose (e.g., product data linked to a specification section in response to a code provision). [...]" (Poirier et al., 2024, p. 26). Regarding uniformity, harmonization, and interoperability of content, the issues that were highlighted pertained to the lack of uniformity and harmonization could cause issues in interpretation and more importantly in digitalization. In the NRC DCP report, it was stated that one of the key requirements for the DCP was for it to facilitate the uniformity, harmonization, and interoperability of all NRC content.

As a reminder, the NRC's resources to be digitalized within the NRC DCP are the codes (NMCC), the master specifications (NMS) and the material database (CCMC). Each resource is developed and managed independently within one of NRC's branches and is structured according to a domain specific logic. Each resource is currently digitalized to a different level and extent. Table 1 summarizes the current state of NRC resource digitalization. All three resources are well structured, being comprised of multiple volumes, chapters, sections, articles, etc. However, each resource operates independently and is not semantically linked. This lack of semantic linking between all three resources limits the potential for their machine-interpretability.

Table 1. Current state of NRC resource digitalization

	NMCC	NMS	CCMC
Description	Designed to ensure the performance of buildings, with a specific focus on occupant safety.	Standardized construction project specification templates for both systems and products	Technical information on construction product compliance with national and provincial codes on NRC website
Current format and structure	5 volumes with distinct divisions, parts, sections, subsections, articles and statements (clauses).	864 specific resources distributed across Reference guides (2), Design specifications (103) and Construction specifications (759)	Two main types of entries: reports (247) and listings (208)
Content type	Directive regulatory statements, explanatory statements	Depending on specification type, each resource contains articles that relate to reference standards, product descriptions or submittals	Technical product information as it relates to performance for certain types of requirements
Current state of digitalization	Authored in XML, available in .pdf and .xml formats, soon HTML	Authored in XML, available in .pdf, .docx or .rtf	Authored in HTML, available in HTML and .json
Machine-interpretability	High degree of conversion yet complex. Highly relevant	High degree of conversion. Medium relevancy (product specifications highly relevant,	High degree of conversion. Highly relevant
Priorities for the digitalization process	1. Classify clauses 2. Organize clauses 3. Interpret clauses 4. Express clauses 5. Prepare rulesets	1. Prioritize relevant sections and domains 2. Extract necessary data points 3. Harmonize data points across sections 4. Codify data points	1. Extract metadata, properties and product types 2. Define data structure for reported and listed products 3. Harmonize metadata and property requirements 4. Codify product information

2.1. NMCC

The NMCC is digitized in the sense of being available electronically as a PDF, however it remains a document designed for human consumption rather than machine processing. Its potential for automated compliance checking, integration with digital tools and modern digital construction workflows is therefore limited.

Formats and Structure

The codes are available through the NRC Publications Archive in free electronic PDF format, with print versions available for purchase. They are structured as conventional technical documents. The documents are structured following a traditional hierarchical organization. All codes share a consistent three-division structure:

- **Division A**: Defines scope, objectives, functional statements, and compliance conditions
- **Division B**: Contains acceptable solutions (technical requirements)
- **Division C**: Contains administrative provisions

They are further organized following a consistent hierarchical numbering system: Part, Section, Subsection, Article, Sentence level, Clauses and Subclauses.

Content Types

The codes are generally composed of technical specifications and requirements, tables and numerical data (load values, dimensions, ratios), definitions and terminology, and commentary and user guides. There are also references to external standards.

Current State of Digitalization

The current level of digitalization of the NMCC can be considered LOW to MEDIUM based on the assessment described in Table 2.

Suitability for Machine-Interpretability

The current level of machine-interpretability of the NMCC can be considered POOR based on the limitations described in Table 3.

Table 2. Current state of digitalization of NMCC resources

Feature	NBC	NFC	NPC	NECB	Status
Electronic PDF	✓	✓	✓	✓	Available free
Online Viewing Platform	✓	✓	✓	✓	Via ICC Digital Codes, TRAX, soon CBHCC
Print Version	✓	✓	✓	✓	Purchasable
Structured Data (XML/JSON)	✓	✗	✗	✗	Available through licensing agreement
Database Format	✗	✗	✗	✗	Limited conversion of sentences to relational database
API Access	✗	✗	✗	✗	Not available
Machine-interpretable Metadata	✗	✗	✗	✗	Not available
Linked Open Data	✗	✗	✗	✗	Not available

Table 3. Limitations with machine-interpretability of the NMCC

Limitation	Description
Document-Centric Format	• All four codes are PDF documents designed for human reading • Linear text structure with embedded tables and references • Lack of semantic markup or structured metadata • Cross-references embedded in text (e.g., "See Article 3.5.2.1.")
No Computational Infrastructure	• Lack of APIs for querying code provisions • Lack of structured databases of requirements • Lack of machine-interpretable taxonomy or classification system • Lack of unique persistent identifiers for code provisions
Complex Interrelationships	• Extensive external standard references (e.g., CSA, ASTM, ISO, NEMA standards) • Cross-references between the four codes • Provincial amendments create code variants • Referenced standards not integrated or linked
Limited Search Capabilities	• Basic PDF text search only • Lack of semantic search or natural language querying • Lack of filtering by building type, occupancy, or requirement category • Lack of automated compliance checking capability
Limitations for automated compliance checking	• Cannot programmatically extract applicable requirements for a specific building type • Cannot automatically trace requirement dependencies • Cannot compute cumulative requirements across multiple code sections • Cannot validate design parameters against code limits

Limitations for BIM integration	• No standardized data exchange formats • No IFC (Industry Foundation Classes) mapping • No integration with design software • Manual code interpretation required
Limitations for AI/ML Applications	• Unstructured text not suitable for training • No labeled datasets of requirements • Cannot extract structured rules for expert systems • Semantic relationships not encoded

2.2. NMS

The NMS is currently authored using XML with a schema for document structure with little metadata. The document-based delivery (PDF/RTF/DOCX) to users and xml to licensed publishers. While the content is comprehensive and well-organized using industry-standard classification systems (MasterFormat/UniFormat), its current formats have low machine-interpretability. The active digitalization initiative by NRC signals a shift toward BIM-compatible, machine-interpretable formats.

Formats and Structure

The NMS is currently available in PDF, RTF (Rich Text Format) and DOCX formats. The DOCX format, which is used by staff Technical Advisors and external subject matter experts for collaborative development of content updates, is converted to an XML format at the end of each semi-annual update cycle, and provided to selected third party providers of specification viewing tools/services. Publishers make NMS available as Rich Text Format files for use with word processing software or as files for use with proprietary software

It follows the MasterFormat (50 Divisions with 6 or 8-digit section numbers) and the UniFormat (9 main groups of level 1 elements) classification systems, with sections structured according to SectionFormat (three-part format) and PageFormat for layout. The NRC database contains approximately 860 specification sections comprising about 7,000 pages in each language (English and French).

Content Types

The NMS includes two primary specification types: prescriptive specifications which are detailed descriptions of exact materials, systems, and installation processes and performance specifications which are requirements focused on functional outcomes. Content includes standardised parts, bracketed alternative text, guidance notes, reference standards, project options presented in square brackets, and cross-references to related sections.

Current State of Digitalization

The current level of digitalization of the NMS can be considered LOW to MEDIUM since the current XML, DOCX, and RTF formats represent traditional document-based content delivery with limited structured data capabilities. This assessment is based on the fact that the content is currently only human-readable and editable, it is not natively machine-interpretable, it requires manual extraction and processing for computational analysis and is designed for human specification writers rather than automated systems.

The NMS is available through various software platforms including NBS Chorus (cloud-based specification software), SPECedit (support ended December 31, 2023), and Word document libraries with macros. Some publishers like Chalkline offer cloud-based visual specification systems that store building model and NMS specification data on cloud servers.

Although XML is converted to DOCX for third party specification viewer, the conversion of frequent content updates is manual, time consuming and not scalable. NRC is currently digitalizing the NMS, including digitalized lifecycle management of content updates, to provide industry and regulators with greater access to content in machine-interpretable formats, including a BIM compatible format.

Suitability for Machine-Interpretability

The current level of machine-interpretability of the NMS can be considered POOR based on the limitations described in Table 4.

Table 4. Limitations with machine-interpretability of the NMS

Limitation	Description
Document-Centric Format	• Although authored and maintained in XML and publishable as such, it is primarily published as DOCX and RTF, which are primarily presentation formats, not structured data formats. Moreover, the XML format doesn't leverage metadata to enable machine-interpretability.
Text-based content	• Specifications are written in natural language with editorial conventions. However, technical sections have been moving towards a consistent organizational structure based on SectionFormat, with consistent use of many article titles and certain terms to aid digitalization.
Manual processing required	• Options in brackets and notes require human interpretation
No native API, partially structured schema	• Content isn't exposed through standardized data exchange formats • NMS XML does have a schema (XSD), although it isn't standardized with other systems

2.3. CCMC Product registry

The CCMC Product registry is currently in a transitional state characterized by early-stage digital transformation. While evaluation content remains delivered as structured PDFs that offer moderate human readability but poor machine interpretability, the NRC has made progress by implementing a JSON API that provides good machine-interpretable access to assessment registry metadata including product information, CCMC numbers, and MasterFormat classifications. Although a significant portion of CCMC data is managed in a relational database with well-structured schemas, workflows and internal reporting, the external distribution of detailed technical evaluation content, test results, and compliance opinions is largely in PDF format with limited suitability for automated processing.

Formats and Structure

The CCMC evaluations are currently published online in HTML format or as PDF documents which are human-readable but have limited machine-interpretability. They are available through a browsable web registry that lists all active CCMC product assessments, with filters (company, material, MasterFormat, status, etc.). Each item shows structured fields like product name, code edition(s), MasterFormat, and Technical Guide references.

CCMC has implemented a JSON API for its Registry of Product Assessments, which provides machine-interpretable access to assessment metadata. The JSON API structures information according to defined specifications, with each API record representing one CCMC product assessment.

Content Types

The CCMC produces two main types of assessments: (1) evaluation reports for products without existing standards, and (2) evaluation listings for products covered by specific standards. Each assessment links to an HTML or PDF evaluation report or listing which follows a standardized structure with sections including product descriptions, compliance opinions based on National Building Code requirements, conditions and limitations, technical evidence including material properties and performance requirements, and manufacturer information.

In its push to digitize, the CCMC has developed metadata fields for the JSON API: product, company, applicable code edition(s), Technical Guide IDs, MasterFormat® numbers, and status (active/withdrawn available on request).

Current State of Digitalization

The CCMC curates an online Registry of Product Assessments which contains over 450 active product evaluations. The registry is searchable by CCMC number, company name, product name, or product category. Moreover, products are indexed using the MasterFormat system. Assessments are “active” if they are present there, and withdrawn versions are accessible by request.

The primary deliverable remains PDF (human-readable). However, the registry pages themselves present semi-structured HTML cards/tables that surface key fields for browsing.

The CCMC JSON API provides structured access to CCMC numbers, organization names (English/French), product names, evaluation directive documents, MasterFormat numbers, assessment status, and website information.

Suitability for Machine-Interpretability

The current level of machine-interpretability of the CCMC Product registry can be considered LOW based on the limitations described in Table 5.

Table 5. Limitations with machine-interpretability of the CCMC

Limitation	Description
Limited Machine Readability	• The core evaluation reports remain in PDF format, which contains unstructured text, tables, and figures that are not easily machine-interpretable. • Technical data is presented in tables within PDFs rather than structured data formats. • HTML web pages provides content online.
Structured Data	• The JSON API represents a significant step toward machine-interpretability by providing metadata about assessments in a structured format. • Many fields are already structured (company, product, MasterFormat®, Technical Guide, applicable code editions, status), reside in a mature relational database structured, and can be programmatically parsed from the registry pages. • However, CCMC data is mostly siloed from NMS and NMCC content repositories, leading to duplication of certain NMS and NMCC data in the CCMC database and inefficient data refreshes when NMS and NMCC updates are published.

3. Overall digitalization strategy

As part of the NRC DCP, a three-tiered architecture is proposed, with each tier being "[...] pivotal in managing and leveraging data within the NRC Digital Construction Platform [Figure 1]. These tiers comprehensively address key aspects of data management and utilization, laying the foundation for effective digital transformation and enhanced interoperability across the built asset industry." (Poirier et al., 2024, p. 30). Central to the digitalization of the NRC'S resources is the Common Data Framework (CDF) (the first tier of the DCP). The CDF "encompasses essential components such as syntax, semantics, concepts, processes and workflows, practices and methodologies, and services. The goal is to ensure robust and coherent data organization that promotes interoperability across various dimensions of the built asset industry and its digitalization. By adopting standardized resources and practices, this tier lays the foundation for consistent data management and seamless integration of information. As such, it not only lays the foundation for the development of the DCP, but also for the development of platforms by other interested parties utilizing common resources." (Poirier et al., 2024, p. 30).

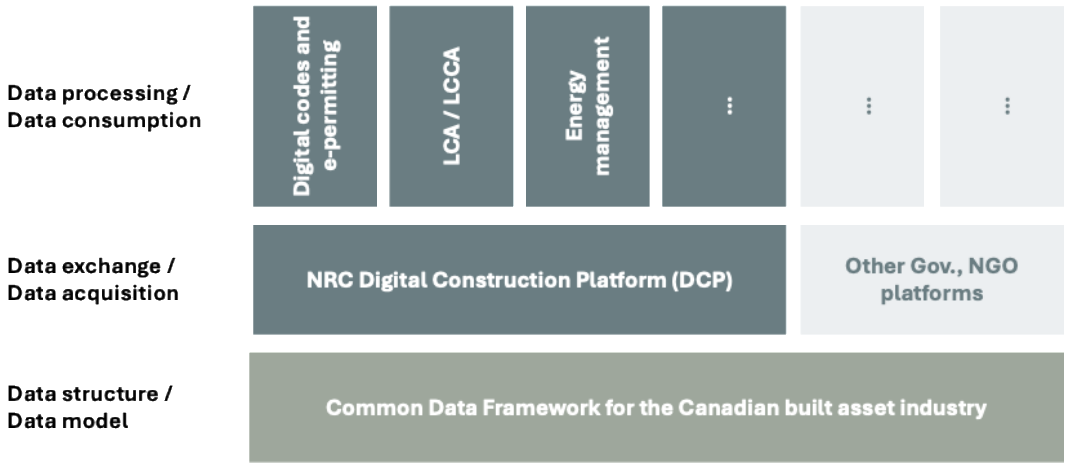

Figure 1. NRC's Digital Construction Platform layered framework (Taken from Poirier et al. 2024)

The digitalization of the NRC's resources will therefore be addressed through the CDF which provides a layered semantic infrastructure for creating machine-interpretable relationships between building elements, code requirements, specifications, and approved materials. The CDF encompasses three interconnected layers: (1) a taxonomy layer that establishes controlled vocabularies and classification hierarchies for NRC resources; (2) an ontology layer that defines semantic relationships, constraints, and reasoning rules between concepts; and (3) a mapping layer that aligns NRC-specific terms with external standards such as IFC and BOT. Through the CDF, existing ontologies will be evaluated and adopted where suitable, and new ontologies will be developed where gaps exist. These semantic structures will enable cross-referencing between resources and support automated reasoning while preserving the

existing organizational logic. The CDF will also support evolving digital processes, workflows, and methodologies as the DCP matures, as described in Poirier et al. (2024). Once digitalized, strategies to make the resources available, through APIs or others, as well as validate them should be developed.

To establish this core ontology and the different mappings to support the transition towards digitalized and machine-interpretable resources a series of activities should be undertaken. The digitalization strategy follows a progressive, building-block approach where each activity establishes the foundation for subsequent work. The strategy proceeds through five sequential phases: (1) validating requirements through use case development, (2) translating requirements into system functions, (3) formalizing information structures through data modelling, (4) establishing governance through metadata requirements, and (5) operationalizing the framework through pilot data dictionary development. This phased approach ensures that semantic infrastructure is grounded in validated user needs and aligned with technical implementation realities. Figure 2 illustrates this progression and the activities are described below. A first iteration of these activities was undertaken to develop the data models discussed in section 4.

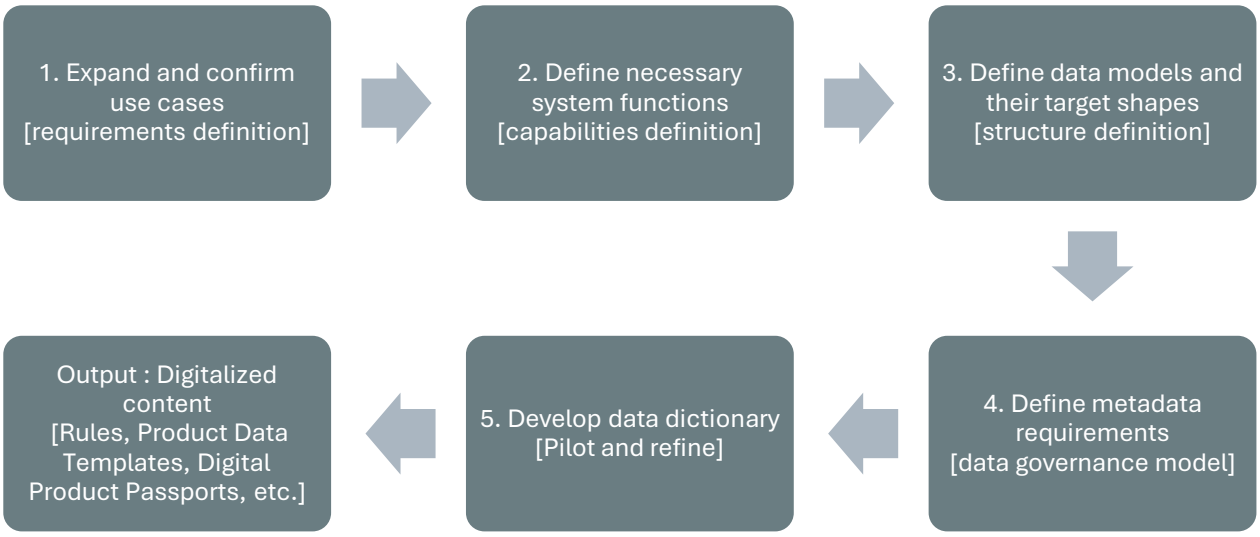

Figure 2. Digitalization strategy

1. Expand and confirm use cases: This activity involves collaborating with stakeholders to expand upon initial use case identified in the NRC DCR roadmap report and validate their completeness and accuracy. Workshops and interviews with key users can be conducted to elaborate on each use case, detailing actors, preconditions, main flows, alternative paths, and expected outcomes as laid out in ISO 29481-1:2025. The confirmation process ensures that all critical business scenarios are captured, prioritized appropriately, and agreed upon by stakeholders before proceeding with system design and development. Key tasks include:

1. Review and inventory existing use cases from the NRC DCR roadmap report
2. Conduct stakeholder workshops to identify gaps and validate priorities
3. Document each use case with actors, preconditions, main flows, and expected outcomes
4. Establish traceability between use cases and downstream system requirements

2. Define necessary system functions: Based on the identified use cases, this step defines the specific technical functions the system must provide. Capabilities such as searching and filtering data, sending notifications, managing access permissions, and generating exports should be defined. Each function should directly support one or more use cases and represent a discrete capability that developers can implement within the roadmap. Key tasks include:

1. Map each use case to required system capabilities
2. Define functional requirements for data access, querying, and filtering
3. Specify integration points (APIs, authentication, notifications)
4. Prioritize functions based on use case criticality and implementation complexity

3. Define data models and their target shapes: This activity establishes the logical structure of information within the system by creating classes and their attributes. Core entities like Item, Milestone, User, and Dependency, should be defined and properties for each entity (such as title, status, dates, owner) should be derived. This creates a blueprint for how data will be organized, related, and stored in the system. At this point a canonical model, instantiated through RDF/OWL and JSON-LD, can be defined. Key tasks include:

1. Identify core entities and their relationships (e.g., Item, Milestone, User, Dependency)
2. Define attributes and cardinality for each entity
3. Develop a canonical data model using RDF/OWL and JSON-LD serialization
4. Align data models with existing standards where applicable (e.g., IFC, BOT)

4. Define metadata requirements: At this stage, additional contextual information needed to manage and govern the data effectively is defined. This includes temporal metadata (creation and modification dates), ownership details (who created or edited items), classification tags, compliance information, and data lineage. Unique identifiers can be defined for each instance. Metadata ensures traceability, supports audit requirements, and provides the context needed for proper data management. Key tasks include:

1. Define temporal metadata (creation, modification, versioning timestamps)
2. Establish ownership and provenance tracking (creator, editor, source)
3. Specify classification and tagging schemes for discoverability
4. Assign unique persistent identifiers (URIs/IRIs) to each resource instance

5. Pilot data dictionary development: This final activity creates a comprehensive reference that defines every data field in detail. For each field, the name, data type, length constraints, whether it's required, its business meaning, valid values, and examples will be listed. The data dictionary serves as the authoritative source of truth for developers, analysts, and business users, ensuring everyone interprets and uses data consistently throughout the project lifecycle. Within the CDF, a taxonomy layer should be deployed supporting the deployment of the data dictionary. In performing the analysis, classes and properties can be mapped to exiting schemas (eg. IFC 4.3 add 2) or ontologies (eg. BOT). Lastly, within the data dictionary, unique persistent IDs for every concept/term/clause/property should be defined. Key tasks include:

1. Establish a governance structure and strategy for the data dictionary
2. Prepare content for ingestion and parsing
3. Initiate segmentation and parsing
4. Develop a delivery framework and interface

Once the data dictionary, and the broader CDF, is developed, work around digitalization of the content, namely the preparation of rules as well as formalization of product data templates or digital product passports within the CCMC registry can be undertaken.

4. Towards the NRC CDF: Preliminary digitalization step

As part of this project aimed at moving towards a fully digitalized and machine-interpretable set of NRC resources underscored by the CDF, a preliminary digitalization step was performed. This step consisted in developing a preliminary set of use cases, activity diagrams and a data model for each resource. Moreover, a consolidated data model was developed and a set of metadata requirements specified.

The three sub-models, the NMCC, the CCMC and the NMS, are related through the meta model. Across the models Building Elements, represented mainly as Product (or ProductType) in the models, serve as the central hub connecting codes, specifications, and materials. Hierarchical relationships, both within and across resources, maintain existing organizational logic while enabling cross-references. Another key aspect that is the development of the model embraces the notion of Separation of Concerns, whereby each resource maintains its native structure while being related through common concepts and relationships.

4.1. Metamodel

The Metamodel provides common language for digitalization of the resources by defining core classes and properties shared across all resources, ensures interoperability, by enabling NBC, NMS, and CCMC data to interoperate across domains and with others through a shared ontology (i.e., a formal, explicit specification of shared concepts, relationships, and constraints that provides a common vocabulary across resources), and extensibility, allowing domain-specific sub-classes without impacting the foundational data model. In essence, the metamodel acts a “spine” that connects different regulatory, specification, and product datasets, their classes and attributes.

The Meta model aims to normalizes semantics (e.g., “ProductType” means the same thing in NBC, NMS, and CCMC) by allowing linking constraints to evidence, meaning that it is possible to see not only what’s required but whether and how a product meets it, and ultimately is key in supports automated checking, whereby requirements can be converted to machine-interpretable rulesets. Table 6 defines the classes and their purpose in the model. Table 7 defines the attributes, the classes they belong to and their types in the model, whereas Table 8 defines the relationships between the classes and their cardinalities. Figure 3, Figure 4, and Figure 5 depict the Metamodel.

- **Sources & provenance:** A SourceDocument is publishedBy one Organization. Specification Section and Product each carry Provenance , and every Provenance is recorded by one Agent.

- **From clauses to requirements**: Each RegulatoryClause defines one Requirement . A Requirement specifies one or more Specification Section(s) *and appliesTo one or more ProductType(s)*.
- **Requirements and constraints**: A Requirement represents a human-authored intent expressed in natural language, while a Constraint is a machine-executable check derived from one or more Requirements. Each Constraint is linked to its source Requirement(s) via the *derivedFrom* property, enabling traceability from automated validation results back to regulatory intent.
- **Products:** A Product may referTo one or more ProductType(s).
- **Properties, methods, and evidence**: A Property is linked to one or more ReferenceStandard(s) and is associated with a Link. A PropertyValue may reference a TestMethod and is supported by zero or more Evidence records; each Evidence is a TestResult with optional Unit and TestMethod and is tied to a Property.
- **Standards ecosystem**: A TestMethod is published by an Organisation and may implement a ReferenceStandard; a ReferenceStandard may referTo a TestMethod.

The metamodel ensures semantic consistency across NBC, NMS, and CCMC by mapping source-specific terminology to common classes. For example, an NBC "Article" (e.g., 9.10.9.6) is represented as a RegulatoryClause, an NMS "Section" (e.g., 09 29 00) is represented as a SpecificationSection, and a CCMC "Evaluation Report" is represented as Evidence linked to a Product

This normalization allows queries like "find all products that satisfy requirement X" to operate across all three resources without source-specific logic. The common Requirement class serves as the bridge: a RegulatoryClause *defines* a Requirement, which in turn *appliesTo* one or more ProductTypes and is *specifiedBy* one or more SpecificationSections. Examples of other types of queries and functions the meta model aims to support, whereby the classes and their relationships in the metamodel supports one or more function, include :

- **Regulatory traceability:** Which code clause(s) define the requirements for a given product type?
- **Specification mapping:** Which NMS sections specify how to meet a given NBC requirement?
- **Product compliance:** Does product X have evidence demonstrating compliance with requirement Y?
- **Evidence provenance:** What test method was used to generate this evidence, and who performed the test?
- **Standards ecosystem:** Which reference standards define the test methods for a given property?
- **Temporal validity:** What requirements were in effect for a building permitted in 2018?
- **Classification lookup:** What products fall under MasterFormat division 09 29 00?

Regarding versioning, all entities in the metamodel are assigned persistent IRIs that remain stable across versions. When an entity is updated, the new version receives a new IRI while maintaining a supersedes link to the previous version. This approach ensures that historical references remain resolvable,

compliance evidence tied to older versions remains valid and traceable and temporal queries (e.g., "what was required in NBC 2015?") are supported.

Table 6. Metamodel classes

Class Name	Purpose	Role
SourceDocument	Represents a source document containing regulatory or specification information	Serves as the root container for all regulatory and specification content. Examples include NBC 2020, NMS Section 09 29 00, or a CCMC Evaluation Report. Enables tracking of publication metadata and organizational ownership.
Organization	Represents an organization (publisher, manufacturer, etc.)	Captures the entities responsible for publishing codes, standards, specifications, or products. Used to establish authority and provenance (e.g., NRC, CSA, ULC, or a product manufacturer).
RegulatoryClause	Represents a regulatory clause from codes or standards	The unit of regulatory content extracted from codes like the NBC. Each clause carries its own identifier, text, and effective date range, enabling precise referencing and temporal queries. A clause may be prescriptive (specifying exact solutions) or performance-based (specifying outcomes).
Requirement	Represents a specific requirement derived from a clause	The normalized expression of what must be achieved. While a RegulatoryClause contains the original legal text, a Requirement captures the machine-interpretable intent — what property, threshold, or condition must be satisfied. This separation allows multiple clauses to map to the same requirement and supports translation into executable constraints.
Constraint	Represents a machine-executable check derived from a Requirement	The formal, executable expression of a Requirement in a language suitable for automated validation (e.g., SHACL, SPARQL, IDS). While a Requirement expresses human-authored intent, a Constraint translates that intent into computable logic that can be evaluated against product data or building models. Each Constraint links back to its source Requirement(s) via *derivedFrom*, ensuring traceability from validation results to regulatory intent.
SpecificationSection	Represents a section within a specification document	Links to NMS or similar specification content that describes *how* to meet a requirement. For example, NMS 09 29 00 (Gypsum Board) specifies materials and methods that satisfy fire-resistance requirements.

Provenance	Tracks origin and creation information for entities	Provides audit trail and lineage for any entity in the model. Records who created or modified data, when, and from what source. Essential for trust, accountability, and change management.
Agent	Represents a person or system that creates or modifies entities	Distinguishes between human actors (e.g., a code official, specification writer) and automated systems (e.g., a parsing tool, validation engine). Supports attribution and accountability in provenance tracking.
ProductType	Represents a type or category of product	An abstract classification of products (e.g., "Type X Gypsum Board," "Fire Door Assembly") independent of any specific manufacturer. Requirements apply to ProductTypes, enabling generic compliance rules that any conforming product can satisfy.
Product	Represents a specific product from a manufacturer	A commercially available product with a manufacturer, model number, and associated evidence. Products are linked to one or more ProductTypes to indicate which categories they belong to.
Classification	Organizes products/specifications using standard systems	Aligns entities with industry classification systems (MasterFormat, UniFormat, OmniClass) to support discovery, filtering, and interoperability with existing construction industry tools and databases.
Property	Represents a measurable or definable characteristic	A named attribute that can be measured or specified (e.g., fire-resistance rating, thermal conductivity, thickness). Properties are defined independently of any specific value, enabling reuse across requirements, products, and specifications. Each property can be mapped to external dictionaries.
PropertyValue	Represents a specific value for a property	The actual measured or declared value of a property for a specific product or context. Includes optional unit, test method, and date to capture how and when the value was determined.
Unit	Represents a unit of measurement	Standardized units (e.g., hours, mm, kPa) aligned with QUDT or OM ontologies to ensure consistent interpretation of numeric values across systems.
ReferenceStandard	Represents a reference standard or code	External standards (e.g., CAN/ULC-S101, ASTM E119) that define test methods, material specifications, or performance criteria. Standards are published by organizations and may be referenced by properties, requirements, or test methods.
TestMethod	Represents a standardized testing method	The procedure used to measure a property or verify compliance (e.g., CAN/ULC-S101 for fire endurance testing). Links evidence to the method by which it was obtained, supporting reproducibility and trust.

Evidence	Represents evidence supporting claims or values	Documentation that substantiates a property value or compliance claim. May be a test report, calculation, certificate, or other artifact. Each evidence record is linked to the property it supports and optionally to the test method used.

Table 7. Metamodel attributes

Class	Attribute	Data Type	Description
Source Document	id	IRI	Unique identifier for the source document
	title	string	Title of the source document
	edition	string	Edition of the source document
	publishedOn	date	Publication date of the source document
	version	string	Version identifier of the document
Organization	id	IRI	Unique identifier for the organization
	name	string	Name of the organization
Regulatory Clause	id	IRI	Unique identifier for the regulatory clause
	type	string	Type of regulatory clause
	label	string	Label of the regulatory clause
	text	string	Text content of the regulatory clause
	validFrom	date	Date from which this version is valid
	validTo	date	Date until which this version is valid (null if current)
	supersedes	IRI	Reference to the previous version of this clause
	supersededBy	IRI	Reference to the newer version (if deprecated)
Requirement	id	IRI	Unique identifier for the requirement
	label	string	Label of the requirement
	text	string	Text content of the requirement
	version	string	Version of the requirement
Constraint	id	IRI	Unique identifier for the constraint
	expression	string	Formal expression of the constraint (e.g., SHACL shape, SPARQL query)
	language	{SHACL, SPARQL, IDS}	Language used to express the constraint
	derivedFrom	Requirement	The requirement from which this constraint is derived
	severity	{violation, warning, info}	Severity level if constraint is not satisfied

Specification Section	id	IRI	Unique identifier for the specification section
	number	string	Number of the specification section
	title	string	Title of the specification section
Provenance	id	IRI	Unique identifier for provenance information
	createdBy	Agent	Agent who created the entity
	createdOn	date	Date when the entity was created
	derivedFrom	IRI	Source from which the entity was derived
Agent	id	IRI	Unique identifier for the agent
	name	string	Name of the agent
	role	string	Role of the agent
ProductType	id	IRI	Unique identifier for the product type
	label	string	Label of the product type
	mapping	URI	mapping for the product type (to bSDD or other)
Product	id	IRI	Unique identifier for the product
	name	string	Name of the product
	manufacturer	Organization	Organization that manufactures the product
	identifier	string	Identifier for the product
	version	string	Version or revision of the product
Classification	system	{MasterFormat / UniFormat}	Classification system (MasterFormat or UniFormat)
	code	string	Code within the classification system
	title	string	Title of the classification
Property	id	IRI	Unique identifier for the property
	label	string	Label of the property
	definition	string	Definition of the property
	unit	Unit?	Kind of unit for the property (optional)
	mapping	URI	mapping for the property (to bSDD or other)
PropertyValue	id	IRI	Unique identifier for the property value
	value	any	Value of the property
	unit	Unit?	Unit of the property value (optional)
	date	date?	Date of the property value (optional)
Unit	id	IRI	Unique identifier for the unit
	symbol	string	Symbol representing the unit
	system	QUDT/OM	Unit system (QUDT or OM)
ReferenceStandard	id	IRI	Unique identifier for the reference standard
	code	string	Code of the reference standard
	edition	string	Edition of the reference standard

TestMethod	id	IRI	Unique identifier for the test method
	code	string	Code of the test method
	title	string	Title of the test method
	publisher	Organization	Organization that published the test method
Evidence	id	IRI	Unique identifier for the evidence
	kind	{text, calc, report}	Kind of evidence (text, calculation, or report)
	link	URL	URL link to the evidence

Table 8. Metamodel relationships and cardinalities

From Class	To Class	Rel.	Card.	Meaning
RegulatoryClause	SourceDocument	isIn	1..1	Each RegulatoryClause is contained in exactly one SourceDocument
RegulatoryClause	Provenance	prov	1..*	Each RegulatoryClause has one or more Provenance records
SourceDocument	Organization	publishedBy	1..1	Each SourceDocument is published by exactly one Organization
Requirement	Provenance	from	1..*	Each Requirement has one or more Provenance records
Requirement	Constraint	derivedFrom	0..*	A Constraint is derived from zero or more Requirements
Requirement	ProductType	appliesTo	1..*	Each Constraint applies to one or more ProductTypes
Requirement	Property	appliesTo	1..*	A Requirement applies to one or more Properties
Provenance	Agent	createdBy	1..1	Each Provenance record is created by an Agent
RegulatoryClause	Requirement	defines	1..*	Each RegulatoryClause defines one or more Requirements
SpecificationSection	Provenance	from	1..*	Each SpecificationSection has one or more Provenance records
SpecificationSection	ProductType	specifies	1..*	Each SpecificationSection specifies one or more ProductTypes
Product	Organization	manufacturedBy	1..1	Each Product is manufactured by exactly one Organization
Product	ProductType	refersTo	0..*	A Product may refer to zero or more ProductTypes

Product	Classification	classified As	0..*	A Product may be classified under zero or more Classifications
Product	Property	has	0..*	A Product has zero to many Properties
ProductType	Classification	classified As	0..*	A ProductType may be classified under zero or more Classifications
Property	ReferenceStandard	isDefined By	1..*	Each Property is defined by one or more ReferenceStandards
Property	Unit	has	0..1	Each Property has zero or one Unit
Property	PropertyValue	has	0..1	Each Property has zero or one PropertyValue
PropertyValue	Unit	has	0..1	Each PropertyValue has zero or one Unit
Property	TestMethod	has	0..*	Each Property has zero to many TestMethods
PropertyValue	Evidence	supported By	0..*	A PropertyValue may be supported by zero or more Evidence records
TestMethod	ReferenceStandard	compliesTo	1..*	A TestMethod complies to one or many ReferenceStandards

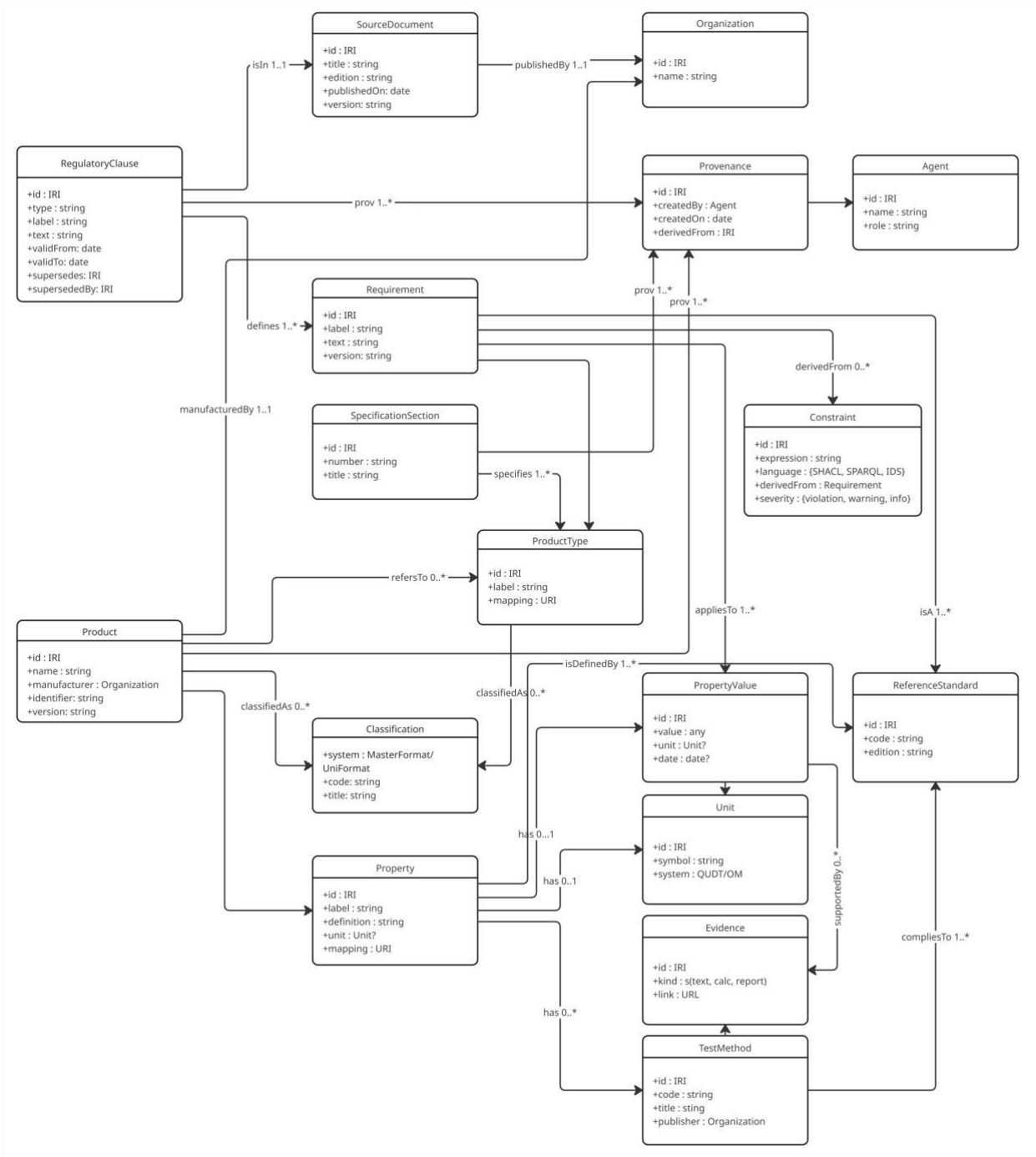

Figure 3. Meta Model

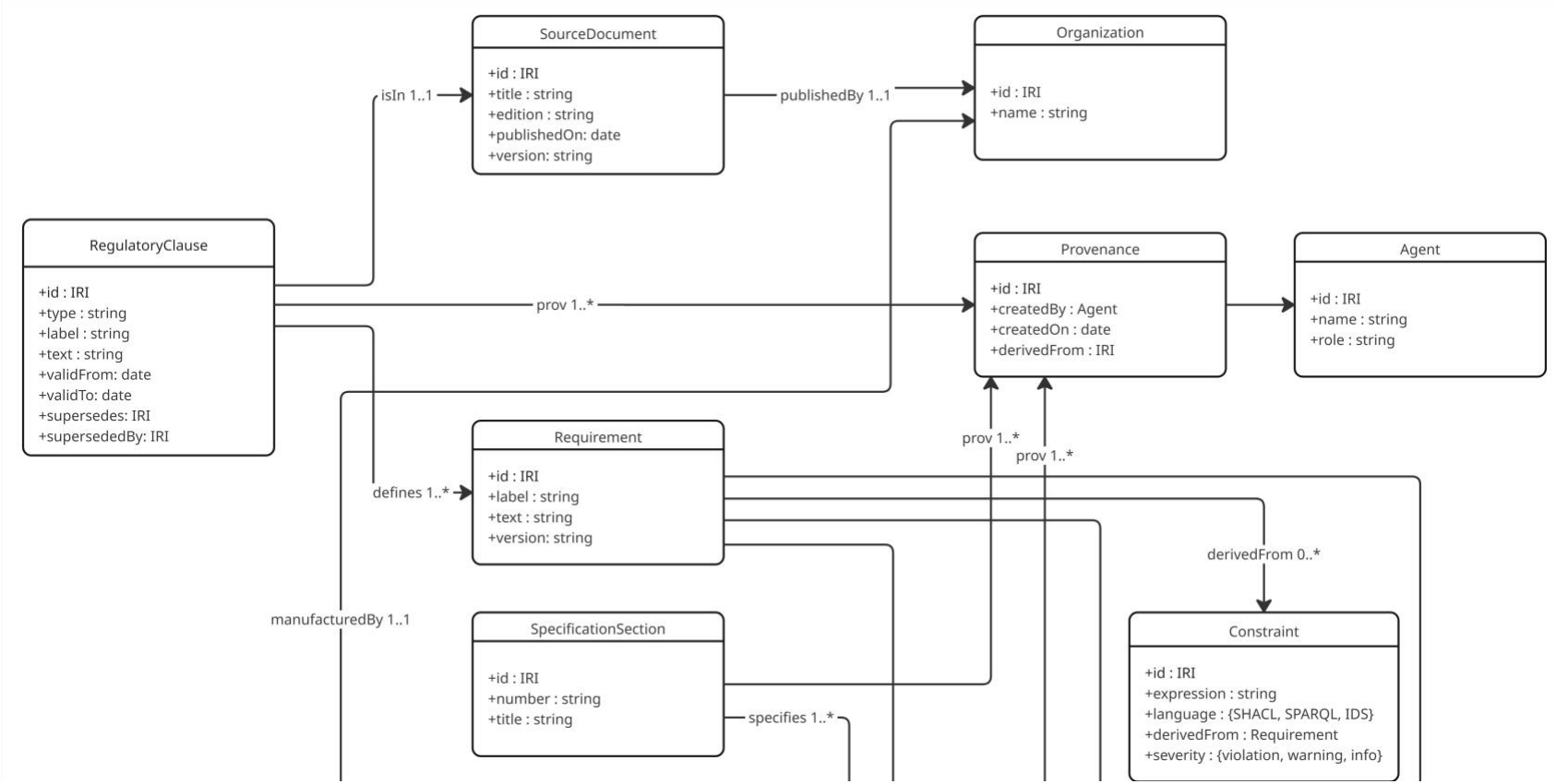

Figure 4. Meta Model (part 1)

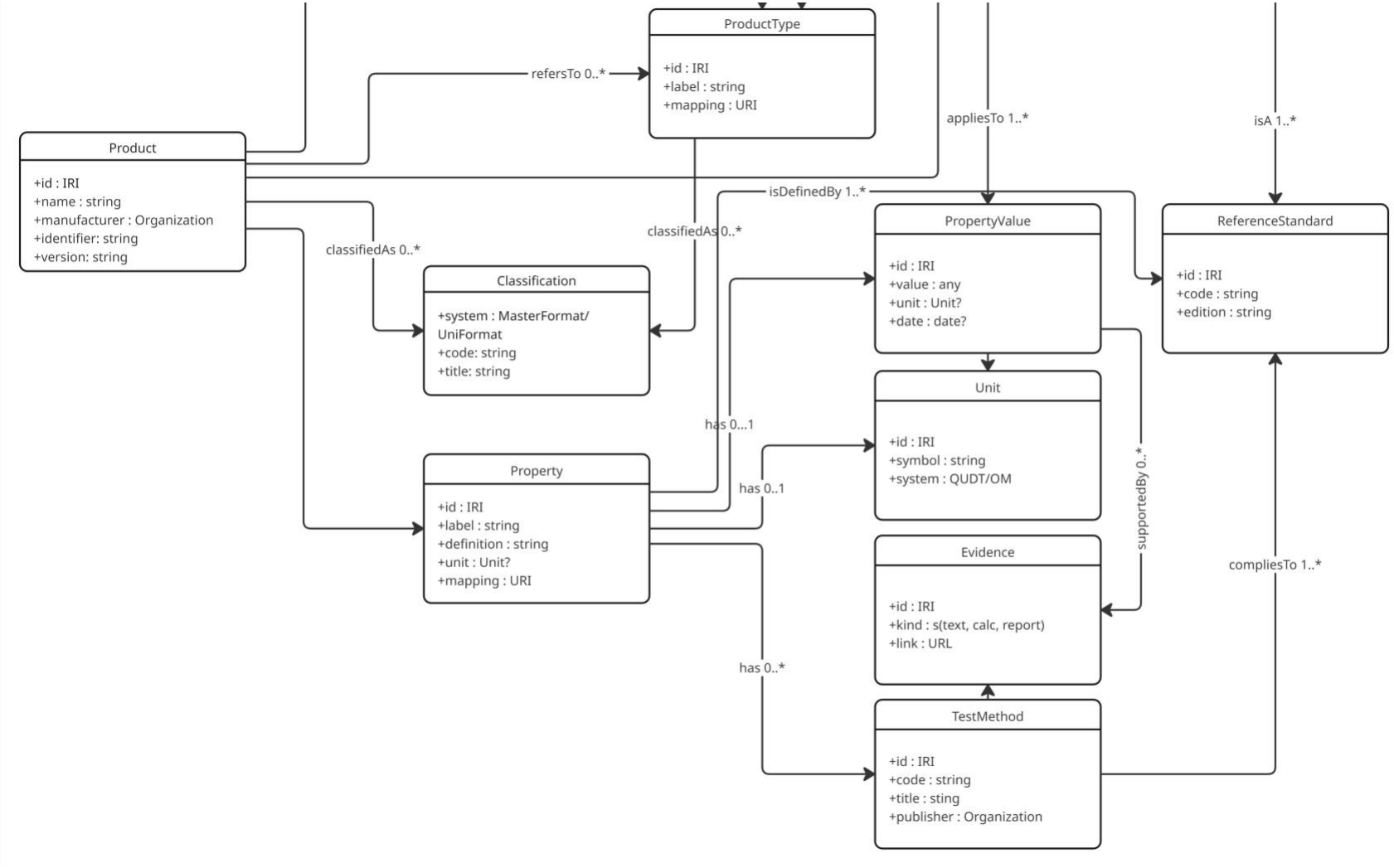

Figure 5. Meta Model (part 2)

4.2. NMCC Data Model

The NMCC data model is defined to support the use cases and the stakeholders deploying them as well as the activities undertaken within these use cases.

Use cases

The use-case diagram shown in Figure 6 captures how stakeholders interact with the NMCC within a single system boundary in the DCP ("NMCC Portal & APIs"). Actors include Architects & Engineers, Builders, Building Owners, Operators, Environmental Groups, Manufacturers & Business Interests, Government Regulatory Bodies, and Third-Party Developers. Each actor connects to a specific set of platform functions, emphasizing compliance, permitting, licensing, governance, and developer enablement. Use-cases shown in Figure 6 include:

- **Code access as a common dependency**: *Verify Design Compliance* and *Access Codes During Construction* both *include Access NMCC Codes & Standards*, highlighting that consultation of codified requirements underpins both design and execution phases.
- **Administrative outcomes depend on submissions**: *Receive Permit Decision extends Submit Building Permit Application*; likewise, *Receive Operating License extends Request Operating License*, clarifying that outcomes occur only when prerequisite actions are completed.
- **Policy input influences code adoption**: *Submit Environmental Input / Participate in Consultation extends Adopt & Modify NMCC for Jurisdiction*, indicating that environmental feedback informs regulatory adaptation.
- **Developer ecosystem relies on open interfaces**: *Develop Code Checkers & Compliance Systems* includes *Access NMCC APIs & Published Specifications*, showing that external tooling is enabled by published interfaces and specifications.
- **Regulatory enforcement and product declarations**: Regulators use *Enforce Compliance*, while manufacturers declare conformity via *Declare Product Compliance / Provide Product Data*, completing the compliance loop from requirement to market readiness.

In support of these use cases, digitalization of the NMCC could help answer queries of the following type:

- What are all the requirements in NBC Division B, Part 9 related to fire separations?
- Which sentences in Article 9.10.9.6 contain mandatory ("shall") clauses?
- What properties must a product type satisfy to meet a given requirement?
- Which reference standards are cited by clauses in Section 9.10?
- What is the definition of a specific term used in the code?
- What was the text of this clause in NBC 2015 vs NBC 2020?

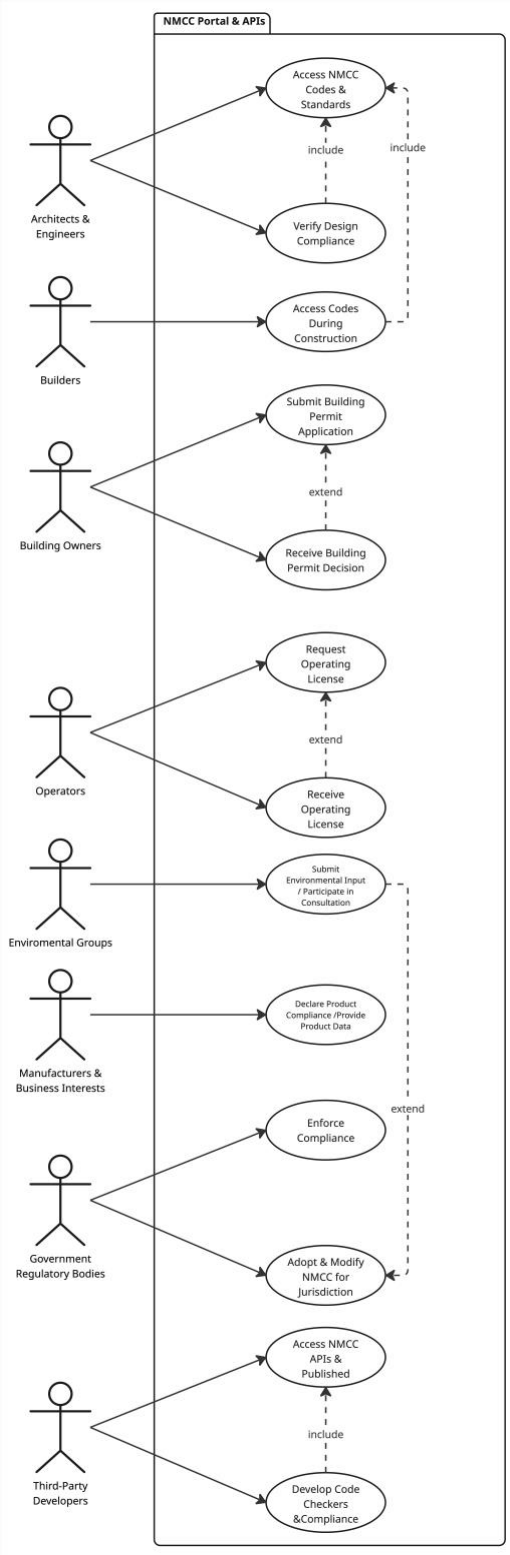

Figure 6. NMCC Use-Case diagram

Activity diagrams

Figure 7, Figure 8, and Figure 9 depict the end-to-end workflow from content preparation to compliance decisions and feedback within a single platform flow.

Main flow (top-to-bottom):

1. **Content & versioning.** NMCC clauses are digitized into machine-interpretable structures and placed under version control and archiving.
2. **Rule & Information Requirements (IR) services.** Current specifications and regulations are configured as executable rule sets; Information Requirements and BIM modelling guidelines are defined and maintained.
3. **Provincial/regional adaptation.** Authorities adapt NMCC for their jurisdiction and publish the adapted code back to the DCP.
4. **Land-use data preparation.** Land-use plans are converted to OGC-compliant formats and published for downstream use.
5. **Web/APIs & integration.** Digital code management, rule configuration and IR services are exposed via secure APIs and integrated with existing systems to minimize disruption.
6. **Project & regulatory environment.** A project thread is created in an openCDE; IFC models and supporting documents are submitted for review (permit, land-use, and urban).
7. **Automated compliance checking.** Three services run in parallel:
8. **Decision & feedback loop.**
    - **Compliant:** proceed with regulatory decisions and planning; results are recorded.
    - **Not compliant:** BCF feedback is delivered via the openCDE; the design/model/data are revised and resubmitted, closing the loop.

Platform-wide capabilities (bottom fork):

- **Security & access control:** RBAC, encryption, audit and policy enforcement.
- **Data standardization & exchange:** openCDE API connections for interoperable data flows across CDEs.
- **Training & support:** guides, onboarding, helpdesk to drive adoption.
- **Centralized reporting & analytics:** usage metrics, compliance tracking, KPIs and planning.

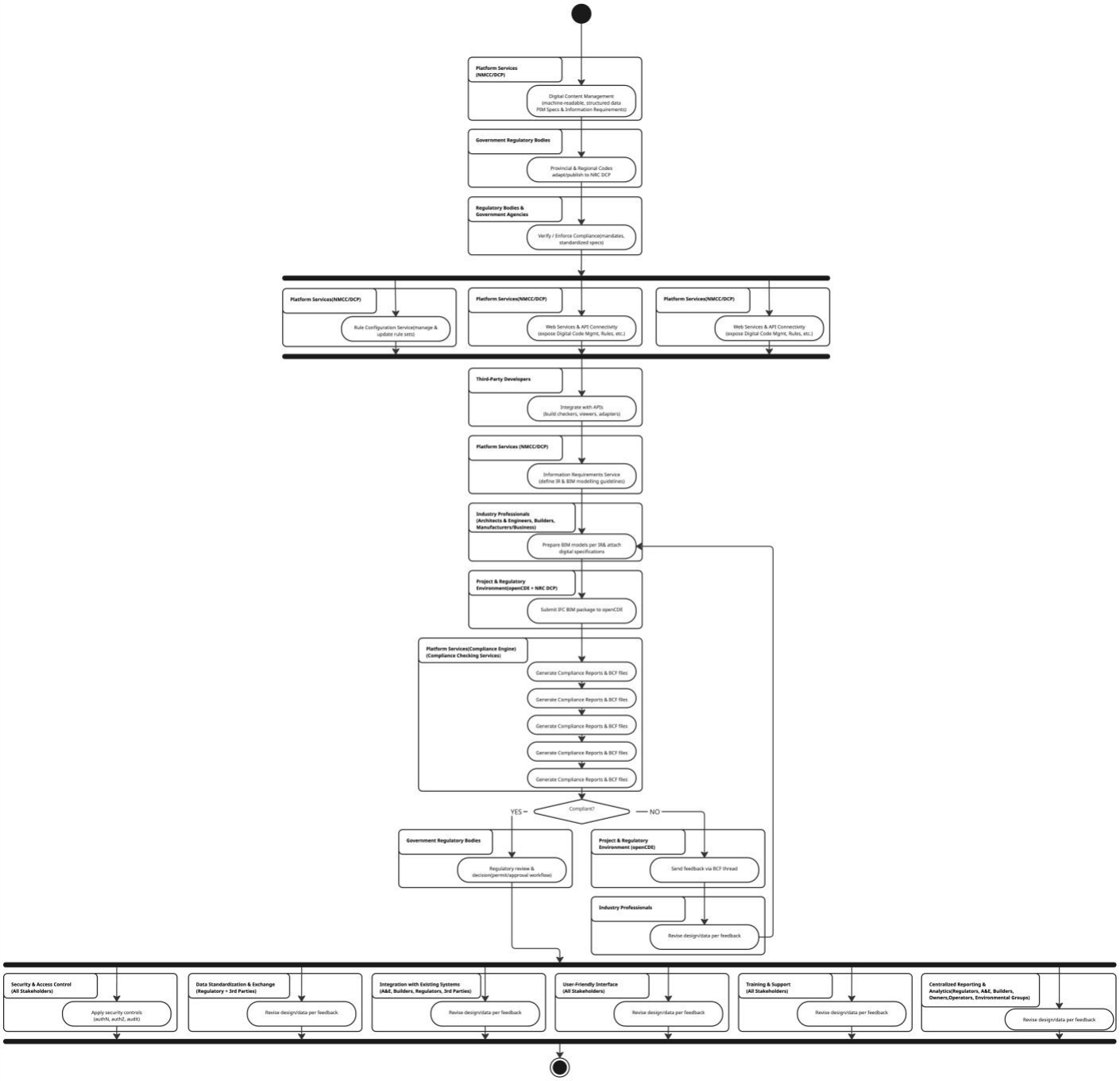

Figure 7. NMCC Activity Diagram

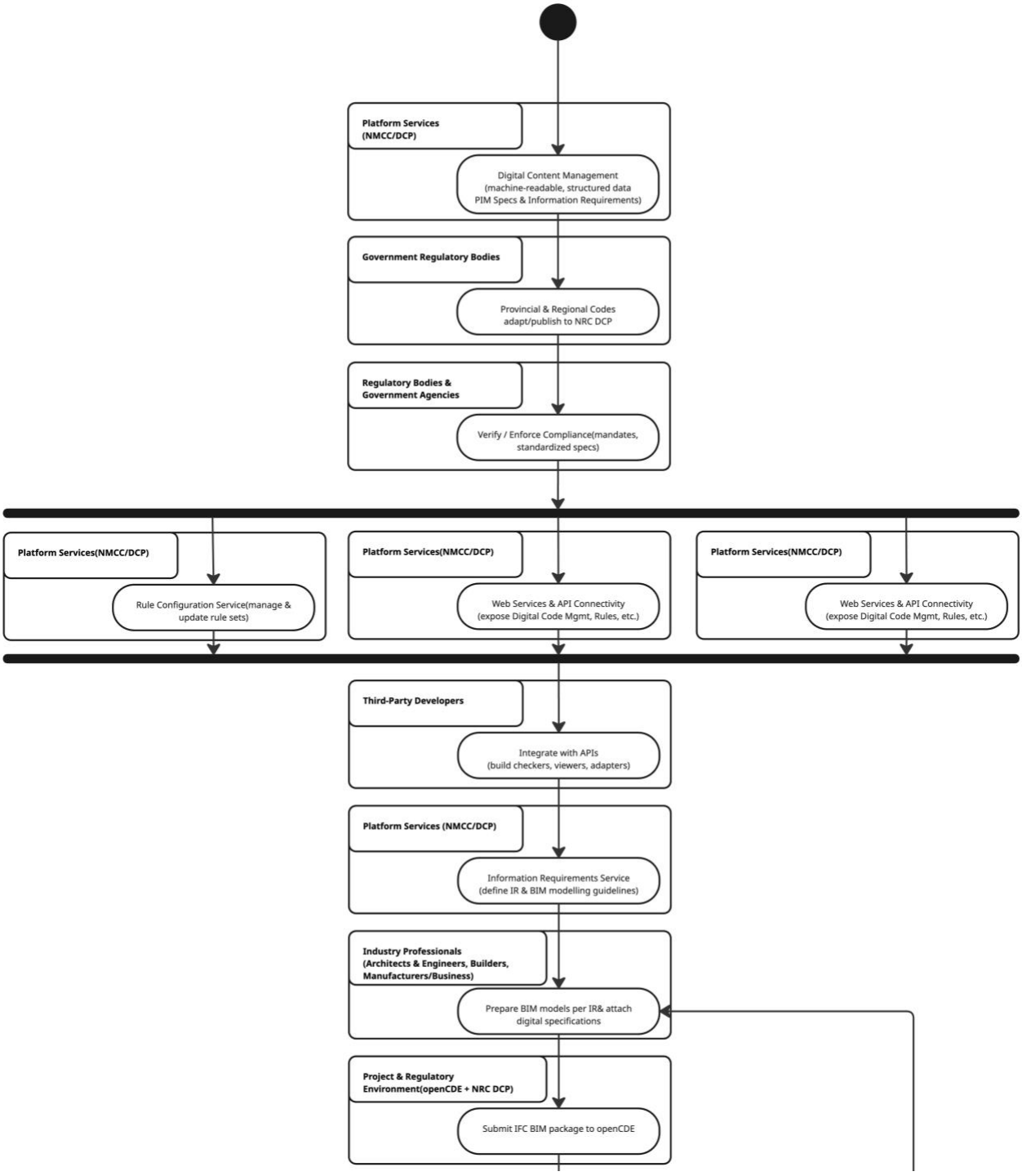

Figure 8. NMCC Activity Diagram (part 1)

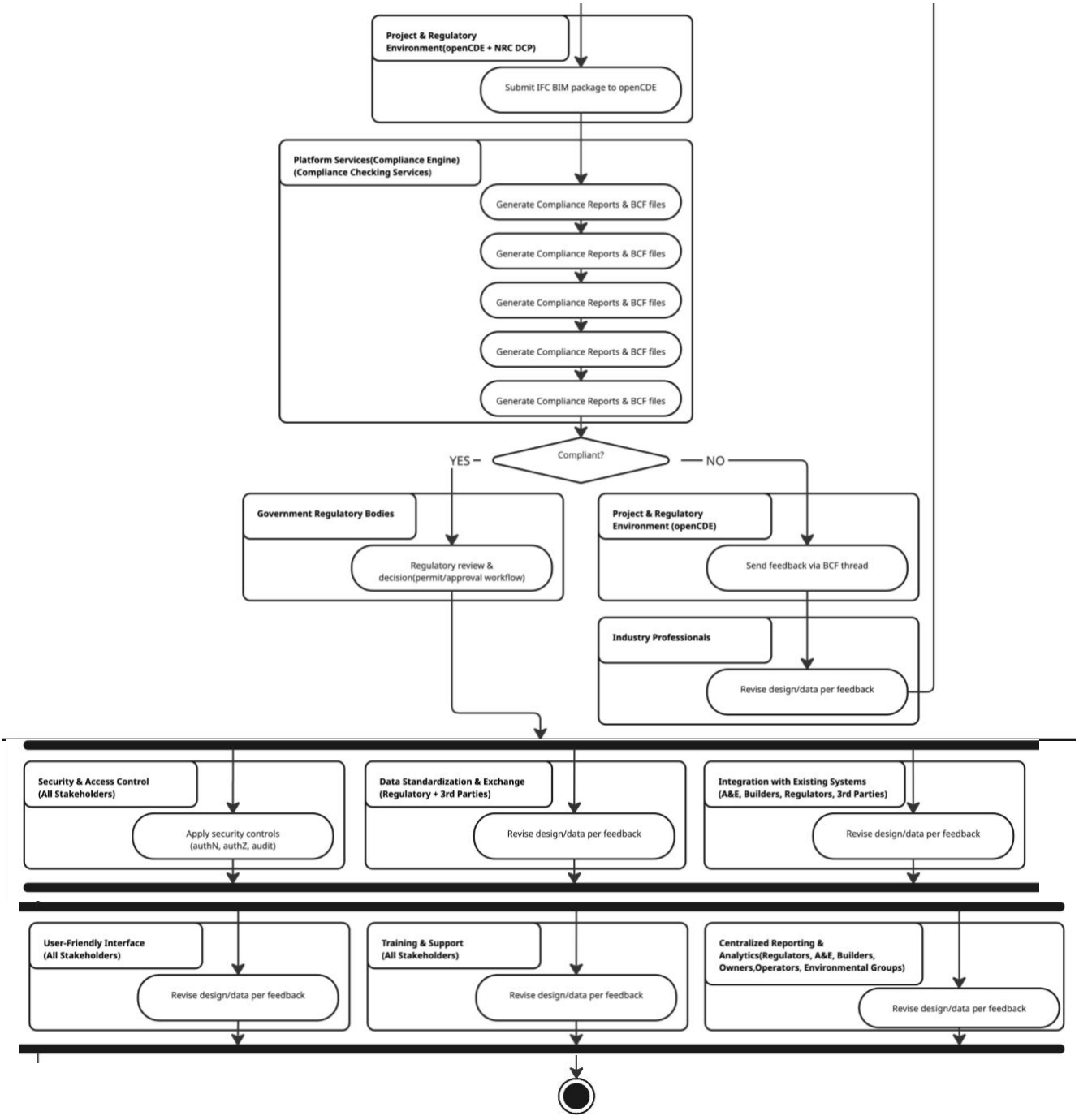

Figure 9. NMCC Activity Diagram (part 1)

Data model

Table 9 describes the classes within the NMCC data model whereas
Table 10 describes its attributes and Table 11 describes the relationships and cardinalities between classes. Figure 10 illustrates the data model for the NMCC. The relationships are as follows:

- **Code hierarchy.** CodeVolume (NBC/NFC/NPC/NECB) contains Division (A/B/C) → Part → Section → Subsection → Article → Sentence.
- **From sentences to rules.** Each Sentence may contain one or more RegulatoryClause (typed as *shall/may/shall_not*) and optional Subclause elements. A RegulatoryClause can *define* Requirement(s) that apply to ProductType(s) and constrain specific Property(ies).
- **From requirements to constraints.** A Requirement *appliesTo* one or more ProductType(s). A Constraint is *derived from* a Requirement and *constrains* one or more Property(ies), enabling automated validation of product data against code-specified thresholds expressed as PropertyValues.
- **Requirements.** A RegulatoryClause can define Requirement(s) that apply to ProductType(s) and constrain specific Property(ies).
- **Properties, values & units.** A Property represents a measurable characteristic (e.g., fire-resistance rating). A Property may have a PropertyValue that captures a specific threshold or limit. A PropertyValue may optionally reference a default Unit for consistent interpretation.
- **References & terminology.** Clauses can reference external ReferenceStandard(s) and reuse controlled DefinedTerm(s).
- **Provenance.** Every clause has mandatory Provenance (source = Code or CCMC), enabling traceability across editions.

Table 9. NMCC data model Classes

Class Name	Purpose	Role in the Data Model
CodeVolume	Represents a complete code volume (NBC, NFC, NPC, NECB)	The top-level container for an entire code publication. Each CodeVolume represents a specific edition (e.g., NBC 2020) and contains Divisions. This class enables queries scoped to a particular code and edition, supporting cross-edition comparisons and historical lookups.
Division	Represents a division within the code (A, B, or C)	Organizes code content into its three main divisions: Division A (Compliance, Objectives, Functional Statements), Division B (Acceptable Solutions), and Division C (Administrative Provisions). Each Division belongs to one CodeVolume and contains Parts.

Part	Represents a part within the code structure	A major organizational unit within a Division (e.g., Part 9 – Housing and Small Buildings, Part 3 – Fire Protection). Parts group related Sections and provide high-level subject matter organization.
Section	Represents a section within the code structure	A thematic grouping within a Part (e.g., Section 9.10 – Fire Protection). Sections contain Subsections and provide the primary navigation structure for code users.
Subsection	Represents a subsection within a section	A finer subdivision within a Section (e.g., Subsection 9.10.9 – Fire Separations in Houses). Subsections group related Articles addressing specific subtopics.
Article	Represents an article within the code structure	The numbered provision level within a Subsection (e.g., Article 9.10.9.6). Articles contain one or more RegulatoryClause(s) and represent discrete, citable regulatory provisions.
Sentence	Represents an individual sentence within regulatory text	The atomic unit of regulatory text within an Article. Each Sentence is numbered (e.g., Sentence (1), (2)) and may contain one or more RegulatoryClause elements. Sentences are the finest granularity at which requirements are expressed.
RegulatoryClause	Represents a regulatory clause with enforcement type	A semantically tagged portion of a Sentence classified by its enforcement type: shall (mandatory), may (permissive), or shall_not (prohibitive). This classification enables automated filtering of mandatory vs. optional requirements and supports rule extraction for compliance checking.
Subclause	Represents a subclause within a clause	Captures nested structure within a RegulatoryClause (e.g., lettered subclauses (a), (b), (c) or roman numerals (i), (ii)). Preserves the hierarchical organization of complex requirements.
Requirement	Represents a specific requirement derived from the code	The normalized expression of what must be achieved. A Requirement is defined by one or more RegulatoryClause(s) and specifies the conditions, thresholds, or properties that must be satisfied. Requirements bridge the gap between legal text and machine-interpretable rules.
Constraint	Represents a machine-executable check derived from a Requirement	The formal, executable expression of a Requirement in a validation language (e.g., SHACL, SPARQL, IDS). While a Requirement expresses human-authored intent, a Constraint translates that intent into computable logic for automated compliance checking. Each Constraint links back to its source Requirement(s) and constrains one or more Properties.

ProductType	Represents a type or category of product	An abstract classification of products referenced in code requirements (e.g., "Type X Gypsum Board," "Fire Door Assembly"). Requirements apply to ProductTypes, enabling generic compliance rules. The optional mapping attribute supports alignment with IFC entities or bSDD classifications.
Property	Represents a measurable or definable characteristic	A named attribute that can be measured, specified, or constrained (e.g., fire-resistance rating, thickness, flame-spread rating). Properties are referenced by Constraints to specify what must be validated. Each Property can be mapped to external dictionaries (bSDD, IFC) for interoperability.
PropertyValue	Represents a specific value for a property	Captures the actual value of a Property in a specific context (e.g., "fire-resistance rating = 1 hour", "thickness ≥ 12.7 mm"). In the NMCC context, PropertyValues represent the thresholds or limits specified in code requirements. Each PropertyValue may include an optional Unit and date. This class enables Constraints to validate measured product data against code-specified thresholds.
Unit	Represents a unit of measurement	Standardized units (e.g., hours, mm, kPa) aligned with QUDT or OM ontologies. Units ensure consistent interpretation of numeric values in requirements and property definitions across systems.
ReferenceStandard	Represents a reference standard cited by the code	External standards (e.g., CAN/ULC-S101, CSA A82.27) that define test methods, material specifications, or performance criteria. The NMCC references hundreds of standards; this class enables tracking which clauses cite which standards and supports change impact analysis when standards are updated.
DefinedTerm	Represents a term with its official definition	Captures the controlled vocabulary defined in Division A of the code. Each DefinedTerm has a label and definition, ensuring consistent interpretation of terminology across clauses. Linking clauses to DefinedTerms supports semantic disambiguation and glossary generation.
Provenance	Tracks origin and creation information for entities	Provides audit trail and lineage for any entity in the model. Records who created or modified data, when, and from what source (e.g., Code edition, CCMC report). Essential for trust, accountability, and tracking changes across code editions.

Table 10. NMCC data model attributes

Class	Attribute	Data Type	Description
Code Volume	id	IRI	Unique identifier for the code volume
	name	{NBC, NFC, NPC, NECB}	Name of the code volume (NBC, NFC, NPC, or NECB)
	edition	string	Edition of the code volume
	year	int	Year of the code volume
Division	id	IRI	Unique identifier for the division
	name	A/B/C	Name of the division (A, B, or C)
	title	string	Title of the division
Part	id	IRI	Unique identifier for the part
	number	string	Number of the part
	title	string	Title of the part
Section	id	IRI	Unique identifier for the section
	number	string	Number of the section
	title	string	Title of the section
Subsection	id	IRI	Unique identifier for the subsection
	number	string	Number of the subsection
	title	string	Title of the subsection
Article	id	IRI	Unique identifier for the article
	number	string	Number of the article
	title	string	Title of the article
Sentence	id	IRI	Unique identifier for the sentence
	number	string	Number of the sentence
	text	string	Text content of the sentence
Regulatory Clause	id	IRI	Unique identifier for the regulatory clause
	type	string	Type of regulatory clause
	label	string	Label of the regulatory clause
	text	string	Text content of the regulatory clause
	validFrom	date	Date from which this version is valid
	validTo	date	Date until which this version is valid (null if current)
	supersedes	IRI	Reference to the previous version of this clause
	supersededBy	IRI	Reference to the newer version (if deprecated)
Subclause	id	IRI	Unique identifier for the subclause
	label	string	Label of the subclause
	text	string	Text content of the subclause
	id	IRI	Unique identifier for the requirement

Requirement	label	string	Label of the requirement
	text	string	Text content of the requirement
Constraint	id	IRI	Unique identifier for the constraint
	expression	string	Formal expression of the constraint (e.g., SHACL shape, SPARQL query)
	language	{SHACL, SPARQL, IDS}	Language used to express the constraint
	derivedFrom	Requirement	The requirement from which this constraint is derived
	severity	{violation, warning, info}	Severity level if constraint is not satisfied
Product Type	id	IRI	Unique identifier for the product type
	label	string	Label of the product type
	mapping	IRI (?)	mapping for the product type (to IFC or bSDD) (optional)
Property	id	IRI	Unique identifier for the property
	label	string	Label of the property
	definition	string	Definition of the property
	unitKind	Unit?	Kind of unit for the property (optional)
	mapping	URI	URI mapping for the property
PropertyValue	id	IRI	Unique identifier for the property value
	value	any	Value of the property
	unit	Unit?	Unit of the property value (optional)
	date	date?	Date of the property value (optional)
Reference Standard	id	IRI	Unique identifier for the reference standard
	code	string	Code of the reference standard
	edition	string	Edition of the reference standard
Defined Term	id	IRI	Unique identifier for the defined term
	label	string	Label of the defined term
	definition	string	Definition of the term
Unit	id	IRI	Unique identifier for the unit
	symbol	string	Symbol representing the unit
	system	QUDT/OM	Unit system (QUDT or OM)
Provenance	id	IRI	Unique identifier for provenance information
	createdBy	Agent	Agent who created the entity
	createdOn	date	Date when the entity was created
	derivedFrom	IRI	Source from which the entity was derived

Table 11. NMCC Data Model relationships and cardinalities

From Class	To Class	Rel.	Card.	Meaning
CodeVolume	Division	has	1..1	Each CodeVolume has one Division.
Division	Part	has	1..*	Each Division has one or more Parts.
Part	Section	has	1..*	Each Part has one or more Sections.
Section	Subsection	has	1..*	Each Section has one or more SubSections.
Subsection	Article	has	1..*	Each SubSection has one or more Articles.
Article	Sentence	has	1..*	Each Article has one or more Sentences.
Sentence	RegulatoryClause	has	0..*	Each Sentence has zero or more RegulatoryClause.
RegulatoryClause	SubClause	has	0..*	Each RegulatoryClause has zero or more SubClauses.
RegulatoryClause	Requirement	defines	0..*	Each RegulatoryClause defines zero or more Requirements
RegulatoryClause	Constraint	derives	0..*	A Constraint is derived from zero or more Requirements
RegulatoryClause	ReferenceStandard	refersTo	0..*	Each RegulatoryClause refers to zero or more ReferenceStandard.
RegulatoryClause	DefinedTerm	uses	0..*	Each RegulatoryClause uses zero or more DefinedTerm.
RegulatoryClause	Provenance	prov	1..*	Each RegulatoryClause has one or more Provenance records
Requirement	ProductType	appliesTo	1..*	A Requirement applies to one or more ProductTypes
Requirement	Property	constrains	1..*	A Requirement constrains one or more Properties
Property	PropertyValue	has	0..1	A property has zero or one PropertyValue.
PropertyValue	DefaultUnit	has	0..1	A PropertyValue has zero or one DefaultUnit.

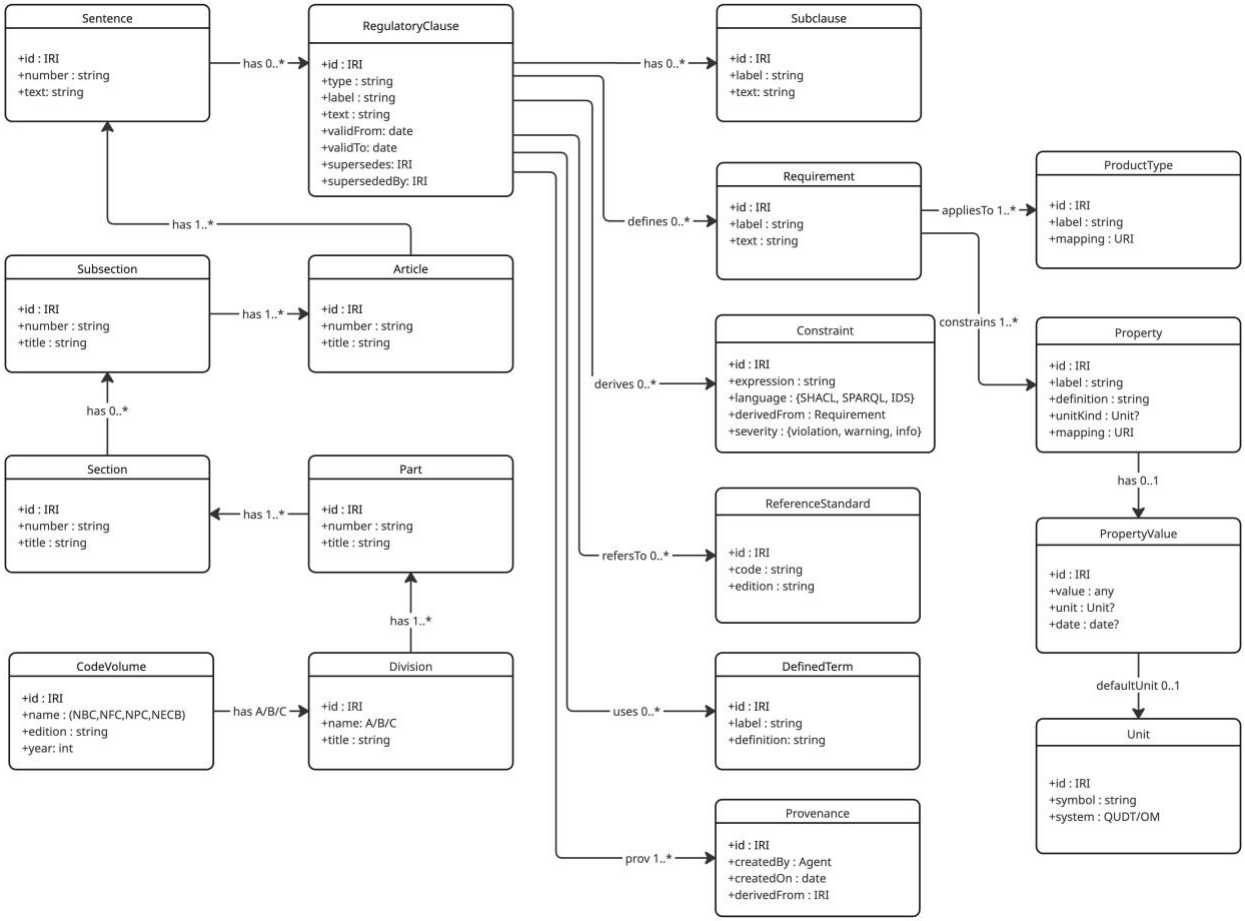

Figure 10. NMCC Data Model

4.3. NMS Data Model

The NMS data model is defined to support the use cases and the stakeholders deploying them as well as the activities undertaken within these use cases.

Use cases

Figure 11 shows how stakeholder groups interact within a single system boundary in the DCP ("NMS Portal & APIs"). Actors and their use cases include:

- **NRC teams (NMS Technical, Content Developers, CDE Admins, Platform /IT /Security /Privacy /Perf /Change):** Drive *Digital Content Integration*, the *Content Development Process*, *Content Management*, and the *Content Publishing Workflow* (which includes the *Standardized CDE Structure*). They also run *Data Security & Privacy*, *Scalability & Performance*, *Monitoring & Analytics* (which extends the *Feedback Mechanism*), and *Change Management* (includes Content Development and Publishing Workflow).
- **Government Agencies (Regulatory Bodies, National Advisory Board):** Participate in *Compliance Checking & Rule Application* and *Integration with NRC DCP*; collaborate via *Collaboration with Other Government Agencies*.
- **Specification Writers & Architects/Engineers:** Use *User Interfaces (Web Services)* and *Independent Web Services (APIs)*, access *Resources & Template Availability*, the *Latest NMS Content (Version Control)*, contribute through the *Feedback Mechanism*, customize via *Customization of NMS for Projects*, and participate in *Compliance Checking & Rule Application*.
- **Construction Specialists & Contractors (incl. Project Managers):** Access *User Interfaces* and *APIs*, retrieve *Latest NMS Content*, provide *Feedback*, perform *Compliance Checking & Rule Application*, and *Customize NMS for Projects* (PMs also engage in *Monitoring & Analytics*).
- **Government & Private Sector (Organizations, Third-Party Vendors:** Consume *User Interfaces* and *APIs*, access *Latest NMS Content*, provide *Feedback*; vendors also participate in *Digital Specifications Services Integration* (**includes** *APIs*).

The key relationships illustrated in Figure 11 are:

- Content Publishing Workflow includes Standardized CDE Structure;
- Integration with NRC DCP includes Publishing Workflow;
- Curated Content Management includes Content Development;
- Digital Specifications Services Integration and User Interfaces both include APIs;
- Customization and Compliance both include Version Control;
- Monitoring & Analytics extends Feedback;
- Change Management includes Content Development and Publishing Workflow.

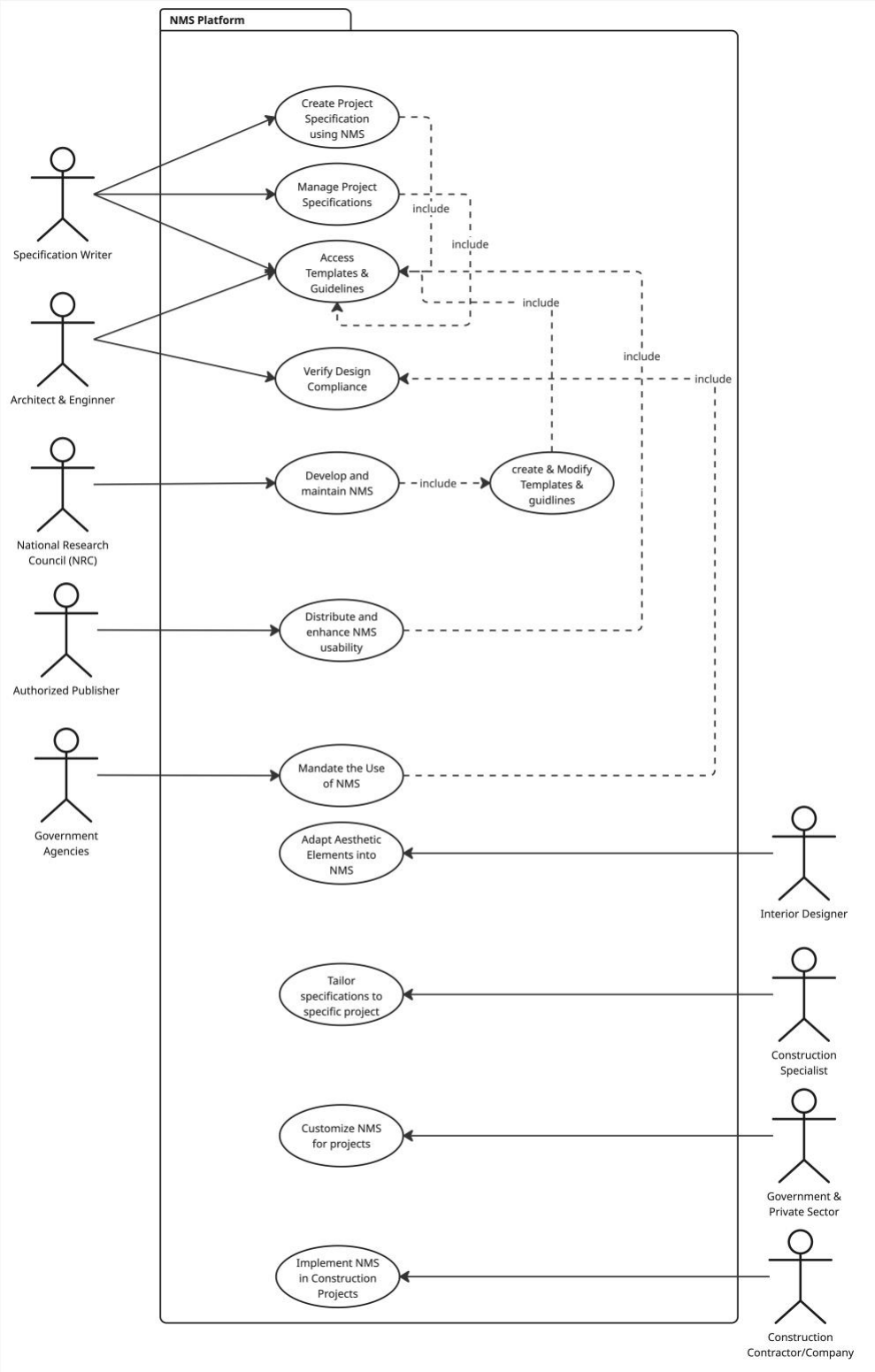

Figure 11. NMS Use-Case Diagram

In support of these use cases, digitalization of the NMS could help answer queries of the following type:

- Which properties must be specified for a given ProductType (e.g., Type X Gypsum Board)?
- Which reference standards are cited in a given specification section?
- What test methods are required to verify a specific QualificationItem?
- What options are available for a given RequirementTemplate?
- What is the definition of a specific term used in the specification?
- What is the revision history and provenance of a specification section?
- Which specification sections reference a particular ReferenceStandard (e.g., ASTM E84)?

Activity diagrams

Figure 12 depicts the end-to-end workflow from content preparation to field use and feedback within a single platform flow.

- **Create & curate.** NRC teams perform Digital Content Integration and the Content Development Process; regulatory bodies provide advisory input. Content then moves through Curated Content Management within a standardized CDE: classify as *Work in Progress*, move to *Shared* for review, decide Ready to Publish? → *Published*; otherwise revise and loop back; archive superseded versions.
- **Expose & integrate.** In parallel, NRC teams integrate compliance checking, push/link content to NRC DCP, and provide User Interfaces (Web) & APIs; Third-party platforms connect via Digital Specifications Services Integration.
- **Use in projects.** Industry professionals' access latest NMS (versioned), use resources & templates, customize for projects, and run compliance checks.
- **Decision & loop.** Regulators verify/enforce compliance. If compliant, contractors implement NMS in projects. If not compliant, industry submits feedback via the portal; NRC teams triage, apply change management, update content/rules/features, and return to curation.
- **Cross-cutting capabilities.** Security & privacy, platform performance, user training & support, and monitoring & analytics operate continuously across the flow.

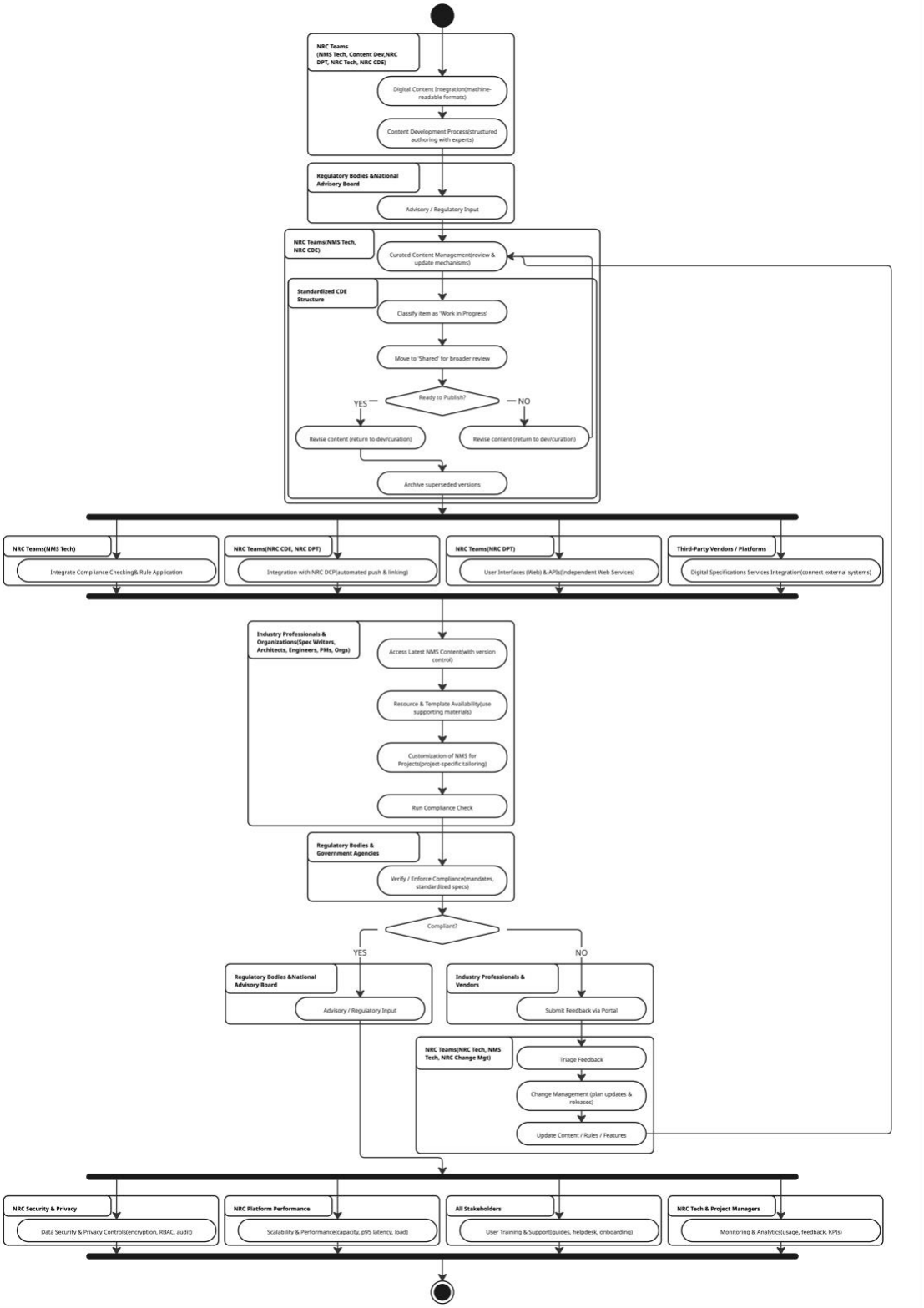

Figure 12. NMS Activity Diagram

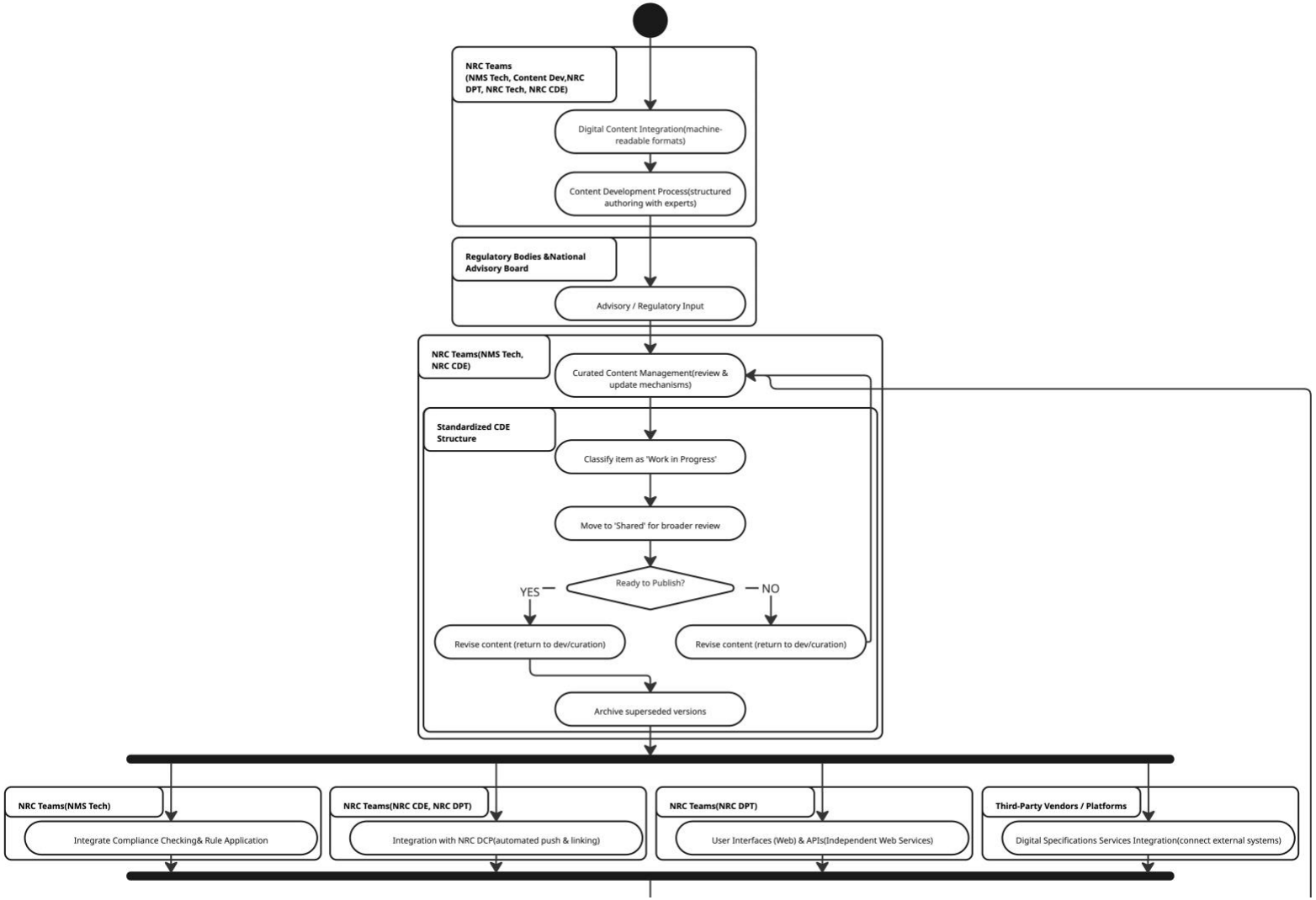

Figure 13. NMS Activity Diagram

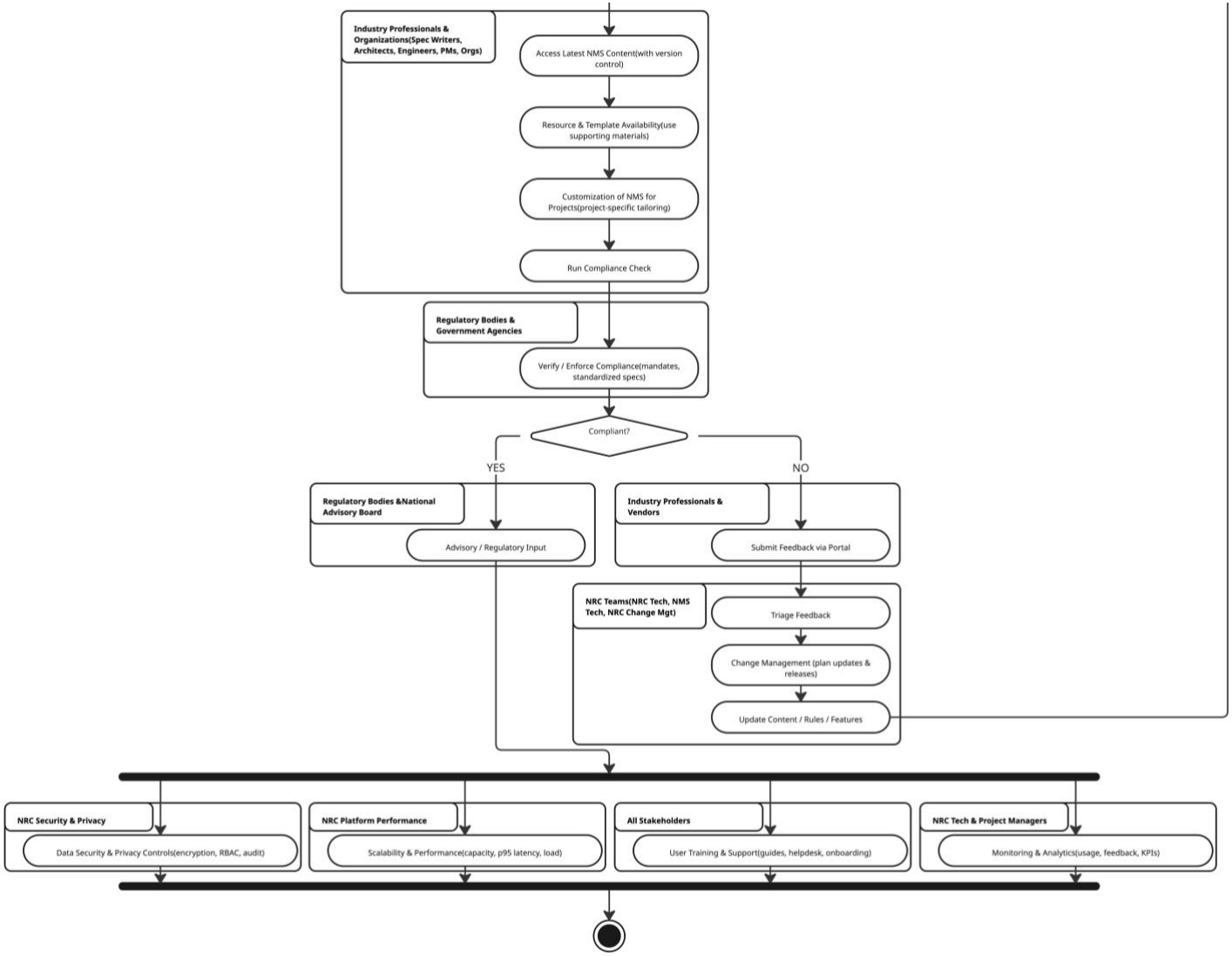

Figure 14. NMS Activity Diagram

Data model

Table 12 describes the classes within the NMS data model whereas Table 13 describes its attributes and Table 14 describes the NMS Data Model relationships and cardinalities. Figure 15 illustrate the data model for the NMS. The model is articulated into three linked zones:

- **Document structure.** Division → SpecificationSection → Article → Paragraph, with Paragraph optionally reusing DefinedTerm.
    - SpecificationSection carries identifiers (id, number, title) plus **edition** and **index (date)**.
- **Requirement package.** A Paragraph can include one or more RequirementTemplate(s) that bundle:
    - QualityCriterion, Submittal, Option, Constraint, Threshold, and Parameter, each linked as shown.
    - Constraint uses a Comparator and has Provenance; Threshold may use a Unit.
    - Parameter captures the role (subject/property/context), and links are provided to the product/property side.
    - RequirementTemplate may reference Classification (MasterFormat/UniFormat).
- **Product, property, and standards.**
    - ProductType has Property(ies); a Property can have a default Unit.
    - QualityCriterion may use a TestMethod and reference a ReferenceStandard; a TestMethod implements a ReferenceStandard and is issued by an Organization; ReferenceStandard is published by an Organization.

Table 12. NMS data model classes

Class Name	Purpose	Role in the Data Model
Division	Represents a division in the specification structure	The top-level organizational container in NMS, corresponding to MasterFormat divisions (e.g., Division 09 – Finishes). Each Division groups related SpecificationSections and provides high-level categorization for specification content.
SpecificationSection	Represents a specification section	A discrete specification unit within a Division (e.g., Section 09 29 00 – Gypsum Board). Each section carries its own number, title, revision, and date, enabling version tracking. Sections contain Articles and serve as the primary deliverable unit for specification writers.
Article	Represents an article within a specification	A major subdivision within a SpecificationSection, typically corresponding to the three-part SectionFormat structure (Part 1 – General, Part 2 – Products, Part 3 – Execution). Articles contain Paragraphs that detail specific requirements.
Paragraph	Represents a paragraph within specification text	The atomic unit of specification content within an Article. Paragraphs contain the actual

		requirement text and may include one or more RequirementTemplate(s) that formalize the specification logic. Paragraphs may also reference DefinedItems for terminology consistency.
DefinedItem	Represents a defined item with its description	Captures controlled vocabulary and terminology used within specifications. Each DefinedItem has a label and definition, ensuring consistent interpretation across sections. Similar to DefinedTerm in NMCC but scoped to specification content.
RequirementTemplate	Represents a template for creating requirements	A structured pattern that bundles the components of a specification requirement: what must be submitted (Submittal), quality criteria to meet (QualificationItem), constraints on properties (Constraint), and available options (Option). Templates enable consistent, machine-interpretable requirement authoring and support reuse across sections.
Classification	Organizes specifications using MasterFormat/UniFormat	Aligns specification content with industry classification systems. MasterFormat organizes by work results (products and activities); UniFormat organizes by building elements (functional systems). This dual classification supports both specification writing and cost estimation workflows.
QualificationItem	Represents qualification criteria for products or contractors	Captures the criteria that must be met to qualify a product, manufacturer, or installer (e.g., "manufacturer with minimum 5 years experience," "product listed by certified testing agency"). Linked to TestMethod and ReferenceStandard for verification.
Submittal	Represents submittal requirements for products/documentation	Specifies what documentation or samples must be provided for review (product data sheets, samples, certificates, test reports). The type attribute distinguishes between productData, samples, and certificates, supporting submittal tracking workflows.
Option	Represents an optional specification element	Captures alternatives or choices within a specification (e.g., "Option A: painted finish, Option B: vinyl-covered"). Options enable specification writers to present bracketed alternatives that project teams can select based on project requirements.

Constraint	Represents a machine-executable check on property values	A formal rule that validates whether a property value meets specification requirements. Each Constraint uses a Comparator (e.g., ≥, ≤, =) and may reference a Threshold. Constraints link back to their source via Provenance, enabling traceability from validation results to specification text.
Threshold	Represents threshold values for constraints	Captures the numeric or categorical limit against which a property value is compared (e.g., "≥ 1 hour", "≤ 25 flame spread index"). May include a range or limit specification and optionally references a Unit for numeric values.
Comparator	Represents comparison operators for evaluating constraints	Defines the logical operator used in a Constraint (e.g., greaterThanOrEqual, lessThan, equals, between). Each Comparator has an operator symbol (≥, ≤, =) and a human-readable label, supporting both machine processing and human readability.
Parameter	Represents parameters for specifications	Captures the role of a referenced entity within a RequirementTemplate. The type attribute indicates whether the parameter represents the subject (object being specified), a property (characteristic being constrained), or context (conditions under which the requirement applies).
Provenance	Tracks origin and creation information	Provides audit trail and lineage for specification content. Records who authored or modified content, when changes were made, and from what source material the content was derived. Essential for managing specification revisions and maintaining editorial accountability.
ProductType	Represents a type or category of product	An abstract classification of products referenced in specifications (e.g., "Type X Gypsum Board," "Acoustic Ceiling Tile"). Specifications apply to ProductTypes rather than specific manufacturer products, enabling competitive bidding. The mapping attribute supports alignment with bSDD or IFC classifications.
Property	Represents a measurable or definable characteristic	A named attribute that can be specified or measured (e.g., fire-resistance rating, thickness, surface burning characteristics). Properties are referenced by Constraints to specify what must be validated. Each Property can be mapped to external dictionaries for interoperability.

PropertyValue	Represents a specific value for a property	Captures the actual value of a Property in a specific context (e.g., "fire-resistance rating = 1 hour", "thickness ≥ 12.7 mm"). In the NMCC context, PropertyValues represent the thresholds or limits specified in code requirements. Each PropertyValue may include an optional Unit and date. This class enables Constraints to validate measured product data against code-specified thresholds.
Unit	Represents a unit of measurement	Standardized units (e.g., hours, mm, kg/m²) aligned with QUDT or OM ontologies. Units ensure consistent interpretation of numeric thresholds and property values across specifications and product data.
TestMethod	Represents a standardized testing method	The procedure used to measure a property or verify compliance (e.g., ASTM E84 for surface burning characteristics). Each TestMethod is issued by an Organization and may implement a ReferenceStandard. Links specification requirements to the verification methods.
ReferenceStandard	Represents a reference standard	External standards (e.g., ASTM, CSA, ULC) that define material specifications, test methods, or performance criteria. NMS references thousands of standards; this class enables tracking which specifications cite which standards and supports change impact analysis.
Organization	Represents an organization	Captures entities that publish standards, issue test methods, or manufacture products (e.g., ASTM International, ULC, CSA Group). Used to establish authority and provenance for referenced standards and test methods.

Table 13. NMS data model attributes

Class	Attribute	Data Type	Description
Division	id	IRI	Unique identifier for the division
	number	int	Number identifying the division
	title	string	Title of the division
Specification Section	id	IRI	Unique identifier for the specification section
	number	string	Number of the specification section
	title	string	Title of the specification section
	revision	string	Revision of the specification section
	date	date	Date of the specification section

Article	id	IRI	Unique identifier for the article
	number	string	Number of the article
	title	string	Title of the article
Paragraph	id	IRI	Unique identifier for the paragraph
	text	string	Text content of the paragraph
DefinedItem	id	IRI	Unique identifier for the defined item
	label	string	Label of the defined item
	definition	string	Definition of the item
Requirement Template	id	IRI	Unique identifier for the requirement template
	name	string	Name of the requirement template
	include	string	Items to include in the requirement
	cavitation	condition	Cavitation condition information
Classification	system	{MasterFormat/ UniFormat}	Classification system (MasterFormat or UniFormat)
	code	string	Code within the classification system
	title	string	Title of the classification
Qualification Item	id	IRI	Unique identifier for the qualification item
	criterion	string	Criterion for the qualification
Submittal	id	IRI	Unique identifier for the submittal
	type	{productData, samples, certificates}	Type of submittal (product data, samples, or certificates)
Option	id	IRI	Unique identifier for the option
	label	string	Label of the option
Constraint	id	IRI	Unique identifier for the constraint
	expression	string	Formal expression of the constraint (e.g., SHACL shape, SPARQL query)
	language	{SHACL, SPARQL, IDS}	Language used to express the constraint
	derivedFrom	Requirement	The requirement from which this constraint is derived
	severity	{violation, warning, info}	Severity level if constraint is not satisfied
Threshold	value	any	Value of the threshold
	limit	limit?	Limit for the threshold (optional)
	range	{lowName}?	Range name for the threshold (optional)
Comparator	id	IRI	Unique identifier for the comparator
	operator	string/enum	The comparison operator symbol or name
	label	string	Human-readable name of the comparator

	operatorSymbol	string?	Mathematical symbol representation (optional)
Parameter	id	IRI	Unique identifier for the parameter
	type	{object/property /context}	Type of parameter (object, property, or context)
Provenance	id	IRI	Unique identifier for provenance information
	createdBy	Agent	Agent who created the entity
	timestamp	date	Timestamp of the provenance
	derivedFrom	IRI	Source from which the entity was derived
ProductType	id	IRI	Unique identifier for the product type
	label	string	Label of the product type
	mapping	URI	Mapping for the product type (to bSDD or other)
Property	id	IRI	Unique identifier for the property
	label	string	Label of the property
	definition	string	Definition of the property
	unitKind	Unit?	Kind of unit for the property (optional)
	mapping	URI	mapping for the property (to bSDD or other)
PropertyValue	id	IRI	Unique identifier for the property value
	value	any	Value of the property
	unit	Unit?	Unit of the property value (optional)
	date	date?	Date of the property value (optional)
Unit	id	IRI	Unique identifier for the unit
	symbol	string	Symbol representing the unit
	system	QUDT/OM	Unit system (QUDT or OM)
TestMethod	id	IRI	Unique identifier for the test method
	code	string	Code of the test method
	title	string	Title of the test method
	date	string	Date of the test method
	organization	Organization	Organization that published the test method
Reference Standard	id	IRI	Unique identifier for the reference standard
	code	string	Code of the reference standard
	edition	string	Edition of the reference standard
Organization	id	IRI	Unique identifier for the organization
	name	string	Name of the organization

Table 14. NMS Data Model relationships and cardinalities

From Class	To Class	Rel.	Card.	Meaning

Division	Specification Section	has	1..*	Each Division contains one or more SpecificationSections.
Specification Section	Classification	refersTo	0..*	Each SpecificationSection refers to zero or more Classification.
Specification Section	ProductType	applyOn	0..*	Each SpecificationSection applies on zero or more ProductType.
Specification Section	Article	has	1..*	Each SpecificationSection contains one or more Articles.
Article	Paragraph	has	1..*	Each Article contains one or more Paragraphs.
Paragraph	DefinedTerm	uses	0..*	A Paragraph may reference zero or more DefinedTerms.
Paragraph	TestMethod	has	0..*	A Paragraph may have zero or more TestMethod.
Paragraph	ReferenceStandard	refersTo	0..*	A Paragraph may refers to zero or more ReferenceStandard.
Paragraph	Requirement Template	includes	0..*	A Paragraph may include zero or more RequirementTemplates.
Requirement Template	QualityCriterion	includes	0..*	A RequirementTemplate may include zero or more QualityCriteria.
Requirement Template	Submittal	requires	0..*	A RequirementTemplate may require zero or more Submittals.
Requirement Template	Option	offers	0..*	A RequirementTemplate may offer zero or more Options.
Requirement Template	Constraint	has	1..*	A RequirementTemplate includes one or more Constraints.
Requirement Template	Provenance	prov	1..*	A RequirementTemplate has one or more Provenance.
Requirement Template	TestMethod	has	0..*	A RequirementTemplate may have zero or more TestMethod.
Requirement Template	ReferenceStandard	refersTo	0..*	A RequirementTemplate may reference zero or more ReferenceStandard.
Constraint	Parameter	involves	1..*	Each Constraint involves one or more Parameter.
Constraint	Comparator	uses	1..1	Each Constraint uses one Comparator.
Constraint	Threshold	has	0..1	A Constraint may have zero or one Threshold.
Parameter	ProductType	subject	0..1	A Parameter may refer to zero or one ProductTypes.
Parameter	Property	refersTo	0..1	A Parameter may refer to zero or one Properties.
Property	PropertyValue	has	0..1	A Property may have zero or one PropertyValue.
PropertyValue	Unit	Default Unit	0..1	A PropertyValue may have zero or one Unit.

TestMethod	ReferenceStandard	implements	0..*	A TestMethos may implement zero or more ReferenceStandard.
ReferenceStandard	Organization	publishedBy	0..1	A ReferenceStandard may be published by zero or one Organization.

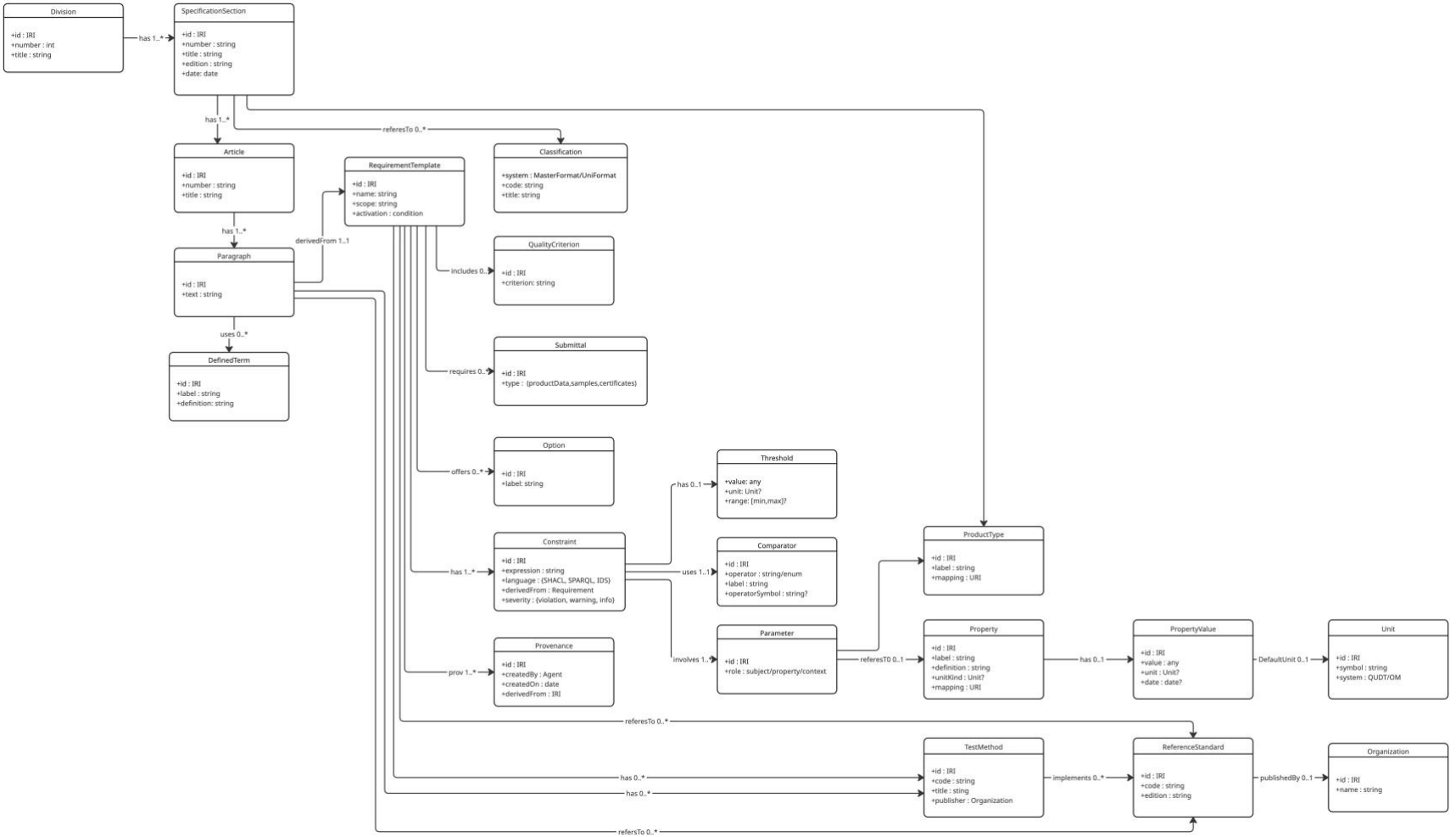

Figure 15. NMS Data model

4.4. CCMC Data Model

The CCMC Material Database data model is defined to support the use cases and the stakeholders deploying them as well as the activities undertaken within these use cases.

Use cases

Figure 16 shows how stakeholders interact within a single system boundary in the DCP ("CCMC Material Database Portal & APIs"). Actors and their use cases include:

- **Regulatory Bodies**: use *Recognize CCMC Evaluations* and *Verify Code Compliance*—both *include* Access CCMC Evaluations.
- **Federal Agencies**: *Approve Products for Public Projects* (*includes* Access CCMC Evaluations).
- **Industry Associations:** *Support & Promote CCMC Evaluations* (*includes* Access CCMC Evaluations).
- **Construction Industry**: *Approve Innovative Products for Projects* (*includes* Access CCMC Evaluations).
- **Architects**: *Access Materials Database* (Validated) (*includes* Access CCMC Evaluations).
- **Engineers:** *Access Product Specs & Performance Data* (*includes* Access CCMC Evaluations).
- **Testing & Inspection Entities**: *Generate Test / Inspection Reports*.
- **Expert & Personnel Entities**: *Provide Technical Analyses & Competency Verification*.
- **Management & Oversight Bodies**: *Perform Audits & System Assessments*.
- **Data Providers**: *Supply Certification Data*.
- **Municipal Bodies**: *Integrate with Municipal E-Permitting Systems*.

In support of these use cases, digitalization of the CCMC could help answer queries of the following type:

- What products have been evaluated by CCMC for a specific ProductType (e.g., structural insulated panels)?
- Does product X have evidence demonstrating compliance with NBC requirement Y?
- What properties have been tested for a specific product, and what were the results?
- Which test methods were used to generate the evidence in an evaluation report?
- What reference standards are cited in a product's evaluation?
- Who is the report holder and manufacturer for a given product?
- Which products are classified under a specific MasterFormat or UniFormat code?

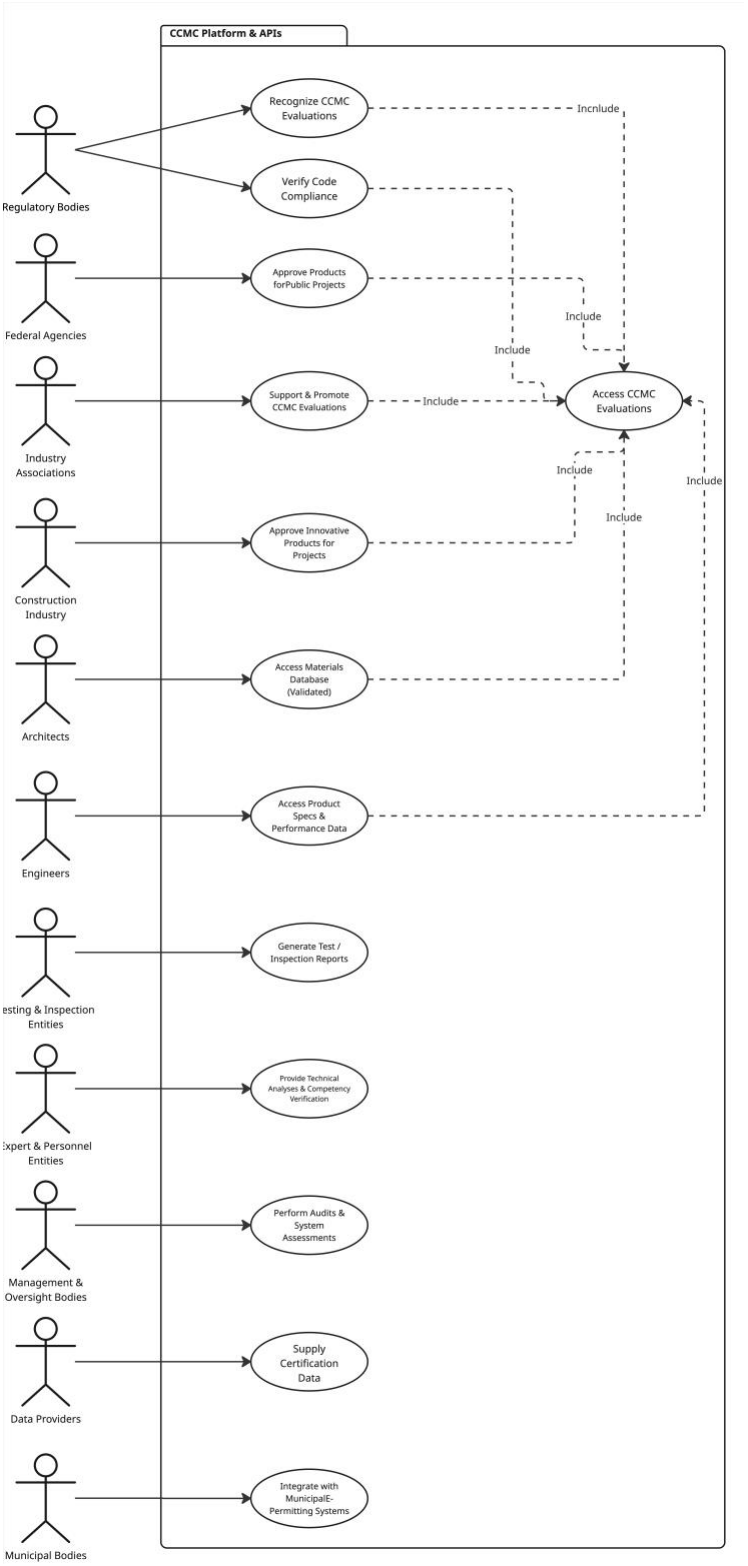

Figure 16. CCMC Use-Case Diagram

Activity diagrams

Figure 17 depicts the end-to-end workflow from content preparation to publication and feedback within a single platform flow:

- **Content integration & linkage.** CCMC/NRC teams integrate CCMC content and cross-link to referenced codes; items move through the CCMC CDE states—Work in Progress → Shared → (Ready to Publish?) → Published—with prior versions archived, then pushed to the NRC DCP.
- **Distribution & access.** The NRC DCP exposes web services/APIs; agencies and developers consume them; industry accesses content in projects; manufacturers submit material applications via the portal.
- **Committee assessment & decision.** Submissions are routed to CCMC committees for assessment. If approved, the DCP publishes evaluations and notifies stakeholders; if not approved, the DCP issues feedback, manufacturers revise, and resubmit.
- **Regulatory publications & interoperability.** CCMC/NRC share regulatory publications and publish via the DCP; collaboration with other CDEs enables interoperability.
- **Platform-wide capabilities.** Security & access control, training & support, version control & archiving, centralized reporting & analytics, and user-interface design run continuously across the workflow.

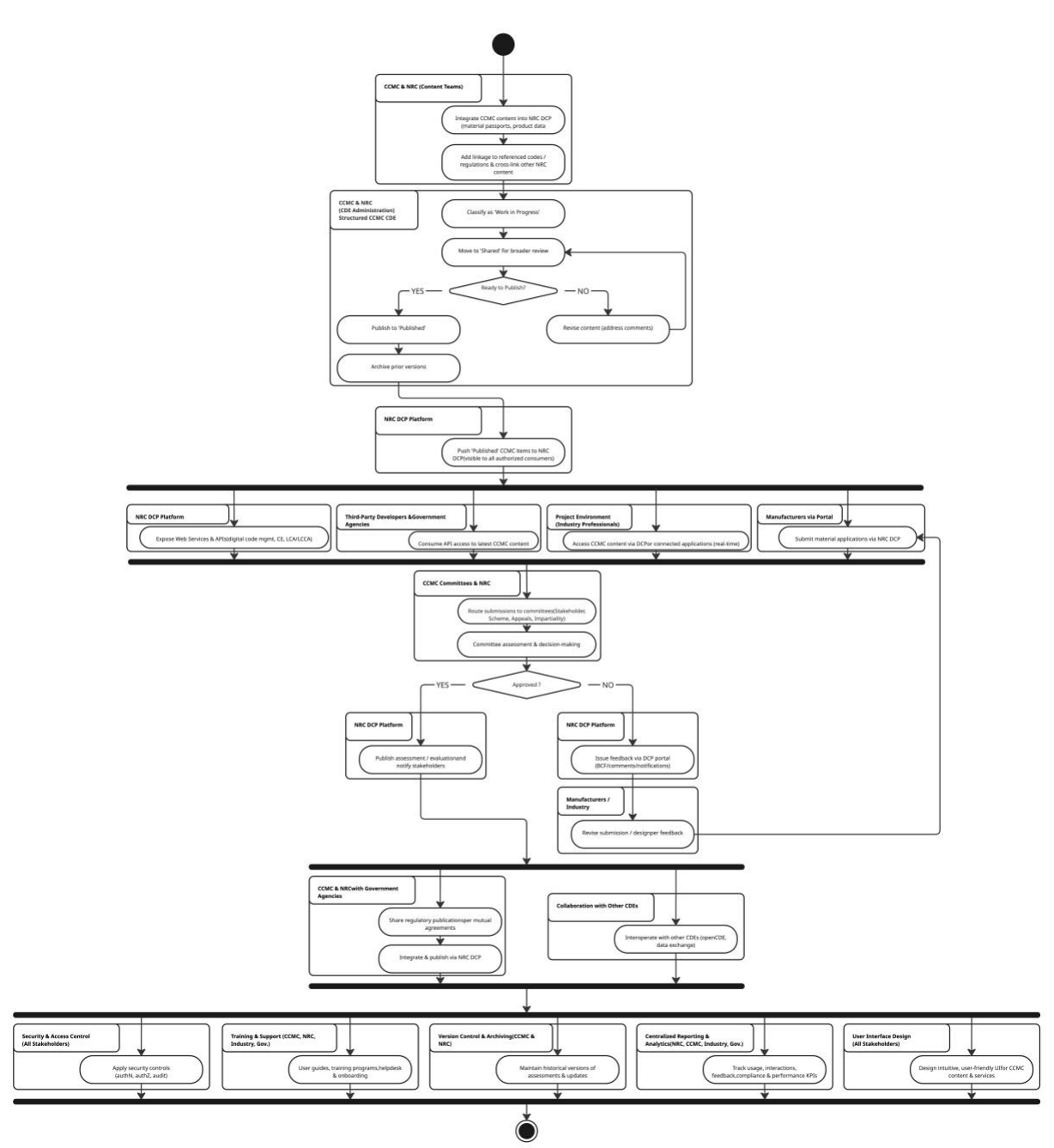

Figure 17. CCMC Activity Diagram

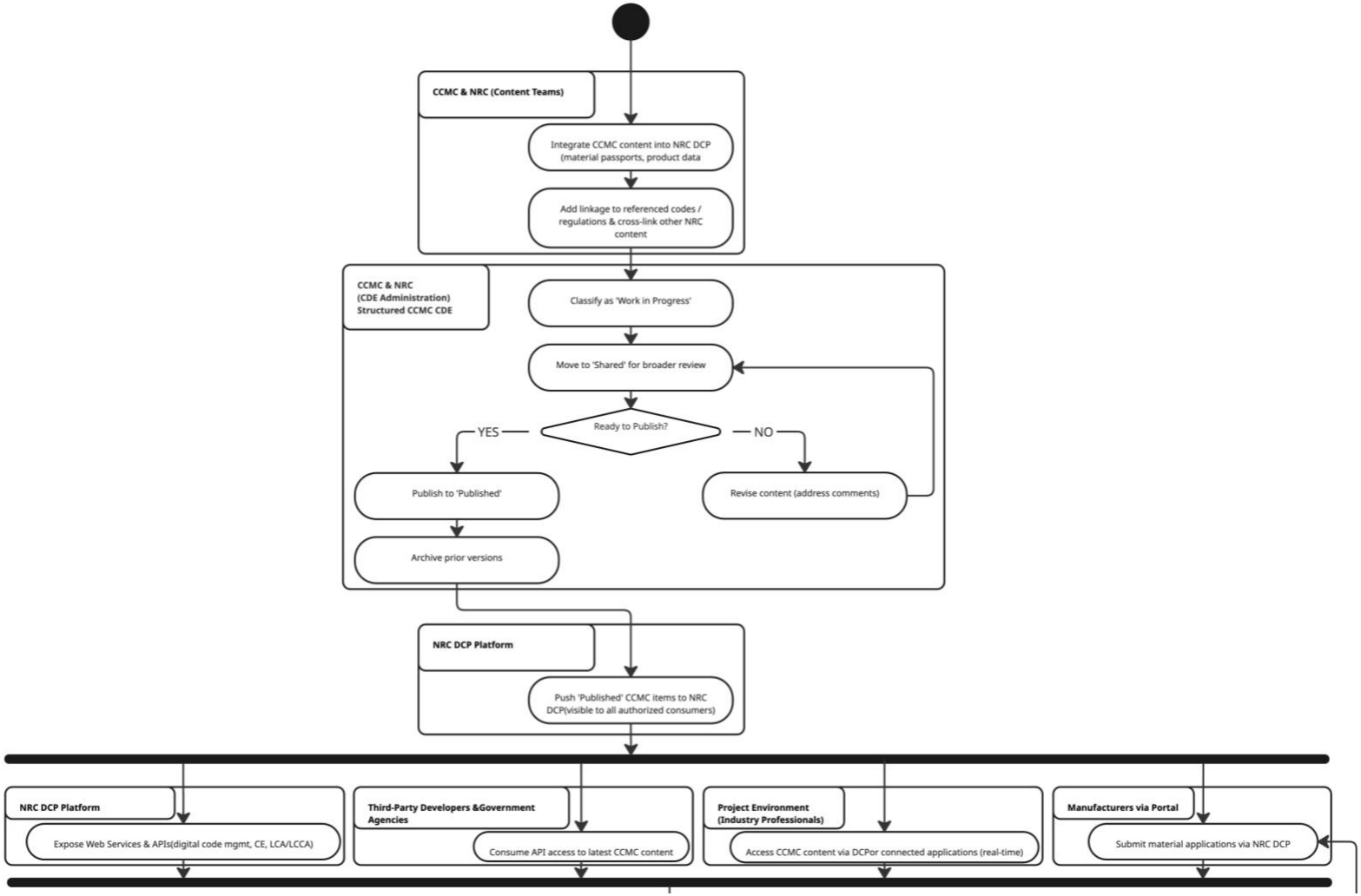

Figure 18. CCMC Activity Diagram (part 1)

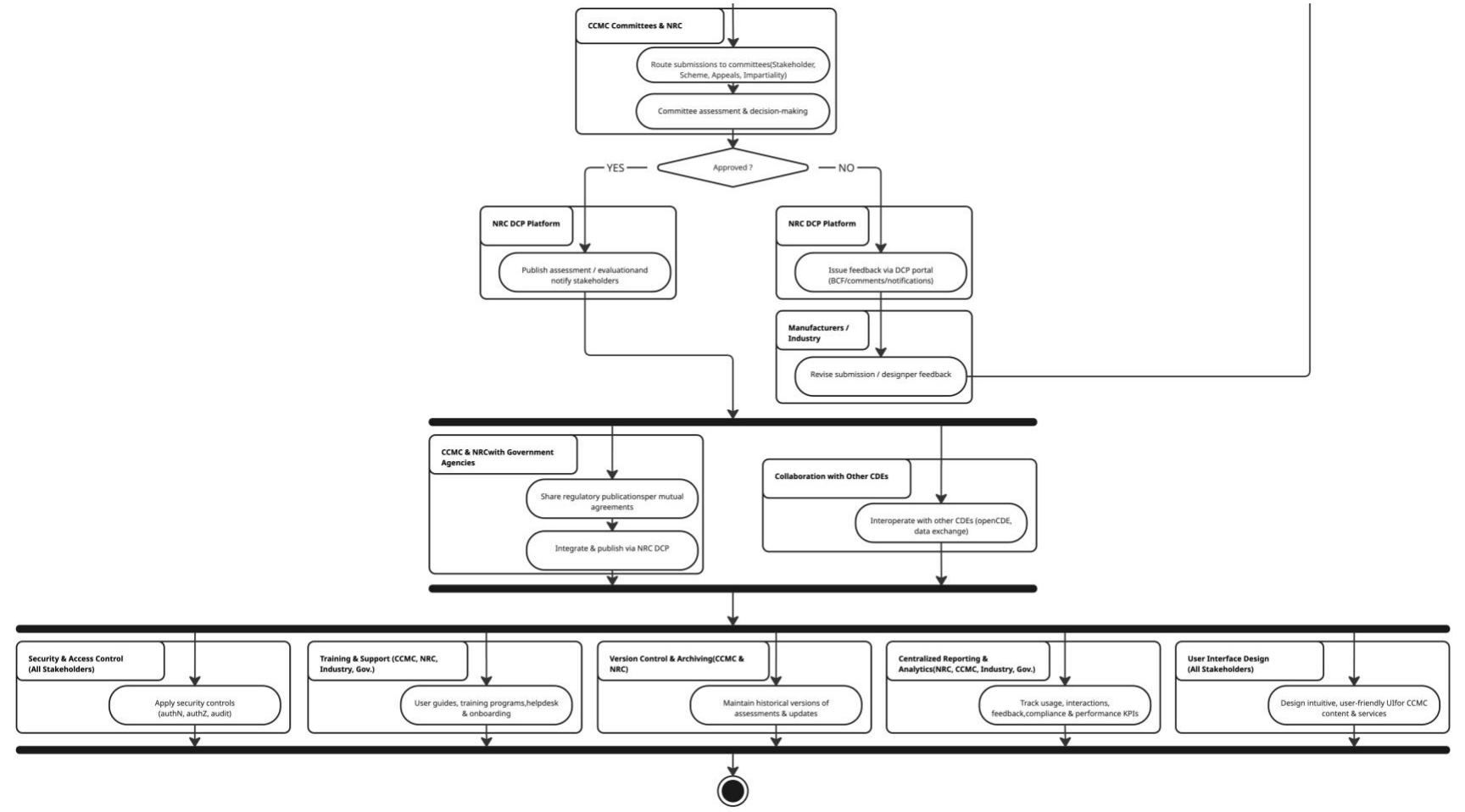

Figure 19. CCMC Activity Diagram (part 2)

Data model

Table 15 describes the classes within the CCMC data model whereas Table 16 describes its attributes and Table 17 . Figure 20 organizes CCMC Data Model for products, evaluations, and measurements as follows:

- **Actors and artefacts.** Organization (role: holder/manufacturer) produces Product(s) and holds EvaluationReport(s).
- **Products and typing.** Each Product isA one ProductType, may be assessedAgainst Requirement(s), classifiedAs Classification(s), has declared PropertyValue(s), and can be coveredBy zero or more EvaluationReport(s).
- **Declared values.** A PropertyValue is basedOn a Property and may specify a Unit, TestMethod, and date.
- **Reports and evidence.** An EvaluationReport includes Evidence; each Evidence is a TestResult.
- **Test results.** A TestResult is for a Property, and may have a Unit and be measuredBy a TestMethod.
- **Methods and standards.** A TestMethod is published by an Organization and may implement a ReferenceStandard; a ReferenceStandard may referTo a TestMethod.
- **Property/system metadata.** Property may indicate a default Unit and an optional bSDDMapping. Unit records symbol and system (QUDT/OM).
- **Classification.** Classification captures system (MasterFormat/UniFormat), code, and bilingual titles.
- **Report metadata.** EvaluationReport carries number, status, isPublished, and lastModified.

Table 15. CCMC data model classes

Class Name	Purpose	Role in the Data Model
Product	Represents a specific product evaluated by CCMC	A concrete, commercially available product submitted for CCMC evaluation. Each Product is manufactured by an Organization, is classified as a ProductType, and may be covered by one or more EvaluationReports. Products carry declared PropertyValues that represent manufacturer claims subject to CCMC verification.
Organization	Represents holder or manufacturer organizations	Captures the entities involved in product evaluation. The role attribute distinguishes between report holders (who commission evaluations) and manufacturers (who produce the products). Organizations may also publish TestMethods and ReferenceStandards.
ProductType	Represents a type or category of product	An abstract classification of products (e.g., "Structural Insulated Panel," "Insulating Concrete Form"). Products are instances of ProductTypes, enabling queries across all products of a given category. The mapping attribute supports alignment with bSDD or IFC classifications.

Requirement	Represents a requirement that products are assessed against	Captures the NBC or other code requirements against which products are evaluated. Each Requirement links to specific code clauses and defines what must be satisfied for compliance. Products are assessedAgainst Requirements to establish their scope of compliance.
Classification	Organizes products using MasterFormat/UniFormat	Aligns products with industry classification systems for discoverability and interoperability. Supports both MasterFormat (work results) and UniFormat (building elements) classification schemes.
PropertyValue	Represents declared property values for products	Captures the manufacturer's declared value for a specific Property (e.g., "fire-resistance rating = 1 hour"). Each PropertyValue is basedOn a Property and may include the Unit, TestMethod used for measurement, and date of determination. These declared values are subject to verification through CCMC evaluation.
Property	Represents a measurable or definable characteristic	A named attribute that can be measured or specified (e.g., fire-resistance rating, thermal resistance, structural capacity). Properties are defined independently of values, enabling reuse across products and requirements. The mapping attribute supports alignment with bSDD or IFC property definitions.
EvaluationReport	Represents a CCMC evaluation report	The official CCMC assessment document that covers one or more Products. Each report carries a unique number, status (draft, active, expired), publication state, and modification date. Reports include Evidence that substantiates compliance claims. This is the authoritative artifact referenced by building officials and specifiers.
Evidence	Represents evidence supporting evaluations	Documentation that substantiates claims in an EvaluationReport. Evidence may be text (written analysis), calc (engineering calculations), or report (laboratory test report). Each Evidence record links to specific TestResults that provide the underlying data.
TestResult	Represents actual test results from laboratory testing	The measured outcome of a test performed on a product. Each TestResult captures the value obtained, the Property being measured, the Unit of measurement, and the TestMethod used. TestResults provide the empirical foundation for Evidence in evaluation reports.
Unit	Represents a unit of measurement	Standardized units (e.g., hours, mm, kPa, $m^2 \cdot K/W$) aligned with QUDT or OM ontologies. Units ensure consistent interpretation of PropertyValues and TestResults across products and reports.
TestMethod	Represents a standardized testing method	The procedure used to measure a property (e.g., CAN/ULC-S101 for fire endurance, ASTM E84 for surface burning). Each TestMethod is published by an Organization and may

		implement a ReferenceStandard. Links test results to the verification procedures used.
ReferenceStandard	Represents a reference standard	External standards (e.g., CAN/ULC-S101, ASTM E119, CSA A23.3) that define test methods, material specifications, or performance criteria. Standards are referenced by evaluation reports and may be linked to specific TestMethods that implement them.

Table 16. CCMC data model attributes

Class	Attribute	Data Type	Description
Product	id	IRI	Unique identifier for the product
	name	string	Name of the product
	manufacturer	Organization	Organization that manufactures the product
	identifiers	string	Identifiers for the product
Organization	id	IRI	Unique identifier for the organization
	name	string	Name of the organization
	role	{holder/manufacturer}	Role of the organization (holder or manufacturer)
Requirement	id	IRI	Unique identifier for the requirement
	label	string	Label of the requirement
	text	string	Text content of the requirement
Classification	system	{MasterFormat/UniFormat}	Classification system (MasterFormat or UniFormat)
	code	string	Code within the classification system
	title	string	Title of the classification
Evaluation Report	id	IRI	Unique identifier for the evaluation report
	number	string	Number of the evaluation report
	status	string	Status of the evaluation report
	isPublished	boolean	Whether the evaluation report is published
	lastModified	date	Date when the evaluation report was last modified
ProductType	id	IRI	Unique identifier for the product type
	label	string	Label of the product type
	mapping	URI	mapping for the product type (to bSDD or other)
Property	id	IRI	Unique identifier for the property
	label	string	Label of the property
	definition	string	Definition of the property
	unitKind	Unit?	Kind of unit for the property (optional)
	mapping	URI	mapping for the property (to bSDD or other)

Property Value	id	IRI	Unique identifier for the property value
	value	any	Value of the property
	unit	Unit?	Unit of the property value (optional)
	method	TestMethod?	Test method used (optional)
	date	date?	Date of the property value (optional)
TestResult	id	IRI	Unique identifier for the test result
	value	any	Value of the test result
	unit	Unit?	Unit of the test result (optional)
	method	TestMethod?	Test method used (optional)
	date	date?	Date of the test result (optional)
Unit	id	IRI	Unique identifier for the unit
	symbol	string	Symbol representing the unit
	system	QUDT/OM	Unit system (QUDT or OM)
Evidence	id	IRI	Unique identifier for the evidence
	kind	{text, calc, report}	Kind of evidence (text, calculation, or report)
	link	URL	URL link to the evidence
TestMethod	id	IRI	Unique identifier for the test method
	code	string	Code of the test method
	title	string	Title of the test method
	publisher	Organization	Organization that published the test method
Reference Standard	id	IRI	Unique identifier for the reference standard
	code	string	Code of the reference standard
	edition	string	Edition of the reference standard

Table 17. CCMC Data model relationships and cardinalities

From Class	To Class	Rel.	Card.	Meaning
Organization	Product	produces	1..*	Each Organization produces one or more Products.
Organization	EvaluationReport	holds	1..*	Each Organization holds one or more EvaluationReports.
Product	ProductType	isA	1..1	Each Product is of exactly one ProductType.
Product	Requirement	assessedAgainst	0..*	Each Product is assessed against zero or more Requirements.
Product	Classification	classifiedAs	0..*	Each Product is classified under zero or more Classifications.

Product	PropertyValue	hasDeclared	0..*	Each Product has zero or more declared PropertyValues.
Product	EvaluationReport	coveredBy	0..*	Each Product is covered by zero or more EvaluationReports.
Product	Property	hasDeclared	0..*	Each Product has declared by zero or more Property.
EvaluationReport	Evidence	is	1..1	Each EvaluationReport is a pieces of Evidence.
EvaluationReport	TestResult	includes	0..*	Each EvaluationReport includes zero or more TestResults.
EvaluationReport	TestMethod	uses	0..*	Each EvaluationReport useses zero or more TestResults.
EvaluationReport	ReferenceStandard	tefersTo	0..*	Each EvaluationReport refers to zero or more TestResults.
PropertyValue	Evidence	SupportedBy	0..*	Each PropertyValue is supported by zero or one Evidence.
Property	Unit	unit	0..1	Each Property is expressed in zero or one Unit.
Property	TestMethod	for	1..1	Each Property has exactly one corresponding TestResult.
Property	PropertyValue	for	1..1	Each Property has one PropertyValue.
TestResult	Unit	unit	0..1	Each TestResult has zero or one Unit.
TestResult	TestMethod	measuredBy	0..1	Each TestResult is measured by zero or one TestMethod.
TestResult	PropertyValue	basedOn	0..1	Each TestResult is based on zero or one PropertValue.
TestMethod	ReferenceStandard	implements	0..*	Each TestMethod implements zero or more ReferenceStandards.

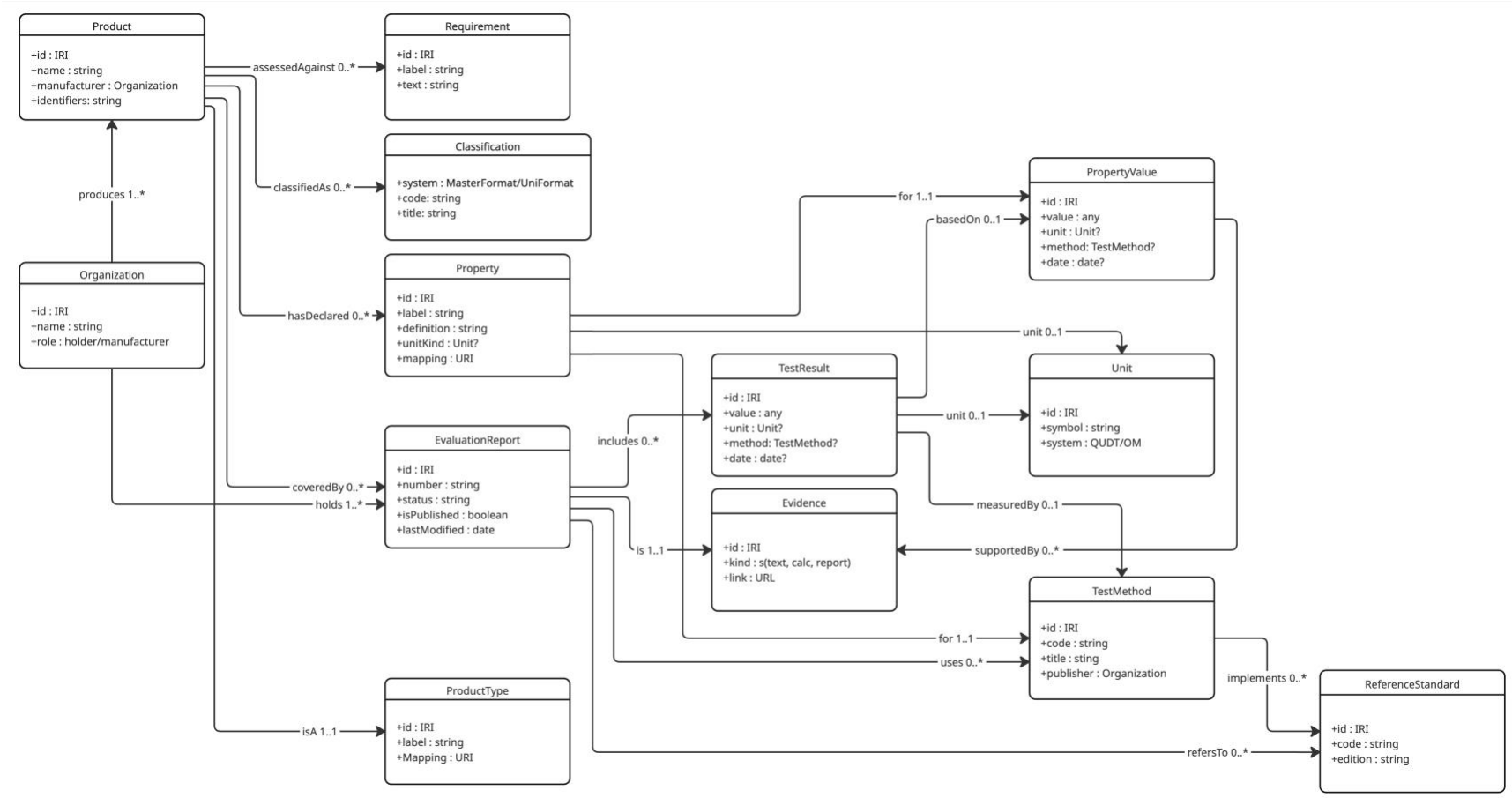

Figure 20. CCMC Data Model

4.5. Consolidated Data Model

Figure 21 presents a single data model that unifies the three domains with shared metadata. The consolidated data model integrates the three domain-specific models (NMCC, NMS, CCMC) into a unified schema that enables cross-resource queries and traceability. Rather than defining new classes, this model shows how existing classes from each domain connect through shared entities (Requirement, ProductType, Property, ReferenceStandard, Unit, Organization, Provenance). The consolidation enables use cases such as:

- Tracing from a code clause to the specification sections that address it and the products that comply
- Finding all products evaluated against a specific requirement
- Identifying which test methods and reference standards support a product's claimed properties

The model includes:

1. **Codes (NMCC).** The regulatory text is modeled as a hierarchy CodeVolume → Division → Part → Section → Subsection → Article → Paragraph → Sentence. Each Sentence can contain a RegulatoryClause (with optional Subclause). A RegulatoryClause defines a Requirement, may referTo ReferenceStandard, use DefinedTerm, and carries Provenance.
2. **Specifications (NMS).** Project-facing constraints are expressed with RequirementTemplate, which includes QualityCriterion, requires Submittal, offers Option, and has Constraint, Parameter, and Threshold. Constraint uses a Comparator (with its own Provenance); Threshold uses a Unit. Parameter can point to the subject (ProductType), the checked Property, and the Unit. Requirement specifies one or more SpecificationSection(s).
3. **Products & Evaluations (CCMC).** Organization produces Product(s) and holds EvaluationReport(s). A Product isA ProductType, may be assessedAgainst Requirement(s), be classifiedAs Classification, have declared PropertyValue(s), and be coveredBy EvaluationReport(s). Each PropertyValue is basedOn a Property and may record Unit and TestMethod. Reports include Evidence; each Evidence is a TestResult (with Property, optional Unit, and TestMethod). TestMethod can implement a ReferenceStandard and is issuedBy an Organization.
4. **Shared metadata.** SourceDocument is publishedBy an Organization. Provenance (with Agent) attaches to SpecificationSection and Product, ensuring traceability across sources and edits. Property can use a Unit and be governedBy / isA ReferenceStandard.
5. **Attribute Optionality and Unit Conventions.** Attributes marked with a question mark (e.g., Unit?, date?, TestMethod?) are optional. The following rules govern their usage:
    a. Units:
        i. Units are mandatory for all quantitative PropertyValues and TestResults (numeric values without units are invalid).

ii. Units are optional for qualitative or categorical properties (e.g., color, material type)
iii. All units must reference the QUDT or OM ontology to ensure consistent interpretation across systems
iv. When a Property defines a unitKind, all PropertyValues for that Property must use a compatible unit from the same dimension (e.g., length, time, temperature)

b. Comparators:
   i. Comparators are modeled as a separate class with the following attributes:
      1. operator: the logical operator (e.g., greaterThanOrEqual, lessThan, equals, between)
      2. label: human-readable name
      3. operatorSymbol: mathematical symbol (≥, ≤, =, <, >)
   ii. Each Constraint uses exactly one Comparator to define how a Property value is evaluated against a Threshold

c. Dates:
   i. Dates are optional but recommended for PropertyValues and TestResults to establish temporal validity
   ii. When present, dates use ISO 8601 format (YYYY-MM-DD)

d. TestMethod:
   i. TestMethod is optional for declared PropertyValues (manufacturer claims)
   ii. TestMethod is mandatory for TestResults in Evidence (verified claims must cite the method used)

6. **Cross-resource links.** A Requirement appliesTo ProductType and constrains Property, connecting code clauses to specification templates and to product/property data. These links allow Product declarations and EvaluationReport evidence to be traced back to the RegulatoryClause and SpecificationSection they satisfy.

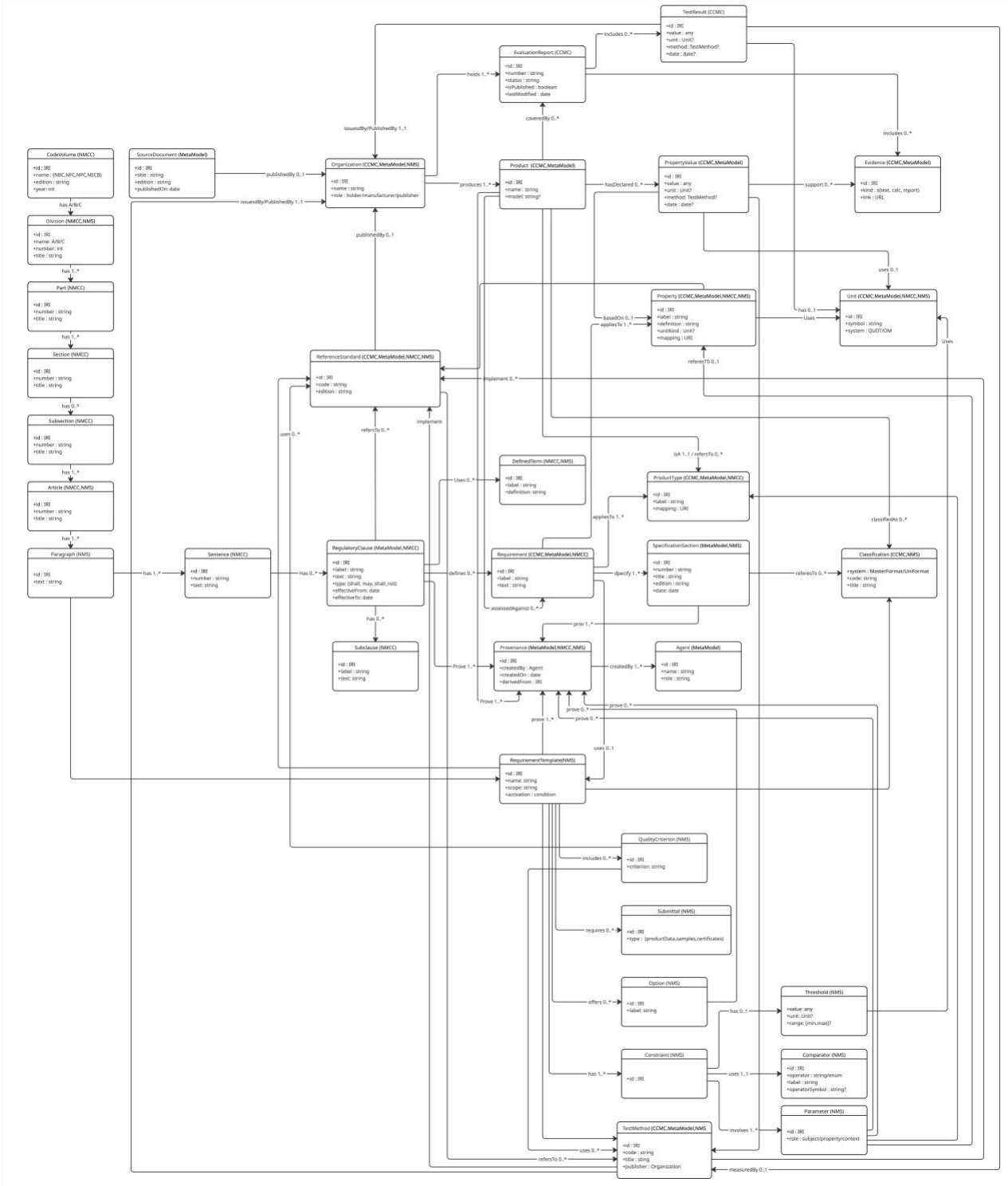

Figure 21. Consolidated Data Model

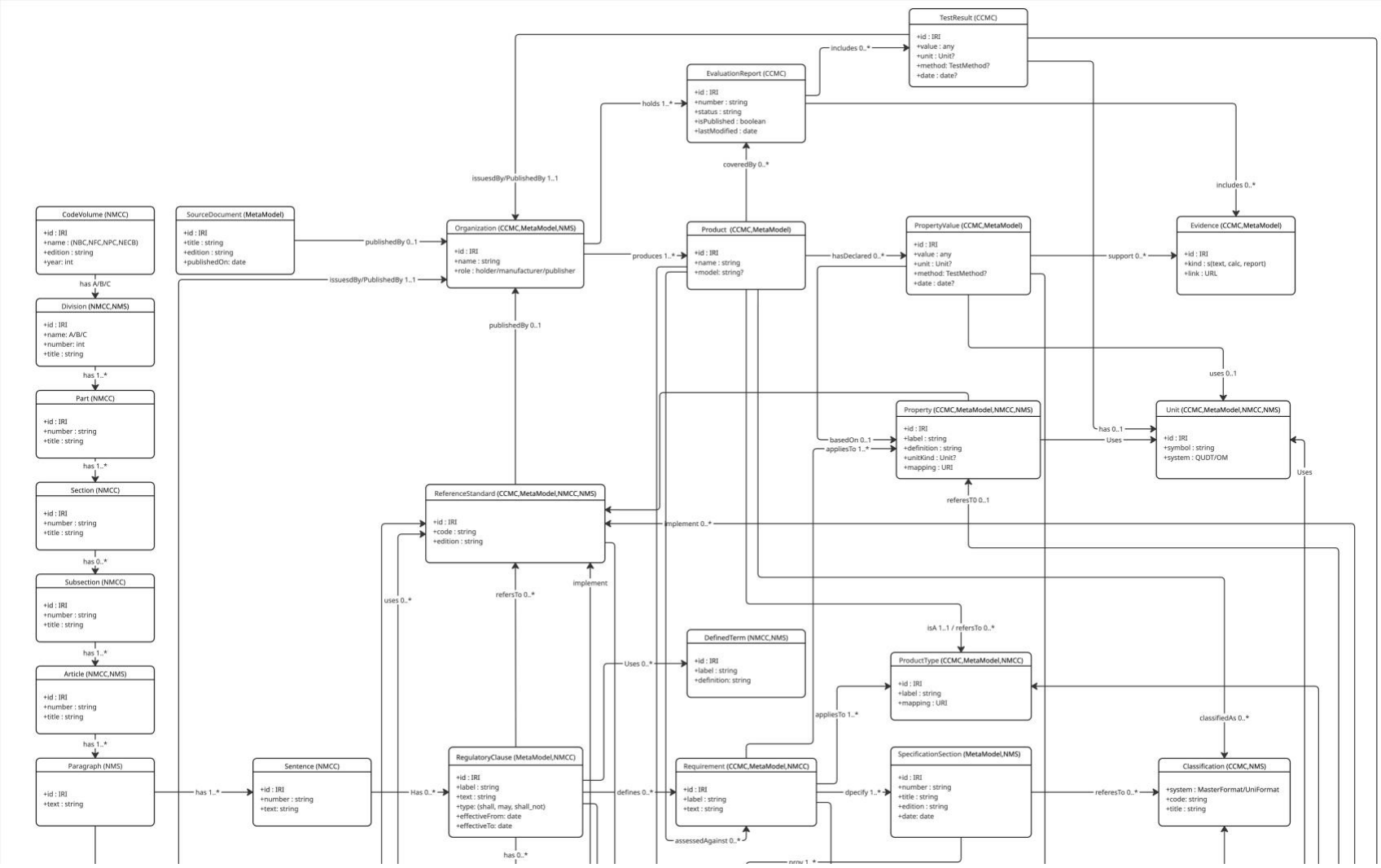

Figure 22. Consolidated data model (part 1)

Figure 23. Consolidated data model (part 2)Figure 24. Consolidated data model (part 1)

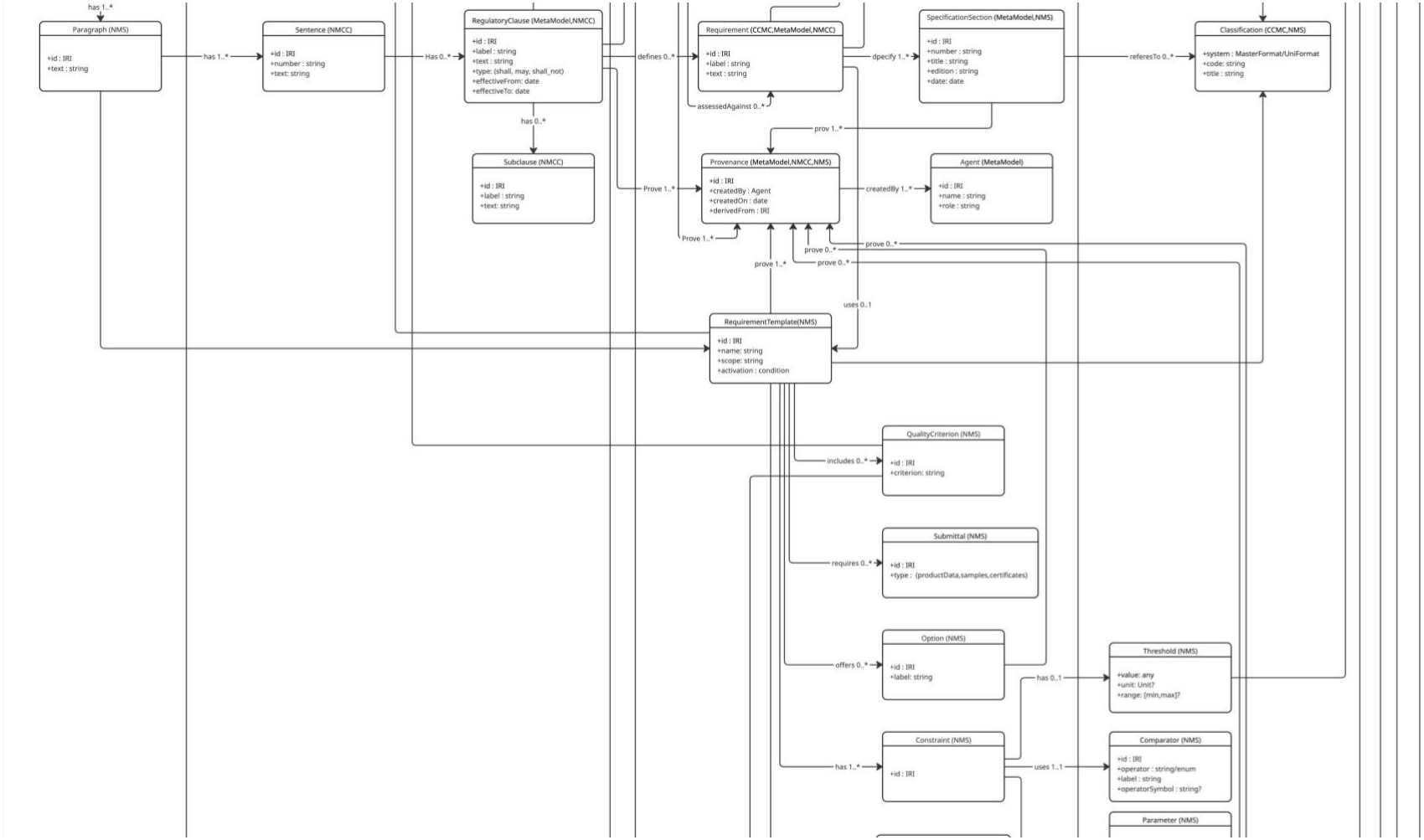

Figure 25. Consolidated data model (part 2)

Figure 26. Implementation scenarioFigure 27. Consolidated data model (part 2)

5. Machine-interpretable format requirements

In digitalizing their available resources, NRC is committing to providing and making available key data assets in digital formats in an open, neutral, interoperable and transparent manner. The principles of Open Government of Canada[1] and the Policy on Services and Digital[2] can be called upon in this regard. The objective of Open Government of Canada is to "[...] maximize the release of government information and data of business value to support transparency, accountability, citizen engagement, and socio-economic benefits through reuse, subject to applicable restrictions associated with privacy, confidentiality, and security", whereas the Policy on Services and Digital is to improve client service experience and government operations through digital transformation approaches. Although both policies are not completely related to the NRC's DCP initiative, they can serve to position the DCP's foundational mechanisms, namely as when it comes to open data requirements. Indeed, while the Open Government of Canada initiative pertains to governmental data assets, its principles could be applied to government resources that are made available to the public such as the NRC's resources. The same goes for the Policy on Services and Digital, which states that information and data should be discoverable and considered an open resource. In both cases, data and information should be open, neutral, interoperable and transparent. This means that data is preserved and accessible over the long-term, ensures interoperability between different software which reducing licensing fees and preventing vendor lock-in, can enable community-driven development and improvement (open-source) and ensure transparency in how data is stored.

To this effect, the U.S. Congress Open Government Data Act[3] states that any 'open Government data asset', is considered a data asset maintained by the Federal Government that is (A) machine-interpretable, (B) available in an open format, (C) not encumbered by restrictions that would impede use or reuse; and (D) based on an underlying open standard that is maintained by a standards organization. In this case, the term 'machine-interpretable' means a format in which information or data can be easily processed by a computer without human intervention while ensuring no semantic meaning is lost.

Machine-interpretable formats combine structured syntax, semantic clarity, validation capability, and technical robustness within a standards-compliant, interoperable, and extensible framework. Such a format allows machines not only to read data, but to understand, validate, and act upon it. More

[1] https://www.tbs-sct.canada.ca/pol/doc-eng.aspx?id=28108 accessed October 09, 2025
[2] https://www.tbs-sct.canada.ca/pol/doc-eng.aspx?id=32603 accessed October 09, 2025
[3] https://www.congress.gov/115/bills/s760/BILLS-115s760is.pdf accessed June 26, 2025

specifically, machine-interpretable format requirements define the technical and semantic characteristics that allow data to be automatically processed, parsed, validated, and understood by software systems without human intervention. These requirements will ensure consistency, interoperability, and automation in the NRC DCP.

5.1. Core Requirements

Machine-interpretable formats must include complete, unambiguous documentation of their structure, syntax, and semantics so that different systems can interpret them consistently. 'Parsability' is equally crucial as the data must follow predictable rules that can be reliably processed by parsers and automated tools.

It is important to adhere to recognized standards such as ISO, W3C, or IEEE to ensure interoperability and broad tool support across platforms. The format should represent structured data, where hierarchical relationships between elements are explicit. Data types such as strings, integers, dates, and enumerations should be clearly defined to prevent misinterpretation and support accurate validation and computation.

A robust schema validation mechanism (for instance, XML Schema, JSON Schema, or SHACL for RDF data) allows automated systems to verify that data instances conform to the defined model. This helps detect inconsistencies early and ensures that the data remains machine-consistent and semantically valid.

5.2. Technical Characteristics

To achieve reliable automation, the format must enable deterministic parsing, ensuring that identical inputs always yield the same interpretation. No ambiguity should exist in how elements are defined or related; every field or tag must have exactly one meaning.

Character encoding should be explicitly stated (commonly UTF-8) to prevent corruption or misinterpretation across systems. For XML- or RDF-based formats, namespace support is essential to prevent naming collisions and maintain clarity when integrating data from multiple sources.

Each dataset or instance should include a version identifier, allowing software to manage format evolution and ensure compatibility. Well-defined error handling procedures must specify how systems respond to malformed or incomplete data, preventing data loss or silent failures.

5.3. Practical Requirements

Beyond formal design, machine-interpretable formats must also meet practical usability and integration needs. Wide tool support across programming languages and platforms ensures accessibility and

sustainability. The format should allow forward and backward compatibility, enabling systems to handle both older and newer data versions.

Extensibility is another key principle, whereby new attributes or structures should be added without breaking existing workflows. While not mandatory, human-readability (for example, using text-based formats such as JSON or XML rather than binary) facilitates debugging and transparency during development.

The format should also support metadata embedding, including provenance, authorship, timestamps, and contextual information, among other metadata requirements. This metadata is vital for traceability, validation, and governance, especially in regulatory or compliance-driven environments.

5.4. Semantic and Interoperability Considerations

Each term, property, or relationship will be explicitly defined through an ontology hosted within a data dictionary compliant with ISO 12006-2 and ISO 12006-3. Properties should be defined following approaches identified in ISO 23386. Semantic annotations ensure that meaning is unambiguous and machine-interpretable. The development of a data dictionary for NRC resources is a high priority item.

The format should support cross-schema mapping and interoperability with other data standards (e.g., IFC to RDF to JSON-LD and vice-versa), using consistent identifiers and equivalence relationships. Persistent, globally unique identifiers (URIs or GUIDs) are critical for maintaining referential integrity across datasets and systems.

In advanced implementations, machine-executable logic, represented through SHACL rules, SPARQL constraints, or logic-based expressions, enables automated reasoning, compliance checking, or inference, turning static data into meaningful information.

5.5. Recommended open data formats

There are several available and recommendable open data formats that can support the NRC DCP initiative. While open does not necessarily mean machine-interpretable, it at least means digitized, which is a step in the right direction. Table 18 is adapted from opendata.swiss[4] and describes recommended

[4] https://www.bfs.admin.ch/bfs/en/home/dienstleistungen/ogd/dokumentation/empfohlenen-formate.html accessed on June 26, 2025

open data formats. Table 19 identifies suggested format types per DCP use case or service. Table 20 maps format types to data model classes.

Table 18. Recommended open data formats (adapted from opendata.swiss)

Data Type	Format	Standard/ Spec.	Applicability	Primary Use Cases	Attributes Supported
Structured Data	CSV (Comma Separated Values)	RFC 4180	All	Data export, tabular reports, spreadsheet integration, bulk data exchange	Simple attribute lists, flat data tables, property values
	JSON (JavaScript Object Notation)	RFC 4627	All	API responses, lightweight data exchange, web applications, configuration files	All attributes; nested objects for relationships (e.g., Product → PropertyValue → Unit)
	JSON-LD (JSON for Linked Data)	W3C JSON-LD 1.1	All	Semantic data exchange, linked data, API responses with context, cross-resource linking	All attributes with semantic context; supports @id for IRIs, @type for classes
	XML (Extensible Markup Language)	W3C REC	NMCC, NMS	Document markup, DITA content, legacy system integration, structured authoring	Hierarchical content (CodeVolume → Division → Part → Section); document metadata
	RDF/Turtle	W3C Turtle	All	Ontology definitions, semantic triples, SHACL shape definitions, linked data	All classes and relationships; ideal for Property, ProductType, Requirement definitions
	RDF/N-Triples	W3C N-Triples	All	Bulk RDF data loading, streaming, large-scale graph processing	All classes and relationships; optimized for machine processing
	Apache Parquet	Apache Parquet	All	Large-scale analytics, data warehousing, columnar storage	PropertyValue, TestResult aggregations; analytics on Evidence, EvaluationReport

	IFC (Industry Foundation Classes)	ISO 16739-1:2024	NMS, CCMC	BIM model exchange, product geometry, building elements, spatial context	ProductType (via IfcTypeProduct), Property (via IfcPropertySet), Classification
Text	TXT (Plain Text)	ISO 8859-1, ISO 8859-15, ISO 10646 (UTF-8)	All	Simple text export, logs, human-readable dumps	Text attributes: RegulatoryClause.text, Requirement.text, DefinedTerm.definition
	XHTML	W3C REC	NMCC, NMS	Web publishing, styled document output, online code/specification viewing	Document structure with presentation; Section, Article, Paragraph content
	PDF/A	ISO 19005	All	Archival documents, published reports, official publications, print output	EvaluationReport (published), SourceDocument, ReferenceStandard documents
	DOCX (Office Open XML)	ISO/IEC 29500	All	Editable documents, collaborative authoring, tracked changes	Document drafts, specification sections under review
	ODT (OpenDocument Text)	ISO/IEC 26300-1:2015	All	Open-source document editing, vendor-neutral text documents	Same as DOCX; preferred for open-source toolchains
	XLSX (Office Open XML Spreadsheet)	ISO/IEC 29500	All	Tabular data with formulas, data entry templates, interactive analysis	PropertyValue tables, TestResult summaries, Classification codes
	ODS (OpenDocument Spreadsheet)	ISO/IEC 26300-1:2015	All	Open-source spreadsheet editing, vendor-neutral tabular data	Same as XLSX; preferred for open-source toolchains
	DITA XML	OASIS DITA 1.3	NMCC, NMS	Structured authoring, topic-based content, reusable components, localization	RegulatoryClause, Sentence, Article, SpecificationSection; supports specializations for compliance expressions

Geodata	GeoJSON	RFC 7946	NMCC	Geographic boundaries, jurisdictional zones, site location data	Land-use context, jurisdictional applicability of codes
	KML (Keyhole Markup Language)	OGC KML	NMCC	Map visualization, geographic annotations, location markup	Jurisdictional boundaries, regional code applicability
	GML (Geography Markup Language)	OGC GML	NMCC	Complex geographic features, spatial data exchange	Detailed spatial context for code application
	GeoPackage	OGC GeoPackage	NMCC	Portable geodatabase, offline geographic data, mobile applications	Combined geographic and attribute data for field use
Images & Graphics	PNG	ISO/IEC 15948:2004	All	Diagrams, figures, lossless graphics, documentation images	Data model diagrams, use case diagrams, activity diagrams
	SVG (Scalable Vector Graphics)	W3C REC	All	Scalable diagrams, interactive graphics, web-based visualizations	Data model diagrams (editable), traceability chain visualizations
	JPEG2000	ISO/IEC 15444-1:2004	CCMC	Product images, test specimen photographs, evidence documentation	Evidence.link (image attachments), product documentation
	TIFF	TIFF Revision 6.0	CCMC	High-quality archival images, scanned documents, test report images	Archival evidence, scanned certificates
Video/Audio	MPEG4	ISO/IEC 14496	CCMC	Test procedure recordings, training materials, product demonstrations	Evidence (video type), TestMethod demonstrations
	WebM	WebM Documentation	All	Web-based video, training content, online documentation	Training materials, procedure videos
Graph/ Semantic	GraphML	GraphML	All	Network visualization, relationship diagrams, data model visualization	Class relationships, traceability networks

	GEXF (Graph Exchange XML Format)	GEXF	All	Dynamic graph data, temporal networks, graph analytics	Versioned relationships, change tracking visualization
	OWL (Web Ontology Language)	W3C OWL 2	All	Ontology definitions, class hierarchies, semantic reasoning	Metamodel class definitions, Property hierarchies, ProductType taxonomies
	SHACL (Shapes Constraint Language)	W3C SHACL	All	Data validation, constraint definitions, compliance checking	Constraint expressions, validation rules for all classes
	SPARQL Results (JSON/XML)	W3C SPARQL	All	Query results, data retrieval, cross-resource queries	Query outputs for any attribute combination

Table 19. Format type by use case/service

Use Case	Recommended Primary Format	Alternative Formats	Key Attributes
Data Dictionary / Ontology	RDF (Turtle) + OWL	JSON-LD	Property, ProductType, DefinedTerm, Unit definitions, Requirement, Constraint, TestMethod, Jurisdiction, etc.
Validation Rules	SHACL	JSON Schema	Constraint, Comparator, Threshold expressions
API Data Exchange	JSON-LD	JSON, XML	All classes; Product, PropertyValue, EvaluationReport
Document Authoring (NMCC)	DITA XML	XML, DOCX	RegulatoryClause, Sentence, Article, Section hierarchy
Document Authoring (NMS)	DITA XML	XML, DOCX	SpecificationSection, Article, Paragraph, RequirementTemplate
Product Data (CCMC)	JSON-LD	JSON, IFC	Product, PropertyValue, EvaluationReport, Evidence
BIM Integration	IFC + JSON-LD	IFC-XML	ProductType, Property, Classification mappings
Compliance Checking	RDF + SHACL + SPARQL	JSON-LD + JSON Schema	Requirement, Constraint, PropertyValue, TestResult
Published Reports	PDF/A	DOCX, HTML	EvaluationReport, SourceDocument
Bulk Data Export	CSV, Parquet	JSON	PropertyValue lists, TestResult aggregations
Cross-Resource Queries	SPARQL (over RDF)	GraphQL (over JSON)	All relationships; NMCC→NMS→CCMC traceability
Geographic Context	GeoJSON	GML, KML	Jurisdictional applicability, land-use data
Diagrams & Visualizations	SVG	PNG	Data model diagrams, traceability chains

Table 20. Format mapping to data model classes

Class	Primary Serialization	Validation Format	Exchange Format
RegulatoryClause	DITA XML, RDF/Turtle	SHACL	JSON-LD
Requirement	RDF/Turtle	SHACL	JSON-LD
Constraint	RDF/Turtle, SHACL	SHACL	JSON-LD
RequirementTemplate	DITA XML, RDF/Turtle	SHACL	JSON-LD
SpecificationSection	DITA XML	XML Schema, SHACL	JSON-LD, XML
Product	RDF/Turtle	SHACL	JSON-LD, IFC
ProductType	RDF/Turtle, OWL	SHACL	JSON-LD, IFC
Property	RDF/Turtle, OWL	SHACL	JSON-LD, IFC (IfcPropertyDefinition)
PropertyValue	RDF/Turtle	SHACL	JSON-LD, IFC (IfcPropertySingleValue)
EvaluationReport	RDF/Turtle	SHACL	JSON-LD, PDF/A (published)
Evidence	RDF/Turtle	SHACL	JSON-LD
TestResult	RDF/Turtle	SHACL	JSON-LD, CSV (bulk)
TestMethod	RDF/Turtle	SHACL	JSON-LD
ReferenceStandard	RDF/Turtle	SHACL	JSON-LD
Unit	RDF/Turtle (QUDT/OM)	SHACL	JSON-LD
Classification	RDF/Turtle	SHACL	JSON-LD
Provenance	RDF/Turtle (PROV-O)	SHACL	JSON-LD
Organization	RDF/Turtle	SHACL	JSON-LD
DefinedTerm	RDF/Turtle, SKOS	SHACL	JSON-LD

6. Implementation scenario

The following section provides an implementation scenario demonstrating how the digitalized NRC resources work together to support automated compliance verification. The workflow would include (1) Identify applicable NBC requirement, (2) Link to NMS specification guidance, (3) Select CCMC-evaluated product, and (4) Validate compliance automatically. In addition, the digitalization process aims to support and inform decision making throughout the project lifecycle. The architect should therefore be able to query and call upon the resources to validate their decision through model-base workflows.

In this example, an architect is designing a three-storey multi-unit residential building under Part 9 of the NBC. The design includes fire-rated wall assemblies separating dwelling units, which must comply with NBC 2020 requirements, reference appropriate NMS specification sections, and incorporate CCMC-evaluated products with verified performance data.

The compliance verification workflow operates in two main phases: a Design Workflow where the architect queries NRC resources to inform design decisions and develop the BIM model, and a Validation Workflow where the completed model is checked against code requirements with results fed back for correction if needed as shown in Figure 24.

In Figure 24, the central column shows the architect's design process progressing from initial requirements query through design development, product selection, specification preparation, and model finalization. The left column shows the Code resources (NMCC) that inform the design with NBC requirements, constraints, and defined terms. The right column shows the Materials resources (NMS and CCMC) that provide specification templates and evaluated products with supporting evidence.

Design Workflow

During the design workflow, the architect first queries the NMCC to identify applicable NBC requirements for the project context. For fire separations between dwelling units, NBC Division B, Part 9, Section 9.10, Article 9.10.9.16 establishes requirements including a minimum fire-resistance rating of one hour. The NMCC also provides defined terms such as "fire separation," "dwelling unit," and "fire-resistance rating" to ensure consistent interpretation.

With requirements understood, the architect searches the CCMC database for products meeting the identified constraints. For example, searching for Type X gypsum board products with fire-resistance ratings of at least one-hour returns evaluated products such as those covered by CCMC Reports. Each product listing includes evidence from test results conducted per reference standards such as CAN/ULC-S101.

The architect then develops the BIM model incorporating the selected products, with material properties, classification references, and links to CCMC evaluation reports embedded in the IFC data. The model drives the preparation of project specifications based on NMS Section 09 29 00 (Gypsum Board), ensuring consistency between the model and the specification documents.

Validation Workflow

Once the design is finalized, the IFC model enters the validation workflow. The information model contains the geometric and property data along with linked specifications and CCMC reports. A transformation layer extracts properties from the IFC entities, maps them to the NRC CDF ontology as JSON-LD, and identifies the spatial context to determine applicable requirements.

The validation engine then applies requirements and constraints derived from three sources: NMCC constraints, expressed as IDS-XML or SHACL shapes that encode NBC requirements, NMS constraints from specification templates that define submittal and quality requirements, and CCMC evidence verification that confirms declared product properties are supported by valid evaluation reports.

If validation passes, the design is deemed compliant. The system issues a compliance certificate documenting full traceability from each validated element back to its source NBC clause, supporting NMS section, and CCMC evidence. The project is then ready for permit application.

If validation fails, the system generates feedback using the BIM Collaboration Format (BCF). Each issue identifies the specific IFC element that failed, the NBC clause that was violated, the actual versus required values, and recommended corrections including references to NMS sections and CCMC-evaluated products that would satisfy the requirement. This BCF file can be imported directly into the architect's BIM authoring tool, enabling targeted revisions. The architect may then re-query the NMCC if requirements are unclear or search CCMC for alternative products before resubmitting for validation.

Feedback loop through BCF

When a wall assembly fails validation due to insufficient fire-resistance rating, the BCF feedback includes the element identification (wall name and IFC GlobalId), the requirement statement from NBC 9.10.9.16, the discrepancy between actual and required values, the spatial context confirming the wall separates dwelling units, recommended actions including specific CCMC report numbers for compliant assemblies, and direct links to the relevant NBC clause and NMS section in the DCP. This structured feedback enables the architect to understand precisely what failed, why it failed, and how to correct it, with full traceability to authoritative NRC resources.

The workflow demonstrates several integration points enabled by the consolidated data model. IFC properties map to NRC CDF Properties allowing BIM data to be validated against code constraints.

Classification references link to MasterFormat codes used by both NMS and CCMC, enabling cross-resource queries. External references can point to CCMC evaluation reports, establishing evidence chains from model elements to verified test results. Spatial relationships in IFC determine which requirements apply based on element function and adjacent spaces. Finally, BCF feedback references NBC clauses, NMS sections, and CCMC reports using persistent URIs, creating links back to authoritative sources.

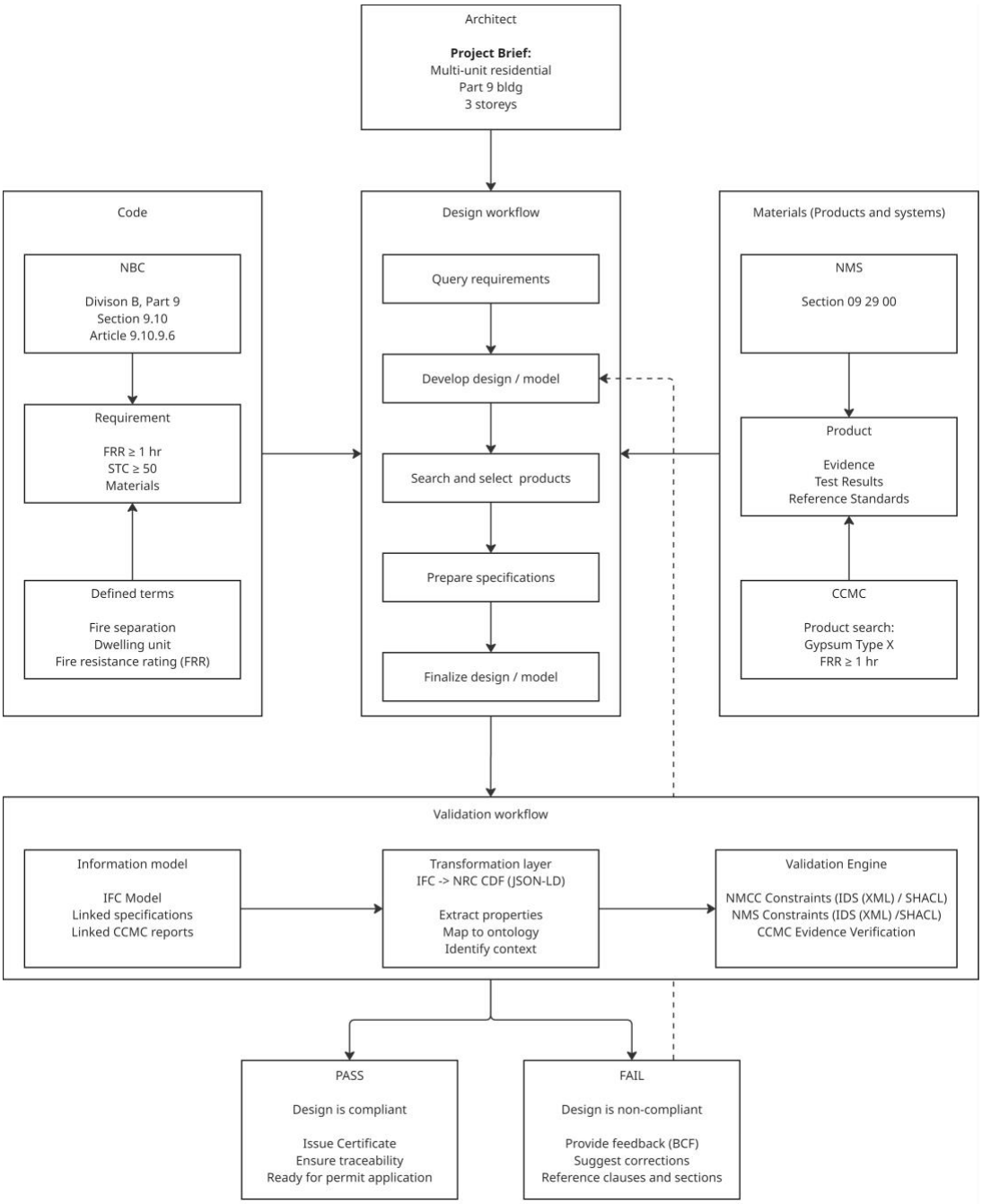

Figure 28. Implementation scenario

Instantiation

To illustrate the scenario, its requirement chain is described in Table 21. As described above, the chain starts from a NBC Clause formulated into a Requirement. The solution is described in a NMS Template which is fulfilled by CCMC Evidence. The information is then retrieved from the model to validate compliance.

Table 21. Scenario requirement chain

Step	Resource	Identifier	Content
1	NBC 2020 Clause	9.10.9.16 3)	"[...] dwelling units that contain 2 or more storeys including basements shall be separated from the remainder of the building by a fire separation having a fire-resistance rating of not less than 1 h."
2	Requirement	FRR-DU-1H	Fire-resistance rating ≥ 1 hour for fire separations between dwelling units
3	NMS Section	09 29 00	Gypsum Board — Type X for fire-rated assemblies
4	CCMC Report	XXXXX-R	Type X Gypsum Board 12.7mm — Fire-resistance rating: 1 hour (tested per CAN/ULC-S101)
5	IFC Element	IfcWall Pset_WallCommon IfcExternalReference	Wall assembly with FireRating property and CCMC reference

The following JSON-LD snippet shows how this chain could be represented as linked data with persistent URIs:

```
{
 "@context": {
  "nrc": "https://data.nrc.gc.ca/cdf/",
  "ifc": "https://standards.buildingsmart.org/IFC/RELEASE/IFC4/",
  "qudt": "http://qudt.org/schema/qudt/",
  "unit": "http://qudt.org/vocab/unit/",
  "nrc:ifcClass": { "@type": "@id" }
 },

 "@id": "nrc:nmcc/nbc2020/9.10.9.16/3",
 "@type": "nrc:RegulatoryClause",
 "nrc:text": " Except as provided in Sentence (4), dwelling units that contain 2 or more storeysincluding basements shall be separated from the remainder of the building by a fireseparation having a fire-resistance rating of not less than 1 h..",
 "nrc:clauseType": "shall",

 "nrc:defines": {
  "@id": "nrc:requirement/FRR-DU-1H",
  "@type": "nrc:Requirement",
  "nrc:label": "Fire Separation - Dwelling Units - 1 Hour",

  "nrc:appliesTo": { "@id": "nrc:producttype/fire-separation-assembly" },
  "nrc:constrains": { "@id": "nrc:property/fire-resistance-rating" },

  "nrc:minValue": 1.0,
  "nrc:unit": { "@id": "unit:HR" },

  "nrc:specifiedBy": { "@id": "nrc:nms/09-29-00" },

  "nrc:satisfiedBy": {
   "@id": "nrc:ccmc/XXXXX-R",
   "@type": "nrc:EvaluationReport",

   "nrc:covers": {
    "@id": "nrc:ccmc/product/type-x-gypsum-12.7mm",

    "nrc:hasPropertyValue": {
     "nrc:property": { "@id": "nrc:property/fire-resistance-rating" },
     "nrc:value": 1.0,
     "nrc:unit": { "@id": "unit:HR" },
     "nrc:testedPer": { "@id": "nrc:standard/CAN-ULC-S101" }
    }
   },

   "nrc:validatedIn": {
    "@id": "ifc:3MVDFw6f90Nfh9rP1dl_k5",
    "@type": "ifc:IfcWall",
    "nrc:ifcGuid": "3MVDFw6f90Nfh9rP1dl_k5",
    "nrc:ifcName": "W-FS-001"
   }
  }
 }
}
```

The following SHACL shape encodes the NBC requirement as a machine-executable constraint:

```
@prefix sh:   <http://www.w3.org/ns/shacl#> .
@prefix nrc:  <https://data.nrc.gc.ca/cdf/> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .
nrc:FireSeparationDwellingShape
   a sh:NodeShape ;
   sh:targetClass nrc:FireSeparation ;
   # Context: must separate dwelling units
   sh:property [
      sh:path nrc:separates ;
      sh:class nrc:DwellingUnit ;
      sh:minCount 2 ;
      sh:message "Fire separation must be between at least two dwelling units."@en ;
   ] ;
   # Constraint: FRR ≥ 1 hour
   sh:property [
      sh:path nrc:fireResistanceRating ;
      sh:datatype xsd:decimal ;
      sh:minCount 1 ;
      sh:minInclusive 1.0 ;
      sh:message "NBC 9.10.9.16.(3): Fire separation between dwelling units requires FRR ≥ 1 hour."@en;
   ] ;
   # Evidence: must have CCMC evaluation
   sh:property [
      sh:path nrc:evidencedBy ;
      sh:class nrc:EvaluationReport ;  # or nrc:CCMCEvaluationReport if you have that class
      sh:minCount 1 ;
      sh:message "Fire rating must be supported by at least one evaluation report (e.g. CCMC)."@en ;
   ] ;
   # Traceability: must be linked to an IFC element that has a GUID
   sh:property [
      sh:path nrc:validatedIn ;
      sh:class nrc:IFCElement ;
      sh:minCount 1 ;
      sh:node nrc:IFCElementShape ;
      sh:message "Fire separation must be linked to at least one IFC element with a GUID."@en ;
   ] .
nrc:IFCElementShape
   a sh:NodeShape ;
   sh:targetClass nrc:IFCElement ;
   # Must have an IFC GUID
   sh:property [
      sh:path nrc:ifcGuid ;
      sh:minCount 1 ;
      sh:datatype xsd:string ;
      sh:message "IFC element must have an IFC GUID (nrc:ifcGuid)."@en ;
   ] ;
```

While the SHACL Shape will allow the validation of physical constraints, namely the context and the values of certain data elements, IDS-XML will define what information the IFC model must contain to enable compliance checking. An example IDS for the scenario is provided :

```
<?xml version="1.0" encoding="UTF-8"?>
<ids xmlns="http://standards.buildingsmart.org/IDS">
 <info>
  <title>NBC 2020 Section 9.10.9.16 - Fire Separation Requirements</title>
  <version>1.0</version>
  <author>NRC Digital Content Platform</author>
  <purpose>Code Compliance Verification</purpose>
 </info>
 <specifications>
  <specification name="Fire Separation Wall - Dwelling Units"
                 ifcVersion="IFC4" necessity="required">
   <applicability>
    <entity><name><simpleValue>IFCWALL</simpleValue></name></entity>
    <partOf relation="IFCRELSPACEBOUNDARY">
     <entity><name><simpleValue>IFCSPACE</simpleValue></name></entity>
    </partOf>
   </applicability>
   <requirements>
    <!-- Fire Rating Property -->
    <property cardinality="required"
              instruction="NBC 9.10.9.16.(3): Provide fire-resistance rating ≥ 1 hour">
     <propertySet><simpleValue>Pset_WallCommon</simpleValue></propertySet>
     <name><simpleValue>FireRating</simpleValue></name>
     <value>
      <restriction base="xs:string">
       <pattern value="(1|1\.0|1\.5|2|2\.0)\s*[Hh]our(s)?"/>
      </restriction>
     </value>
    </property>
    <!-- CCMC Reference -->
    <material cardinality="required"
              instruction="Material must reference CCMC evaluation report">
     <value><simpleValue>Type X Gypsum Board</simpleValue></value>
    </material>
    <!-- NMS Classification -->
    <classification cardinality="required"
                    instruction="Must be classified per MasterFormat">
     <system><simpleValue>NMS</simpleValue></system>
     <value>
      <restriction base="xs:string">
       <pattern value="09\s*29\s*00"/>
      </restriction>
     </value>
    </classification>
   </requirements>
  </specification>
 </specifications>
</ids>
```

Finally, should the validation fail, the system would create a topic in BCF that can be imported into the BIM authoring tool:

```
<?xml version="1.0" encoding="UTF-8"?>
<Markup xmlns="http://www.buildingsmart-tech.org/bcf/bcf/2.1">
 <Header>
  <File IfcProject="0BTBFw6f90Nfh9rP1dl_k2" isExternal="false">
   <Filename>ResidentialBuilding.ifc</Filename>
   <Date>2025-01-15T14:30:00Z</Date>
  </File>
 </Header>
 <Topic Guid="a1b2c3d4-e5f6-7890-abcd-ef1234567890"
      TopicType="Error" TopicStatus="Open">
  <Title>NBC 9.10.9.16.(3) Violation: Insufficient Fire-Resistance Rating</Title>
  <Priority>High</Priority>
  <CreationAuthor>NRC DCP Compliance Checker</CreationAuthor>
  <Labels>
   <Label>Code Compliance</Label>
   <Label>Fire Safety</Label>
   <Label>NBC 2020</Label>
  </Labels>
  <Description>
Wall W-FS-001 (GUID: 3MVDFw6f90Nfh9rP1dl_k5) fails NBC 2020 Clause 9.10.9.16.(3).
REQUIREMENT: Fire separation between dwelling units shall have FRR ≥ 1 hour.
ACTUAL: 0.5 hour | REQUIRED: ≥ 1.0 hour
CONTEXT: Wall separates Dwelling Unit A from Dwelling Unit B.
ACTION REQUIRED:
- Revise wall assembly to achieve 1-hour fire-resistance rating
- Specify CCMC-evaluated assembly (e.g., CCMC-14065-R)
- Reference NMS Section 09 29 00 for compliant assemblies
  </Description>
  <BimSnippet SnippetType="IFC4">
   <Reference>3MVDFw6f90Nfh9rP1dl_k5</Reference>
   <ReferenceSchema>IFC4</ReferenceSchema>
  </BimSnippet>
  <DocumentReference Guid="doc-001" isExternal="true">
   <ReferencedDocument>https://codes.nrc.gc.ca/nbc2020/9.10.9.6</ReferencedDocument>
   <Description>NBC 2020 Clause 9.10.9.16</Description>
  </DocumentReference>
  <DocumentReference Guid="doc-002" isExternal="true">
   <ReferencedDocument>https://nms.nrc.gc.ca/section/09-29-00</ReferencedDocument>
   <Description>NMS Section 09 29 00 - Gypsum Board</Description>
  </DocumentReference>
  <DocumentReference Guid="doc-003" isExternal="true">
   <ReferencedDocument>https://ccmc.nrc.gc.ca/report/14065-R</ReferencedDocument>
   <Description>CCMC XXXXX-R - Compliant Type X Gypsum Assembly</Description>
  </DocumentReference>
 </Topic>
</Markup>
```

7. Conclusion and Next Steps

This report has proposed a blueprint for the digitalization of NRC's key resources: the NMCC, the NMS and the CCMC Material database. It addresses the foundational requirements necessary to move from document-centric, human-readable formats to structured, machine-interpretable data assets. Through a detailed audit of the current state of digitalization, this report has identified that while all three resources possess well-defined organizational structures, they remain largely siloed and operate at varying levels of digital maturity. The NMCC, available primarily in PDF format, exhibits high complexity but limited machine-interpretability. The NMS, transitioning from traditional document formats, shows promise with ongoing digitalization efforts but requires substantial work to achieve full machine-interpretability. The CCMC database, having implemented a JSON API for metadata, demonstrates the most advanced digitalization progress, though its detailed technical content remains locked in unstructured PDF documents.

NRC's digitalization strategy, implemented through the DCP and anchored by the CDF, provides a structured pathway to digitize its resources and making them available to industry stakeholders. By developing data models for each resource, including use cases, activity diagrams, and detailed entity-relationship structures, this report has laid the groundwork for creating semantic linkages between codes, specifications, and materials. The consolidated data model demonstrates how these three distinct domains can be unified through shared metadata and common concepts, particularly around building elements, properties, and requirements, while preserving the unique organizational logic of each resource.

Critical to the success of this initiative is adherence to machine-interpretable format requirements that ensure data is not only digitized but computable. The adoption of open, standards-based formats, such as XML, JSON, RDF, and IFC, combined with robust schema validation, explicit semantic definitions through ontologies, and persistent unique identifiers, will enable the automated compliance checking, enhanced interoperability, and improved data management practices envisioned in the NRC DCP.

The development of the NRC Data Dictionary should be maintained as a high-priority foundational activity. This reference will serve as the authoritative source of truth, providing explicit definitions for every term, property, and relationship across all three resources. Through curation, governance, and alignment with international standards and initiatives such as the buildingSMART Data Dictionary (bSDD) and ISO 16739-1, the data dictionary will establish the semantic backbone necessary for true machine-interpretability and cross-platform interoperability.

7.1. Next Steps:

To continue the development of the NRC DCP, and in line with the suggested course of action laid out int NRC DCP R&D roadmap, the following activities should be prioritized:

1. **Establish Governance Structure** (activities 3.1.01, 3.3.02, 3.3.03, and 3.3.04): If not already done, form a steering committee as well as technical and content working groups with representatives from NMCC, NMS, CCMC, and NRC IT services to oversee the project's implementation, ensuring alignment with the PDCSS objectives and industry needs. Develop project charter and risk management and mitigation strategies. As part of this work, the editorial board for the NRC Data Dictionary should be identified and mobilized as indicated in section 3.
2. **Continue Preparatory and Foundational Work**
    a. Digital Content Development (activities 1.1.03, 1.1.04, 1.1.05, 1.1.07, 1.1.08, 1.1.09, 1.1.10, 1.1.12, 1.1.13, 1.1.14): Continue developing the structure, standardized formats, and begin digitalization of content. Focus on the development of the NRC data dictionary and metadata repositories.
    b. Supporting resources (activities 1.2.01 to 1.2.04): Confirm semantic and syntactic resources for the CDF in line with the NRC data dictionary and metadata repository
    c. Digital process formalization (activities 1.3.01, 1.3.02 and 1.3.03): Confirm and refine digital workflows and processes supporting content development, curation and distribution.
3. **Technical Infrastructure Development**
    a. Digital functions and services (activities 2.1.01, 2.1.07 and 2.1.08): Confirm/develop functional requirements for DCP for all three resources.
    b. Repositories and integrations (activities 2.2.03, 2.3.01, and 2.3.02): Initiate the design and implementation of the core digital infrastructure, including data repositories, API frameworks, and integration mechanisms.
4. **Stakeholder Engagement** (activity 3.3.01): Develop and implement a stakeholder engagement plan to ensure buy-in and collaboration from all relevant parties throughout the development process.
5. **Pilot Projects**: Initiate pilot projects for key components of the DCP, particularly in areas such as digital code compliance checking and e-permitting, to test concepts and gather early feedback.

8. References

buildingSMART International. (n.d.). *buildingSMART data dictionary (bSDD)*. https://www.buildingsmart.org

Centre for Digital Built Britain. (n.d.). https://www.cdbb.cam.ac.uk

Construction Specifications Institute. (n.d.). *MasterFormat*. https://www.csiresources.org

Construction Specifications Institute. (n.d.). *UniFormat*. https://www.csiresources.org

ECMA International. (2017). *The JSON data interchange format (ECMA-404)*. https://www.ecma-international.org

European Network for Digital Building Permit. (n.d.). https://www.eunet4dbp.eu

International Organization for Standardization. (n.d.). *ISO standards*. https://www.iso.org

ISO. (2007). ISO 12006-3:2007 Building construction—Organization of information about construction works—Part 3: Framework for object-oriented information. https://www.iso.org/standard/38706.html

ISO. (2015). ISO 12006-2:2015 Building construction—Organization of information about construction works Part 2: Framework for classification. https://www.iso.org/standard/61753.html

ISO. (2024). ISO 16739-1:2024 Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries—Part 1: Data schema. https://www.iso.org/standard/84123.html

ISO. (2020). ISO 23386:2020 Building information modelling and other digital processes used in construction—Methodology to describe, author and maintain properties in interconnected data dictionaries. https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/07/54/75401.html

ISO. (2025a). ISO 23387:2025 Building information modelling (BIM)—Data templates for objects used in the life cycle of assets. https://www.iso.org/standard/85391.html

ISO. (2025b). ISO 29481-1:2025 Building information models—Information delivery manual Part 1: Methodology and format. https://www.iso.org/standard/88515.html

National Research Council of Canada. (2023, May 1). *Platform to Decarbonize the Construction Sector at Scale*. https://nrc.canada.ca/en/research-development/research-collaboration/platform-decarbonize-construction-sector-scale

National Research Council of Canada. (n.d.). *Canadian construction materials centre (CCMC) registry*. Government of Canada. https://nrc.canada.ca

National Research Council of Canada. (n.d.). *National master specifications*. Government of Canada. https://nrc.canada.ca

National Research Council of Canada. (n.d.). *National model construction codes*. Government of Canada. https://nrc.canada.ca

Plan Transition Numérique dans le Bâtiment. (n.d.). https://www.ptnb.fr

Poirier, E. A., Cozzitorto, C., Davari, S., Rezaei, N., Naderi, E., Naghshzan, Z., Grenier, M., Minicola, D., & Jalaei, F. (2024). *Towards a Digital Construction Platform at the National Research Council of Canada: Digitalization of National Construction Codes and Development of a National Common Data Framework* (p. 377). National Research Council of Canada. https://flipbooks.buildingsmartcanada.ca/d0f2fe952c.html

World Wide Web Consortium (W3C). (2008). *Extensible markup language (XML)*. https://www.w3.org

World Wide Web Consortium (W3C). (2012). *OWL 2 web ontology language document overview*. https://www.w3.org

World Wide Web Consortium (W3C). (2013). *SPARQL 1.1 overview*. https://www.w3.org

World Wide Web Consortium (W3C). (2014). *Resource description framework (RDF)*. https://www.w3.org

World Wide Web Consortium (W3C). (2017). *Shapes constraint language (SHACL)*. https://www.w3.org