Ontology Selection and Implementation Guide for Semantic Experts - Part 1

1. Selected Ontologies

The following ontologies provide the optimal balance of adoption and expressiveness while minimizing proliferation:

Ontology	Namespace	Adoption Rate	Primary Use
Dublin Core Terms	`dcterms:`	60-70%	Document metadata, structure
Schema.org	`schema:`	35-45%	General entities, content relationships
SKOS	`skos:`	15-20%	Concept definition, semantic relationships
CiTO	`cito:`	15-20%	Scholarly/evidential relationships

Standard JSON-LD context with properly defined namespaces:

```
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "skos": "http://www.w3.org/2004/02/skos/core#",
    "cito": "http://purl.org/spar/cito/",
    "ex": "http://example.org/"
  }
}
```

> **Note on instance identifiers**: Throughout this guide, we use `ex:` as the prefix for instance identifiers (e.g., `ex:document-1`, `ex:concept-machine-learning`). In a real implementation, you would replace this with your own controlled URI namespace.

2. Entity Type Mapping

Map entities to these classes to maximize interoperability:

2.1 Document Entities

Entity Type	Class	Rationale
Document	`dcterms:Text`	Highest adoption (60-70%) for document representation
Document Section	`dcterms:Text` + `dcterms:isPartOf`	Maintains cohesion with document parent
Document Collection	`dcterms:Collection`	Standard for document groups
Article	`dcterms:Text`	For consistency with other document types
Report	`dcterms:Text`	General report type
Web Page	`dcterms:Text`	Web document for consistency
Message/Comment	`dcterms:Text`	Message as document entity for consistency
Conversation	`dcterms:Collection`	Collection of message documents

2.2 Conceptual Entities

Entity Type	Class	Rationale
Concept	`skos:Concept`	Specifically designed for concept representation
Term	`skos:Concept`	Terms as lexical embodiments of concepts

Laurent-Philippe Albou - March 25th, 2025 - Semantic recommendations
laurent-philippe.albou@roche.com

Entity Type	Class	Rationale
Topic	`skos:Concept`	Subject areas as concepts
Category	`skos:Concept`	Classification concepts
Claim	`cito:Claim`	Assertion that can be supported/disputed

2.3 Agent Entities

Entity Type	Class	Rationale
Person	`schema:Person`	Highest adoption (40-45%) for person entities
Organization	`schema:Organization`	Standard for organizational entities
Software Agent	`schema:SoftwareApplication`	Digital agents/applications

2.4 Content Entities

Entity Type	Class	Rationale
Dataset	`schema:Dataset`	Data collections/sets
Image	`schema:ImageObject`	Visual content
Video	`schema:VideoObject`	Video content
Table	`schema:Table`	Tabular data
Code	`schema:SoftwareSourceCode`	Programming code

3. Property Mapping

3.1 Document Properties

Property	Term	Ontology	Notes
Identifier	`dcterms:identifier`	Dublin Core	Unique identifier
Title	`dcterms:title`	Dublin Core	Document title
Description	`dcterms:description`	Dublin Core	Document description
Creation Date	`dcterms:created`	Dublin Core	When document was created
Modification Date	`dcterms:modified`	Dublin Core	When document was modified
Publisher	`dcterms:publisher`	Dublin Core	Publishing entity
Creator	`dcterms:creator`	Dublin Core	Document author/creator
Contributor	`dcterms:contributor`	Dublin Core	Secondary contributor
Format	`dcterms:format`	Dublin Core	Document format
Language	`dcterms:language`	Dublin Core	Document language
Rights	`dcterms:rights`	Dublin Core	Rights statement
License	`dcterms:license`	Dublin Core	License information
Subject	`dcterms:subject`	Dublin Core	Document topic
Abstract	`dcterms:abstract`	Dublin Core	Document summary

3.2 Structural Relationships

Relationship	Term	Ontology	Notes
Is part of	`dcterms:isPartOf`	Dublin Core	Child to parent relationship
Has part	`dcterms:hasPart`	Dublin Core	Parent to child relationship

Relationship	Term	Ontology	Notes
References	`dcterms:references`	Dublin Core	General reference relationship
Is referenced by	`dcterms:isReferencedBy`	Dublin Core	Inverse of references
Has version	`dcterms:hasVersion`	Dublin Core	Version relationship
Is version of	`dcterms:isVersionOf`	Dublin Core	Inverse of has version
Replaces	`dcterms:replaces`	Dublin Core	Superseding relationship
Is replaced by	`dcterms:isReplacedBy`	Dublin Core	Inverse of replaces
Requires	`dcterms:requires`	Dublin Core	Dependency relationship
Is required by	`dcterms:isRequiredBy`	Dublin Core	Inverse of requires
Conforms to	`dcterms:conformsTo`	Dublin Core	Standard compliance

3.3 Sequential Relationships

Relationship	Term	Ontology	Notes
Next item	`schema:nextItem`	Schema.org	Points to next item in sequence
Previous item	`schema:previousItem`	Schema.org	Points to previous item in sequence

3.4 Content Relationships

Relationship	Term	Ontology	Notes
Mentions	`schema:mentions`	Schema.org	References an entity
Is mentioned in	`schema:mentionedIn`	Schema.org	Inverse of mentions
About	`schema:about`	Schema.org	Primary topic
Describes	`schema:describes`	Schema.org	Provides description
Illustrates	`schema:illustrates`	Schema.org	Provides visual representation
Is illustrated by	`schema:illustration`	Schema.org	Inverse of illustrates
Explains	`schema:explains`	Schema.org	Provides explanation
Is explained by	`schema:explainingDescription`	Schema.org	Inverse of explains

3.5 Evidential Relationships

Relationship	Term	Ontology	Notes
Supports	`cito:supports`	CiTO	Provides supporting evidence
Is supported by	`cito:isSupportedBy`	CiTO	Inverse of supports
Disagrees with	`cito:disagreesWith`	CiTO	Contradicts
Is disagreed with by	`cito:isDisagreedWithBy`	CiTO	Inverse of disagrees with
Uses data from	`cito:usesDataFrom`	CiTO	Data source relationship
Provides data for	`cito:providesDataFor`	CiTO	Inverse of uses data from
Extends	`cito:extends`	CiTO	Builds upon
Is extended by	`cito:isExtendedBy`	CiTO	Inverse of extends
Discusses	`cito:discusses`	CiTO	Substantive discussion
Is discussed by	`cito:isDiscussedBy`	CiTO	Inverse of discusses
Confirms	`cito:confirms`	CiTO	Verifies

Relationship	Term	Ontology	Notes
Is confirmed by	`cito:isConfirmedBy`	CiTO	Inverse of confirms

3.6 Concept Relationships

Relationship	Term	Ontology	Notes
Broader	`skos:broader`	SKOS	Hierarchical parent concept
Narrower	`skos:narrower`	SKOS	Hierarchical child concept
Related	`skos:related`	SKOS	Associated concept
Exact match	`skos:exactMatch`	SKOS	Identical concept
Close match	`skos:closeMatch`	SKOS	Similar concept
Definition	`skos:definition`	SKOS	Formal definition
Preferred label	`skos:prefLabel`	SKOS	Primary term
Alternative label	`skos:altLabel`	SKOS	Synonym
Same as	`schema:sameAs`	Schema.org	Identity relationship
Opposite of	`schema:oppositeOf`	Schema.org	Contrary relationship

4. Implementation Patterns

4.1 Document with Sections

```
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:report2023",
      "@type": "dcterms:Text",
      "dcterms:identifier": "report2023",
      "dcterms:title": "Annual Report 2023",
      "dcterms:created": "2023-12-15",
      "dcterms:creator": {"@id": "ex:person-john-smith"},
      "dcterms:description": "Annual company performance report",
      "dcterms:hasPart": [
        {"@id": "ex:report2023-section1"},
        {"@id": "ex:report2023-section2"},
        {"@id": "ex:report2023-section3"}
      ]
    },
    {
      "@id": "ex:report2023-section1",
      "@type": "dcterms:Text",
      "dcterms:title": "Executive Summary",
      "dcterms:isPartOf": {"@id": "ex:report2023"}
    },
    {
      "@id": "ex:report2023-section2",
      "@type": "dcterms:Text",
      "dcterms:title": "Financial Performance",
      "dcterms:isPartOf": {"@id": "ex:report2023"},
      "schema:nextItem": {"@id": "ex:report2023-section3"},
      "schema:previousItem": {"@id": "ex:report2023-section1"}
    },
```

```
    {
      "@id": "ex:report2023-section3",
      "@type": "dcterms:Text",
      "dcterms:title": "Strategic Outlook",
      "dcterms:isPartOf": {"@id": "ex:report2023"},
      "schema:previousItem": {"@id": "ex:report2023-section2"}
    },
    {
      "@id": "ex:person-john-smith",
      "@type": "schema:Person",
      "schema:name": "John Smith",
      "schema:jobTitle": "Chief Financial Officer"
    }
  ]
}
```

4.2 Concept Network with Synonyms

```
{
  "@context": {
    "skos": "http://www.w3.org/2004/02/skos/core#",
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:concept-machine-learning",
      "@type": "skos:Concept",
      "skos:prefLabel": "Machine Learning",
      "skos:definition": "A field of AI that enables systems to learn from data",
      "skos:narrower": [
        {"@id": "ex:concept-supervised-learning"},
        {"@id": "ex:concept-unsupervised-learning"},
        {"@id": "ex:concept-reinforcement-learning"}
      ],
      "skos:related": {"@id": "ex:concept-artificial-intelligence"}
    },
    {
      "@id": "ex:concept-supervised-learning",
      "@type": "skos:Concept",
      "skos:prefLabel": "Supervised Learning",
      "skos:definition": "ML approach using labeled training data",
      "skos:broader": {"@id": "ex:concept-machine-learning"}
    },
    {
      "@id": "ex:concept-unsupervised-learning",
      "@type": "skos:Concept",
      "skos:prefLabel": "Unsupervised Learning",
      "skos:definition": "ML approach using unlabeled training data",
      "skos:broader": {"@id": "ex:concept-machine-learning"}
    },
    {
      "@id": "ex:concept-reinforcement-learning",
      "@type": "skos:Concept",
      "skos:prefLabel": "Reinforcement Learning",
      "skos:altLabel": "RL",
      "skos:definition": "ML approach based on rewards and penalties",
      "skos:broader": {"@id": "ex:concept-machine-learning"}
    },
    {
```

```
      "@id": "ex:concept-deep-learning",
      "@type": "skos:Concept",
      "skos:prefLabel": "Deep Learning",
      "skos:altLabel": "Deep Neural Networks",
      "skos:definition": "Machine learning with multi-layered neural networks",
      "skos:broader": {"@id": "ex:concept-neural-networks"},
      "skos:related": {"@id": "ex:concept-machine-learning"}
    },
    {
      "@id": "ex:concept-neural-networks",
      "@type": "skos:Concept",
      "skos:prefLabel": "Neural Networks",
      "skos:definition": "Computing systems inspired by biological neural networks",
      "skos:narrower": {"@id": "ex:concept-deep-learning"},
      "skos:related": {"@id": "ex:concept-machine-learning"}
    },
    {
      "@id": "ex:concept-artificial-intelligence",
      "@type": "skos:Concept",
      "skos:prefLabel": "Artificial Intelligence",
      "skos:altLabel": ["AI", "Machine Intelligence"],
      "skos:definition": "The simulation of human intelligence in machines",
      "skos:narrower": {"@id": "ex:concept-machine-learning"}
    }
  ]
}
```


4.3 Citation and Evidential Network


```
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "cito": "http://purl.org/spar/cito/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:paper-smith-2023",
      "@type": "dcterms:Text",
      "dcterms:title": "Climate Change Effects on Alpine Ecosystems",
      "dcterms:creator": {"@id": "ex:person-smith"},
      "dcterms:created": "2023-05-12",
      "schema:about": {"@id": "ex:concept-climate-change-impacts"},
      "cito:supports": {"@id": "ex:claim-anthropogenic-warming"},
      "cito:disagreesWith": {"@id": "ex:claim-natural-variation"},
      "cito:usesDataFrom": {"@id": "ex:dataset-alpine-2022"}
    },
    {
      "@id": "ex:claim-anthropogenic-warming",
      "@type": "cito:Claim",
      "schema:name": "Anthropogenic Warming Claim",
      "schema:description": "Human activities are the primary driver of observed climate change",
      "cito:isSupportedBy": [
        {"@id": "ex:paper-smith-2023"},
        {"@id": "ex:paper-jones-2021"}
      ],
      "cito:isDisagreedWithBy": {"@id": "ex:paper-wilson-2022"}
    },
    {
```

```
      "@id": "ex:claim-natural-variation",
      "@type": "cito:Claim",
      "schema:name": "Natural Variation Claim",
      "schema:description": "Observed changes are primarily due to natural climate cycles",
      "cito:isSupportedBy": {"@id": "ex:paper-wilson-2022"},
      "cito:isDisagreedWithBy": [
        {"@id": "ex:paper-smith-2023"},
        {"@id": "ex:paper-jones-2021"}
      ]
    },
    {
      "@id": "ex:paper-jones-2021",
      "@type": "dcterms:Text",
      "dcterms:title": "Greenhouse Gas Attribution in Climate Models",
      "dcterms:creator": {"@id": "ex:person-jones"},
      "dcterms:created": "2021-09-30",
      "cito:supports": {"@id": "ex:claim-anthropogenic-warming"},
      "cito:disagreesWith": {"@id": "ex:claim-natural-variation"},
      "cito:extends": {"@id": "ex:paper-zhang-2020"}
    },
    {
      "@id": "ex:paper-wilson-2022",
      "@type": "dcterms:Text",
      "dcterms:title": "Natural Climate Cycles: A Historical Perspective",
      "dcterms:creator": {"@id": "ex:person-wilson"},
      "dcterms:created": "2022-03-15",
      "cito:supports": {"@id": "ex:claim-natural-variation"},
      "cito:disagreesWith": {"@id": "ex:claim-anthropogenic-warming"}
    },
    {
      "@id": "ex:dataset-alpine-2022",
      "@type": "schema:Dataset",
      "schema:name": "Alpine Ecosystem Observations 2010-2022",
      "schema:description": "Long-term ecological survey data from alpine regions",
      "cito:providesDataFor": {"@id": "ex:paper-smith-2023"}
    },
    {
      "@id": "ex:paper-zhang-2020",
      "@type": "dcterms:Text",
      "dcterms:title": "Methodologies for Climate Attribution Studies",
      "dcterms:creator": {"@id": "ex:person-zhang"},
      "dcterms:created": "2020-11-10",
      "cito:isExtendedBy": {"@id": "ex:paper-jones-2021"}
    }
  ]
}
```

5. Decision Flow for Ontology Selection

When choosing ontologies and terms for a specific use case, follow this decision process:

1. **Identify entity type**: Document, concept, person, etc.
2. **Select primary ontology** based on entity type:
    - Document entities → Dublin Core Terms
    - Concept entities → SKOS
    - General entities → Schema.org
    - Citation/Evidence → CiTO
3. **Select properties** based on relationship type:
    - Document structure → Dublin Core Terms

    - Content relationships → Schema.org
    - Concept relationships → SKOS
    - Evidential relationships → CiTO
4. **Check adoption**: If two terms have similar function, prefer the one with higher adoption rate.
5. **Maintain ontological cohesion**: Use the same ontology for related properties unless there's a significant adoption advantage (>15-20%).

6. URI/IRI Pattern Recommendations

Consistent URI/IRI patterns enhance interoperability:

6.1 Namespace Strategy

Always include a namespace declaration for your instance identifiers:

```
"@context": {
  "dcterms": "http://purl.org/dc/terms/",
  "schema": "https://schema.org/",
  "ex": "http://example.org/",  // Replace with your namespace
  ...
}
```

In production environments: - Replace `http://example.org/` with a namespace you control - Consider using persistent URI schemes (e.g., w3id.org, purl.org) - Ensure consistency across all your semantic data

6.2 URI Pattern Examples

Entity Type	Pattern	Example
Document	`{namespace}{document-type}-{identifier}`	`ex:report-annual-2023`
Document Section	`{namespace}{document-id}-{section-id}`	`ex:report-annual-2023-section1`
Concept	`{namespace}concept-{concept-name}`	`ex:concept-machine-learning`
Person	`{namespace}person-{person-name}`	`ex:person-john-smith`
Organization	`{namespace}org-{org-name}`	`ex:org-acme-corp`
Dataset	`{namespace}dataset-{dataset-id}`	`ex:dataset-climate-2023`
Claim	`{namespace}claim-{claim-name}`	`ex:claim-anthropogenic-warming`

6.3 Best Practices for URI Construction

1. **Use lowercase kebab-case** for identifier components
2. **Keep URIs persistent** - once published, don't change them
3. **Make URIs descriptive** but not overly long
4. **Avoid encoding implementation details** in URIs
5. **Include namespace prefixes** in JSON-LD context
6. **Document your URI patterns** for maintainers and consumers

Ontology Selection and Implementation Guide for Semantic Experts - Part 2

7. Advanced Implementation Patterns

7.1 Representing Uncertainty and Provenance

For assertions with uncertain or multiple sources:

```
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "cito": "http://purl.org/spar/cito/",
    "ex": "http://example.org/"
  },
```

```
  "@graph": [
    {
      "@id": "ex:claim-global-temperature-rise",
      "@type": "cito:Claim",
      "schema:name": "Global Temperature Rise",
      "schema:description": "Global temperatures will rise by 2-4°C by 2100",
      "schema:creativeWorkStatus": "Probabilistic",
      "schema:confidenceLevel": "High",
      "cito:isSupportedBy": [
        {"@id": "ex:report-ipcc-2021"},
        {"@id": "ex:dataset-nasa-climate-2022"}
      ],
      "dcterms:provenance": "Consensus of multiple climate models"
    }
  ]
}
```

7.2 Temporal Aspects of Relationships

For relationships that change over time:

```
{
  "@context": {
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:role-company-ceo",
      "@type": "schema:Role",
      "schema:roleName": "CEO",
      "schema:startDate": "2018-05-01",
      "schema:endDate": "2023-06-30",
      "schema:member": {"@id": "ex:person-jane-doe"},
      "schema:memberOf": {"@id": "ex:org-acme-corp"}
    }
  ]
}
```

7.3 Weighted Relationships

For relationships with variable strength or relevance:

```
{
  "@context": {
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:assessment-concept-relevance",
      "@type": "schema:AssessAction",
      "schema:agent": {"@id": "ex:algorithm-topic-extraction"},
      "schema:object": {"@id": "ex:concept-artificial-intelligence"},
      "schema:target": {"@id": "ex:paper-research-2023"},
      "schema:result": {
        "@type": "schema:PropertyValue",
        "schema:name": "relevanceScore",
        "schema:value": 0.87
      }
    }
```

```
  ]
}
```

8. Compatibility Considerations

8.1 Schema.org and Dublin Core Mapping

For maximum interoperability, these equivalent properties can be used:

Dublin Core	Schema.org	Notes
dcterms:title	schema:name	Document title
dcterms:description	schema:description	Description
dcterms:creator	schema:creator / schema:author	Creator/author
dcterms:publisher	schema:publisher	Publisher
dcterms:created	schema:dateCreated	Creation date
dcterms:subject	schema:about	Subject/topic
dcterms:language	schema:inLanguage	Document language
dcterms:format	schema:encodingFormat	Document format
dcterms:identifier	schema:identifier	Unique identifier
dcterms:rights	schema:license	Rights information
dcterms:isPartOf	schema:isPartOf	Part-whole relationship
dcterms:hasPart	schema:hasPart	Part-whole relationship
dcterms:references	schema:citation	Reference relationship

Consider dual property assertions for maximum compatibility in cross-system contexts.

8.2 Integration with External Knowledge Bases

For linking to external knowledge bases:

```json
{
  "@context": {
    "skos": "http://www.w3.org/2004/02/skos/core#",
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:concept-artificial-intelligence",
      "@type": "skos:Concept",
      "skos:prefLabel": "Artificial Intelligence",
      "skos:exactMatch": [
        {"@id": "http://dbpedia.org/resource/Artificial_intelligence"},
        {"@id": "http://www.wikidata.org/entity/Q11660"}
      ],
      "schema:sameAs": "http://dbpedia.org/resource/Artificial_intelligence"
    }
  ]
}
```

8.3 Open vs. Closed World Assumptions

Semantic web typically operates under the Open World Assumption: - Absence of information doesn't imply negation - New assertions can be added without contradicting existing ones - Consider explicitly stating negative relationships where needed

9. Best Practices in Implementation

9.1 Maintaining Clean Semantics

1. **Prioritize semantic accuracy over visualization needs**
   - Semantic structure exists to represent meaning, not display requirements

   - Visualization tools should adapt to semantic structure, not vice versa
2. **Use the minimal set of relationships** that adequately represent meaning
   - Avoid redundant relationships
   - Don't create relationships solely for visual display purposes
3. **Preserve namespace prefixes** in all contexts
   - Essential for rebuilding complete URIs
   - Critical for long-term interoperability

9.2 Enhancing Querying Capabilities

Structure your semantic model to support these query types:

1. **Entity retrieval**: "Find all documents by Author X"
2. **Concept exploration**: "What concepts are related to Machine Learning?"
3. **Claim analysis**: "What evidence supports Claim Y?"
4. **Content queries**: "What documents discuss Concept Z in relation to Concept W?"
5. **Temporal questions**: "How did the understanding of Concept A evolve over time?"

9.3 Validation Best Practices

1. **Validate against common semantic patterns**
   - Use JSON-LD Playground for basic validation
   - Check property domains and ranges when possible
2. **Check for common errors**
   - Missing required properties
   - Incorrect property values
   - Inconsistent relationship directionality
   - URI construction errors
3. **Verify contextual integrity**
   - Ensure all referenced entities exist
   - Check for orphaned entities
   - Verify bidirectional relationships match
4. **Validate namespace declarations**
   - Ensure all prefixes used in `@id` values are defined in `@context`
   - Check that namespaces resolve to valid URIs

10. Performance Considerations

10.1 Optimization Techniques

1. **Minimize redundant assertions**
   - Avoid repeating the same relationship in multiple directions
   - Use inference capabilities where available
2. **Limit relationship depth**
   - Deep hierarchical relationships can impact query performance
   - Consider flattening extremely deep hierarchies
3. **Use appropriate identifier strategies**
   - Short, opaque identifiers generally perform better
   - Consider hash-based identifiers for large-scale systems

10.2 Scaling Considerations

For large semantic graphs:

1. **Consider partitioning strategies**
   - By domain
   - By time period
   - By relationship type
2. **Use named graphs for context scoping**
   - Separate assertions by source
   - Enable selective processing
3. **Implement lazy loading patterns**

- Load core semantics first
- Defer loading of extended relationships

11. Usage Scenarios and Best Ontology Choices

11.1 Document Management Systems

Primary Ontologies: Dublin Core Terms, Schema.org **Key Entity Types**: - Document (dcterms:Text) - Section (dcterms:Text) - Person (schema:Person)

Key Relationships: - isPartOf/hasPart (dcterms:isPartOf/dcterms:hasPart) - creator (dcterms:creator) - references (dcterms:references)

Example Implementation:

```json
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:doc-policy-2023",
      "@type": "dcterms:Text",
      "dcterms:title": "Corporate Policy Document",
      "dcterms:creator": {"@id": "ex:person-jane-smith"},
      "dcterms:created": "2023-01-15",
      "dcterms:modified": "2023-03-20",
      "dcterms:hasPart": [
        {"@id": "ex:doc-policy-2023-section1"},
        {"@id": "ex:doc-policy-2023-section2"}
      ]
    }
  ]
}
```

11.2 Knowledge Organization Systems

Primary Ontologies: SKOS, Schema.org **Key Entity Types**: - Concept (skos:Concept) - Category (skos:Concept) - Term (skos:Concept)

Key Relationships: - broader/narrower (skos:broader/skos:narrower) - related (skos:related) - exactMatch/closeMatch (skos:exactMatch/skos:closeMatch)

Example Implementation:

```json
{
  "@context": {
    "skos": "http://www.w3.org/2004/02/skos/core#",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:concept-information-security",
      "@type": "skos:Concept",
      "skos:prefLabel": "Information Security",
      "skos:narrower": [
        {"@id": "ex:concept-encryption"},
        {"@id": "ex:concept-access-control"},
        {"@id": "ex:concept-vulnerability-management"}
      ]
    }
```

```
  ]
}
```

11.3 Research Data Management

Primary Ontologies: Dublin Core, Schema.org, CiTO **Key Entity Types**: - Document (dcterms:Text) - Dataset (schema:Dataset) - Person (schema:Person) - Claim (cito:Claim)

Key Relationships: - creator (dcterms:creator) - usesDataFrom/providesDataFor (cito:usesDataFrom/cito:providesDataFor) - supports/isSupportedBy (cito:supports/cito:isSupportedBy)

Example Implementation:

```
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "cito": "http://purl.org/spar/cito/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:dataset-clinical-trial-123",
      "@type": "schema:Dataset",
      "schema:name": "Clinical Trial 123 Results",
      "schema:creator": {"@id": "ex:team-research-alpha"},
      "cito:providesDataFor": {"@id": "ex:paper-research-2023"}
    },
    {
      "@id": "ex:paper-research-2023",
      "@type": "dcterms:Text",
      "dcterms:title": "Efficacy of Treatment X",
      "dcterms:creator": {"@id": "ex:team-research-alpha"},
      "cito:usesDataFrom": {"@id": "ex:dataset-clinical-trial-123"}
    }
  ]
}
```

11.4 Conversation/Discussion Archives

Primary Ontologies: Dublin Core Terms, Schema.org **Key Entity Types**: - Conversation (dcterms:Collection) - Message (dcterms:Text) - Person (schema:Person)

Key Relationships: - isPartOf/hasPart (dcterms:isPartOf/dcterms:hasPart) - creator (dcterms:creator) - nextItem/previousItem (schema:nextItem/schema:previousItem)

Example Implementation:

```
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:conversation-thread-12345",
      "@type": "dcterms:Collection",
      "dcterms:title": "Discussion on Project Timeline",
      "dcterms:hasPart": [
        {"@id": "ex:message-post-1"},
        {"@id": "ex:message-post-2"}
      ]
    },
```

```
    {
      "@id": "ex:message-post-1",
      "@type": "dcterms:Text",
      "dcterms:title": "Initial question",
      "dcterms:description": "When can we expect the first milestone to be completed?",
      "dcterms:creator": {"@id": "ex:person-manager"},
      "dcterms:isPartOf": {"@id": "ex:conversation-thread-12345"},
      "schema:nextItem": {"@id": "ex:message-post-2"}
    },
    {
      "@id": "ex:message-post-2",
      "@type": "dcterms:Text",
      "dcterms:title": "Response",
      "dcterms:description": "We expect to complete the first milestone by June 15th.",
      "dcterms:creator": {"@id": "ex:person-developer"},
      "dcterms:isPartOf": {"@id": "ex:conversation-thread-12345"},
      "schema:previousItem": {"@id": "ex:message-post-1"}
    }
  ]
}
```

12. Final Decision Checklist

When implementing a semantic model, verify it meets these criteria:

1. **Ontological Minimalism**
   - Uses the smallest set of ontologies needed
   - Avoids redundant ontologies for the same concepts
2. **Adoption Maximization**
   - Prioritizes widely-adopted ontologies
   - Uses high-adoption terms for maximum interoperability
3. **Expressiveness Balance**
   - Contains sufficient terms to capture semantics
   - Avoids overly complex or specialized terms when simpler ones suffice
4. **Ontological Cohesion**
   - Maintains consistent ontology usage for related properties
   - Only breaks cohesion when adoption difference exceeds 15-20%
5. **Proper URI Construction**
   - Uses consistent patterns for different entity types
   - Maintains namespace prefixes for URI reconstruction
6. **Relationship Completeness**
   - Captures all meaningful relationships
   - Includes both structural and conceptual relationships
7. **Separation of Concerns**
   - Keeps semantic structure independent of visualization needs
   - Maintains focus on meaning, not display requirements
8. **FAIR Principles Support**
   - Enables Findability, Accessibility, Interoperability, and Reusability
   - Supports meaningful content exploration beyond basic structure
9. **Consistency Across Types**
   - Uses the same ontology for similar entity types
   - Applies Dublin Core Terms consistently for all document-like entities
   - Applies SKOS consistently for all concept-like entities
10. **Complete Namespace Declarations**
    - Ensures all prefixes used in `@id` values are defined in `@context`
    - Uses proper URI prefixes for instance identifiers

13. Common Pitfalls and Solutions

13.1 Semantic Modeling Mistakes

Pitfall	Solution
Missing namespace declarations	Always include complete `@context` with all prefixes used in `@id` values
Mixing ontologies unnecessarily	Follow the ontological cohesion principle
Choosing specialized over common terms	Prefer higher adoption terms unless specialized ones add significant value
Creating redundant relationships	Define core relationships and avoid duplication
Inconsistent entity typing	Maintain consistent types for similar entities (e.g., all document entities use dcterms:Text)

13.2 Implementation Challenges

Challenge	Solution
Overcomplex relationship models	Start with core relationships and add detail incrementally
Poor scalability with large datasets	Use efficient storage and partitioning strategies
Difficulty integrating with legacy systems	Provide interoperability mappings to common schemas
Inconsistent URI patterns	Establish and document a clear URI strategy
Incomplete bidirectional relationships	Use inference rules or explicit assertions for inverse relationships

14. Future Considerations

When designing semantic models, consider these emerging trends:

14.1 Emerging Standards

- **Schema.org extensions**: Monitor for new vocabulary additions relevant to your domain
- **Domain-specific ontologies**: Evaluate their adoption before incorporating
- **W3C standards evolution**: Follow updates to core semantic web standards

14.2 AI and Semantic Web Integration

- **Large Language Models**: Consider how semantic structure can enhance LLM understanding
- **Automatic metadata extraction**: Leverage AI for generating semantic annotations
- **Vector embeddings**: Explore hybrid approaches combining symbolic semantics with vector representations

14.3 Sustainability Considerations

- **Versioning strategies**: Plan for ontology evolution
- **Backward compatibility**: Ensure changes don't break existing semantic links
- **Documentation**: Maintain clear documentation of semantic design decisions

15. Working with JSON-LD Context and Namespaces

15.1 Complete Context Example

Always include a complete context with definitions for all prefixes:

```json
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "skos": "http://www.w3.org/2004/02/skos/core#",
    "cito": "http://purl.org/spar/cito/",
    "ex": "http://example.org/"
  }
}
```

15.2 URI Expansion

JSON-LD processors expand CURIEs (Compact URIs) using the context:

CURIE in JSON-LD	Expanded URI
`ex:concept-machine-learning`	`http://example.org/concept-machine-learning`
`dcterms:title`	`http://purl.org/dc/terms/title`
`schema:Person`	`https://schema.org/Person`

15.3 Namespace Management Best Practices

1. **Use consistent prefixes** across your organization
2. **Document prefix mappings** in a central location
3. **Avoid ad-hoc prefix creation** - reuse established prefixes
4. **Consider using persistent URLs** for your namespace (e.g., w3id.org)
5. **Include version information** in namespaces for evolving models

16. Additional Implementation Examples

16.1 Educational Resources and Learning Objects

```json
{
  "@context": {
    "dcterms": "http://purl.org/dc/terms/",
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:course-data-science-101",
      "@type": "schema:Course",
      "schema:name": "Introduction to Data Science",
      "schema:description": "Fundamentals of data science and analytics",
      "schema:provider": {"@id": "ex:org-university"},
      "dcterms:hasPart": [
        {"@id": "ex:module-python-basics"},
        {"@id": "ex:module-data-analysis"}
      ]
    },
    {
      "@id": "ex:module-python-basics",
      "@type": "dcterms:Text",
      "dcterms:title": "Python Programming Basics",
      "dcterms:isPartOf": {"@id": "ex:course-data-science-101"},
      "schema:teaches": {"@id": "ex:concept-python-programming"}
    }
  ]
}
```

16.2 Event and Temporal Data

```json
{
  "@context": {
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:event-conference-2023",
      "@type": "schema:Event",
```

```
      "schema:name": "International Semantic Web Conference 2023",
      "schema:startDate": "2023-11-06T09:00:00Z",
      "schema:endDate": "2023-11-10T17:00:00Z",
      "schema:location": {"@id": "ex:venue-convention-center"},
      "schema:organizer": {"@id": "ex:org-semantic-web-society"},
      "schema:subEvent": [
        {"@id": "ex:event-keynote-1"},
        {"@id": "ex:event-workshop-a"}
      ]
    }
  ]
}
```

16.3 Product and Service Description

```
{
  "@context": {
    "schema": "https://schema.org/",
    "ex": "http://example.org/"
  },
  "@graph": [
    {
      "@id": "ex:product-semantic-analyzer-pro",
      "@type": "schema:SoftwareApplication",
      "schema:name": "Semantic Analyzer Pro",
      "schema:applicationCategory": "Semantic Web Tool",
      "schema:operatingSystem": ["Windows", "macOS", "Linux"],
      "schema:offers": {
        "@type": "schema:Offer",
        "schema:price": 299.99,
        "schema:priceCurrency": "USD"
      },
      "schema:featureList": [
        "RDF/OWL Processing",
        "SPARQL Endpoint",
        "Knowledge Graph Visualization"
      ]
    }
  ]
}
```

Conclusion

This guide provides a comprehensive framework for selecting ontologies and implementing semantic models. By following these recommendations, you can create semantic representations that balance adoption, expressiveness, and ontological cohesion while ensuring proper handling of namespaces and identifiers. Remember that the goal is to capture meaningful semantic relationships that support sophisticated querying and analysis, while maintaining the clear separation between semantic structure and visualization concerns.