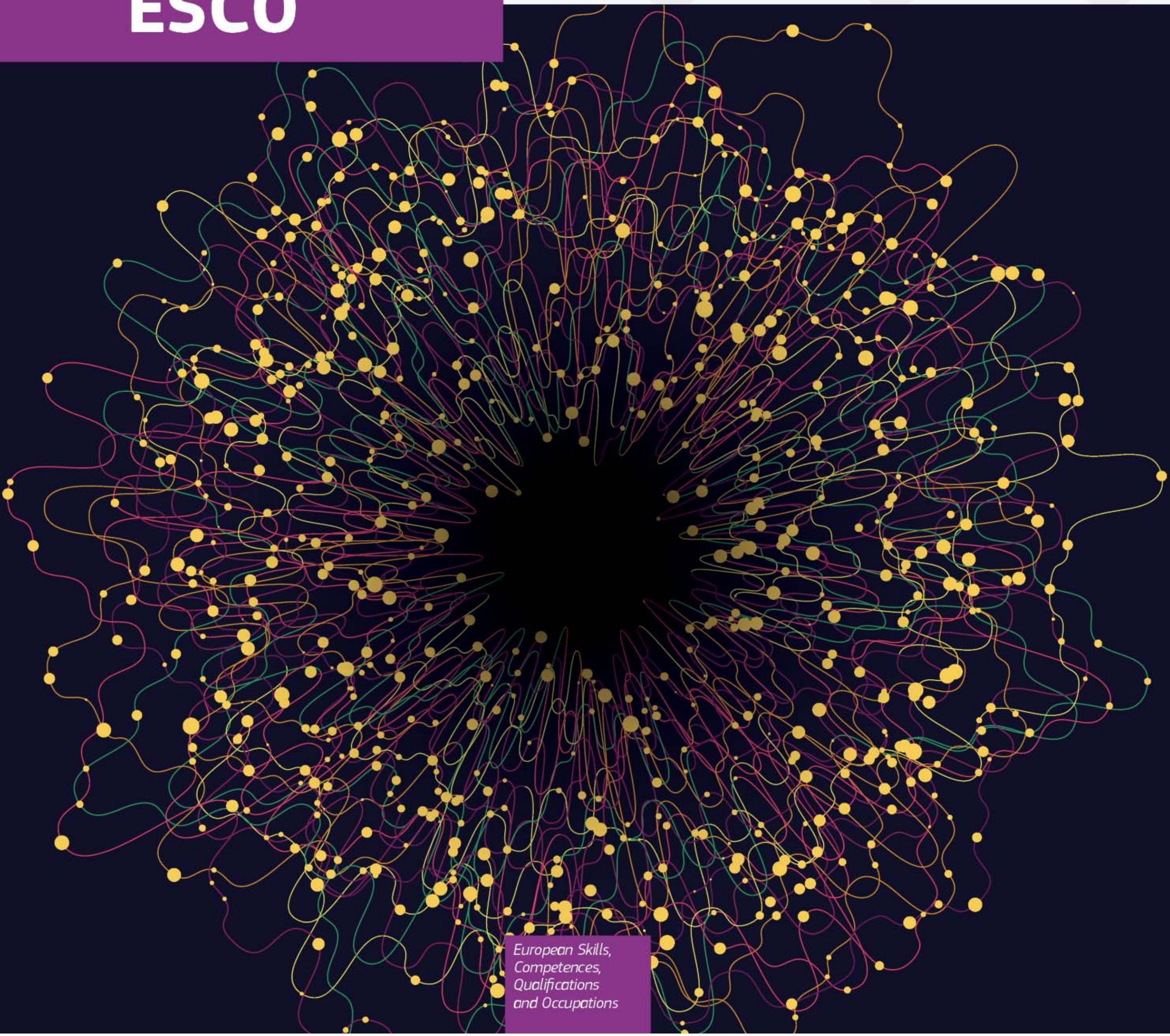
European Commission

# ESCO Service Platform
## Data model

European Skills, Competences, Qualifications and Occupations

ESCO



*European Skills, Competences, Qualifications and Occupations*

# Contents

# 1. The ESCO data model

In general terms, the data model of ESCO is structured on the basis of three pillars, representing a searchable database in 26 languages. These pillars are:

- The occupations pillar
- The knowledge/skills/competences pillar
- The qualifications pillar

These pillars are interlinked in order to show the relationships between them, while occupational profiles show whether skills and competences are essential or optional and what qualifications are relevant for each ESCO occupation. Alternatively, the user can identify a specific skill and see which occupations or qualifications this skill is relates to.

This three-layered approach allows ESCO to organise the available knowledge of the European labour market and the education/training sector in a consistent, transparent and user-friendly way.

**Figure 1: ESCO three pillar structure**

## 1.1 The occupations pillar

Occupations should not be confused with jobs and job titles (which are not covered under ESCO). Their distinction is based on the following definitions:

- **Occupation:** a grouping of jobs involving similar content in terms of tasks, and requiring similar types of skills[1];
- **Job**: a set of tasks and duties executed, or meant to be executed, by one person[2].

[1] Source: Skills base – Labour market information database.

[2] Source: International Labour Organisation, ILO.

***Example:*** *Being the "pilot of a Boeing 747 aircraft for the Paris-New York route" is a job. "Commercial airline pilot" or "civil airline pilot" could be occupations (i.e. groups of jobs, to which this job belongs). Occupations can be used as job titles. An employer recruiting for the above-mentioned position might entitle the vacancy notice with the name of an occupation, e.g. "airline pilot".*

ESCO v1 features 2950 occupations. Each occupation concept defines the content and scope of the occupation, and provides a number of useful metadata[3].

The core element that defines an ESCO occupation is the main idea or understanding of what the occupation is about and how it differs from other occupations. These are captured in the description and scope note.

- A **description** in ESCO is a text field providing a short explanation of the meaning of the occupation and how it should be understood. Most importantly, it clarifies its semantic boundaries. For this reason, a description is always provided for each ESCO occupation.

- A **scope note** in ESCO is sometimes used to make things less ambiguous. It directs the users:

  - to similar occupations that are included in the scope;

  - to alternative occupations that are excluded from it. In this case, it provides hyperlinks to other ESCO occupations.

### *The structure of the occupations pillar*

The occupations in ESCO are structured through their mapping to the International Standard Classification of Occupations (ISCO-08)[4], developed by the International Labour Organisation. The ESCO occupations and their ISCO-08 hierarchy make up the ESCO occupation pillar.

ISCO-08 provides the top four levels, while ESCO occupations provide the fifth and lower levels. Each ESCO occupation is assigned to one ISCO-08 unit group (even if they are not directly related to it, e.g. if they are at level six or seven).

[3] Metadata is data about data: it is structured information that describes, explains, locates or makes it easier to retrieve, use or manage an information resource (NISO 2004: Understanding Metadata. Bethesda, MD: NISO Press).

[4] International Standard Classification of Occupations (ISCO-08).

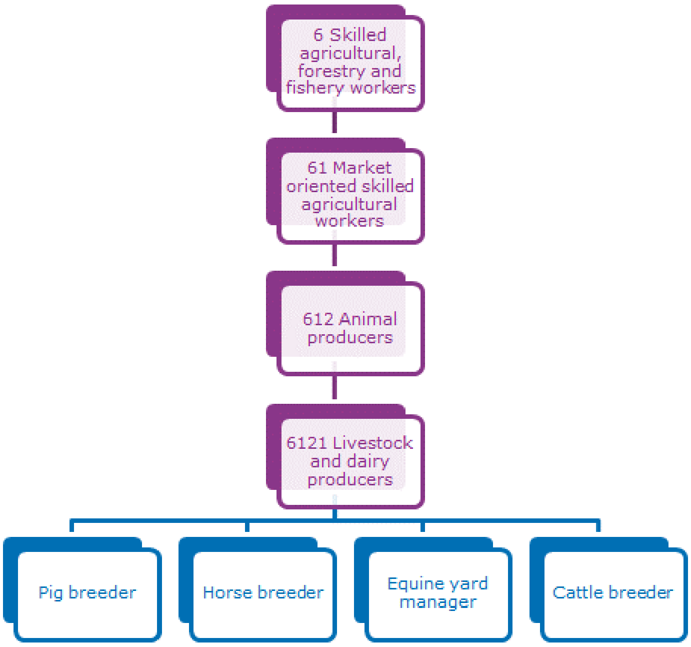

6 Skilled agricultural, forestry and fishery workers
61 Market oriented skilled agricultural workers
612 Animal producers
6121 Livestock and dairy producers
Pig breeder
Horse breeder
Equine yard manager
Cattle breeder

**Figure 2: The structure of the occupations pillar**

### Make sure you understand the following concepts from this section

Occupation, job, the occupations pillar, description, scope note and the occupations pillar structure.

## 1.2 The knowledge, skills and competences pillar

The knowledge, skills and competences pillar, also referred to as the “skills pillar” or “skills and competences pillar”, includes knowledge, skills and competences defined as follows[5]:

- **Knowledge:** The body of facts, principles, theories and practices that is related to a field of work or study. Knowledge is described as theoretical and/or factual, and is the outcome of the assimilation of information through learning.
- **Skill:** The ability to apply knowledge and use know-how to complete tasks and solve problems. Skills are described as cognitive (involving the use of logical, intuitive and creative thinking) or practical (involving manual dexterity and the use of methods, materials, tools and instruments).
- **Competence:** The proven ability to use knowledge, skills and personal, social and/or methodological abilities in work or study situations, and in professional and personal development.

***Example:*** *Working as a "civil airline pilot" requires the competence to combine knowledge about "emergency procedures" and "equipment malfunctions" with skills relating to "reading position coordinates" and "following the flight route".*

### *The content of the skills pillar*

In ESCO v1, the skills pillar contains 13 492 skills concepts. As for the occupations, each one of the concepts defines the content and scope of the skill, and provides a number of useful metadata including the following:

- A preferred term;
- Non-preferred terms;
- Hidden terms;
- An explanation of the concept in the form of a description;
- A scope note;
- The skill type (in the skills pillar, ESCO distinguishes between skill/competence concepts and knowledge concepts);
- A formal definition;
- The relationship between occupations and skills (i.e. the occupations for which the knowledge, skill or competence is essential and the occupations for which the knowledge, skill or competence is optional);

[5] Source: EQF Recommendation: http://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32008H0506(01)&from=EN

- The relationship between knowledge, skills and competences, and other knowledge, skills and competences. This relationship is categorised as essential or optional.

***Example:*** *Procedural law is essential for pleading a case in court; labour law is optional for pleading a case in court.*

- The reusability level, which indicates how widely a knowledge, skill or competence concept can be applied. ESCO distinguishes four levels of skill reusability:
    - **Transversal knowledge, skills and competences** are relevant to a broad range of occupations and sectors. They are often referred to as "core skills", "basic skills" or "soft skills".

      Transversal skills are not usually related directly to occupations in ESCO, except if this link is relevant for their labour market. In fact, they are too abstract to be suitable for job matching. Therefore, they are contextualised. The contextualisation of skills analyses how transversal skills, competences or knowledge are applied in the specific context of a sector or an occupation, and allows knowledge or skill and competence concepts to be created.

***Example:*** *The skill "measure" is too abstract to be linked directly to the "shop assistant" occupation. Measuring is relevant for a large number of occupations and sectors.*

*Through the contextualisation of skills, the skill can be made more specific. A skill named "measure the size of furniture" could, for example, be used in the occupational profile of a specialised furniture seller.*

- **Cross-sector knowledge, skills and competences** are relevant to occupations across several economic sectors.

***Example:*** *"Animal welfare" is relevant in the "agriculture", "veterinary activities" and "amusement and recreation parks" sectors.*

- **Sector-specific knowledge, skills and competences** are specific to one sector, but are relevant for more than one occupation within that sector.

***Example:*** *"Monitoring of livestock" is specific to the sector "husbandry and breeding".*

## The structure of the skills pillar

The ESCO v1 skills pillar does not contain a full hierarchical structure. However, the 13 492 elements of the skills pillar are structured in four different ways:

- Through their relationship with occupations, by using occupational profiles as entry point;
- Transversal knowledge, skills and competences are organised through a skills hierarchy with the following five headings:
  - thinking;
  - language;
  - application of knowledge;
  - social interaction;
  - attitudes and values.
- Through relationships indicating how knowledge, skills and competences are relevant to other knowledge, skills and competences (in particular in cases of the contextualisation of skills);
- Through functional collections that allow the selection of subsets of the skills pillar, according to the function that users have in mind. For example, an organisation may want to use ESCO in an online CV section, where a user indicates his/her language skills. The organisation does not need all the ESCO skills in that section, only the language ones. If a user searches for "Chinese" in this section, the system should suggest "Chinese", "understand spoken Chinese" “understand written Chinese” or "interact verbally in Chinese", but not "traditional Chinese medicine" or "give shiatsu massages". A functional collection would allow the user to pick exactly the skills (or occupations) s/he is looking for.

ESCO v1 includes the following three functional collections:

- Skills from the Digital Competence Framework that have been included in ESCO;
- Language skills;
- Transversal skills.

**Make sure you understand the following concepts from this section**

Knowledge, skill, competence, the skills pillar, transversal, cross-sector, sector-specific, skills pillar structure.

## 1.3 The qualifications pillar

The qualifications pillar aims to collect existing data on qualifications[6]. The final objective is to provide a comprehensive listing of the qualifications that are relevant for the European labour market.

In its qualifications pillar, ESCO will include qualifications following a double approach:

- **Indirect inclusion:** qualifications that Member States include in their national qualifications databases. This information is also available in the "Learning Opportunities and Qualifications in Europe"[7] (LOQ) portal.
- **Direct inclusion:** qualifications that are not part of these databases, but that are also relevant for the European labour market (international qualifications[8]). While an approach for the quality assurance of international qualifications is being discussed at European level, ESCO v1 contains a sample of international qualifications.

In contrast with the occupations pillar and the skills pillar, the qualifications pillar is therefore populated by external sources.

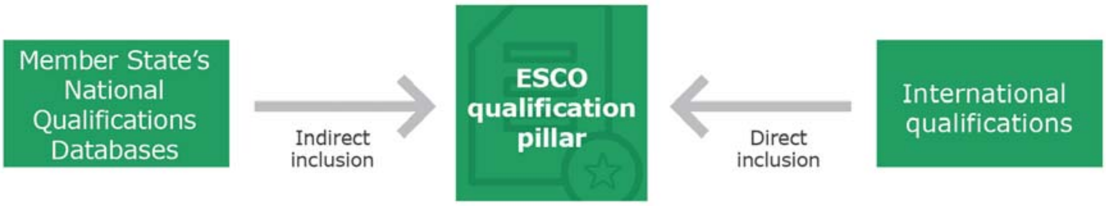

Member State's National Qualifications Databases
Indirect inclusion
ESCO qualification pillar
Direct inclusion
International qualifications

**Figure 3: Qualifications in ESCO**

### *The qualifications metadata schema*

Information can only be comparable throughout Europe if all Member States and other actors have a common view on the attributes that all qualifications share such as the title, the awarding body, the EQF level, the description of the learning outcomes, etc. This common understanding or common language is structured in an agreed 'metadata schema'.

[6] As defined by the European Qualifications Framework Recommendation (EQF), a qualification is the formal outcome of an assessment and validation process which is obtained when a competent body determines that an individual has achieved learning outcomes to given standards (2016).

[7] https://ec.europa.eu/ploteus

[8] As defined by the EQF Advisory Group, an international qualification is a qualification awarded by a legally established international body (association, organisation, sector or company), or by a national body acting on behalf of an international body, that is used in more than one country and that includes learning outcomes assessed with reference to standards established by an international body.

Using a schema for qualifications makes it easier to integrate information with classifications like ESCO, and it enables qualifications to be linked and integrated easily with other structured metadata published on the internet. The current version of the schema (Qualifications Metadata Schema v1.0) is available on the ESCO Service Platform. The qualifications pillar is structured around this schema.

Member States and other stakeholders wishing to publish information on their qualifications in ESCO need to structure their data according to the QMS and upload it in the Qualifications Dataset Register (QDR), a central platform that manages the exchange of qualification data between different stakeholders and European portals. At the time of the writing of this document, the QDR is in its final stages of being deployed at European Commission premises.

## *Linking qualifications to the other pillars*

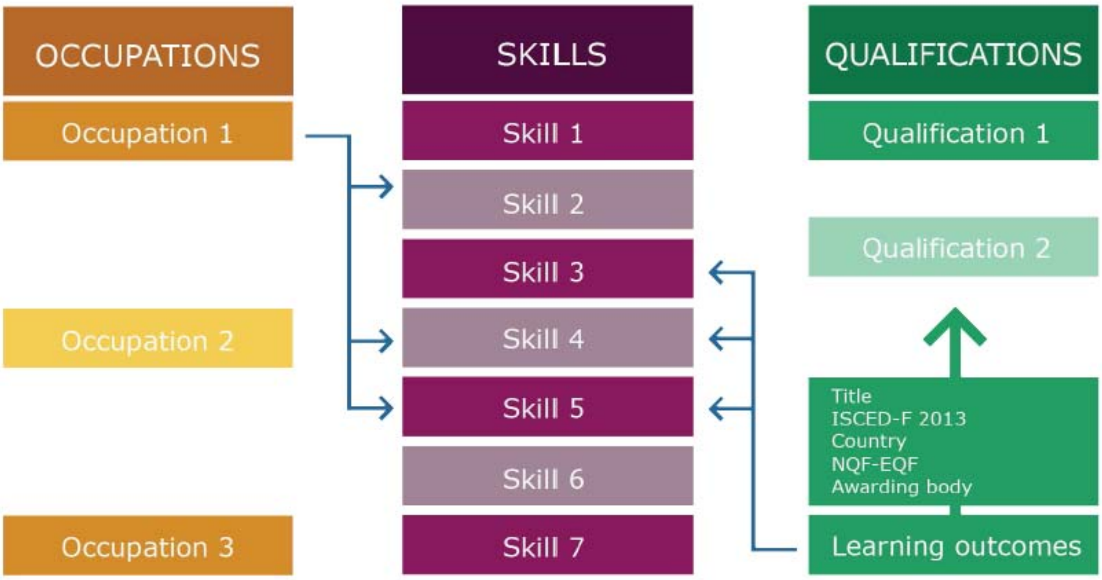


**Figure 4: The link between the three pillar**

### Links with the occupations pillar

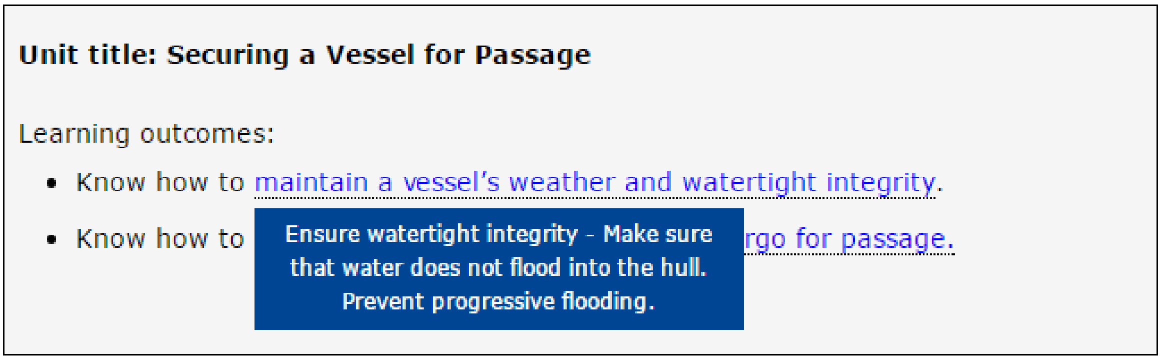
Direct relationships between qualifications in ESCO and the occupations pillar are only displayed if they already exist on a national level. Member States are not developing such data for ESCO. The relationship can indicate, for instance, if a qualification is a requirement in order to work in an occupation in the specific Member State.

Otherwise, the relationship between the occupations and the qualifications is indirect, via the skills pillar.

### Links with the knowledge, skills and competences pillar

Organisations that provide data on qualifications can annotate learning outcome descriptions with ESCO skills terminology. The annotation of qualifications with ESCO skills/competences is one of the most consistent ways of allowing full interoperability between the three pillars: this means adding structured meaning (ESCO skills) to unstructured content (the development of learning outcomes) so that it is machine-readable. To this end, organisations identify knowledge, skills and competence concepts in the skills pillar of ESCO that are relevant in the context of the learning outcomes description of a qualification. Since the semantic annotation is added to the existing format and does not replace it, actors can build on their existing technical infrastructure.

In the example below, the skill “maintain a vessel’s weather and watertight integrity” corresponds to the ESCO skill “ensure watertight integrity”.



**Unit title: Securing a Vessel for Passage**

Learning outcomes:

- Know how to maintain a vessel’s weather and watertight integrity.
- Know how to rgo for passage.

Ensure watertight integrity - Make sure that water does not flood into the hull. Prevent progressive flooding.

**Example of annotation of the “Diploma in maritime studies – sea fishing” qualification, UK-Scotland, Scottish Qualifications Authority. Source: pilot project with the Scottish Qualifications Authority, February 2017.**

The Commission services have piloted this approach in 2017. Technical documentation and a methodology will be available after the publication of ESCO v1.

The following sections describe the ESCO model from a more technical perspective. Full information about the ESCO Data Model Ontology will be published in the third quarter of 2017.

**Make sure you understand the following concepts from this section**

The qualifications pillar, indirect inclusion, direct inclusion, qualifications metadata schema, linking qualifications to other pillars.

# 2. Concepts in ESCO

This section describes the resources that make up the ESCO taxonomy. It begins by giving an overview of what is meant by the word ‘resource’ and ESCO-specific types of resources are then described in more detail.

## 2.1 Linked Open Data concepts

The ESCO taxonomy is published as Linked Open Data, meaning that the data is freely available, that it links to data from other taxonomies, and that other taxonomies can link to ESCO as well. Linked data is often structured according to the RDF (Resource Description Framework) specifications from the W3C[9].

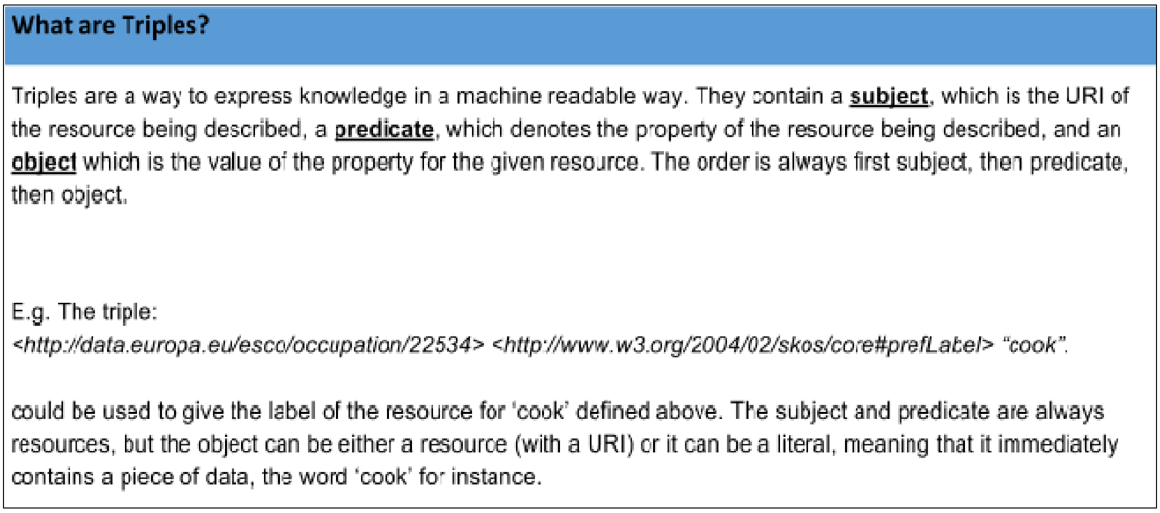
### *Uniform Resource Identifier*

To make linking resources possible, a resource in Linked (Open) Data must be identified using a Uniform Resource Identifier (URI), e.g. http://data.europa.eu/esco/occupation/22534, which could be the URI for the ‘cook’ occupation in ESCO v1.

URIs identify a resource uniquely. They usually take the form of web links which can both humans and machines can follow to find information about an entity from the real world. Humans receive a webpage and machines receive a file in a machine-readable format. A resource in RDF is exactly that – a thing, concept or thought from the real world that is identified by a URI so that knowledge can be expressed about it.

### *Triples*

While URIs identify a resource, the resource also needs to be described. For instance, we need to express that the ‘cook’ entity with the URI above is actually called ‘cook’. In RDF, this is done using ‘triples’. A triple is a piece of data that is made up of three element: a subject, a predicate and an object.



**What are Triples?**

Triples are a way to express knowledge in a machine readable way. They contain a **subject**, which is the URI of the resource being described, a **predicate**, which denotes the property of the resource being described, and an **object** which is the value of the property for the given resource. The order is always first subject, then predicate, then object.

E.g. The triple:
*<http://data.europa.eu/esco/occupation/22534> <http://www.w3.org/2004/02/skos/core#prefLabel> “cook”.*

could be used to give the label of the resource for ‘cook’ defined above. The subject and predicate are always resources, but the object can be either a resource (with a URI) or it can be a literal, meaning that it immediately contains a piece of data, the word ‘cook’ for instance.

[9] See http://www.w3.org/standards/techs/rdf#w3c_all

## Prefixes

The example above illustrates that URIs in RDF can become quite long; this is to ensure they are truly unique. To mitigate this effect somewhat, RDF allows the definition of prefixes to abbreviate the first part of a URI. For instance, in this document, the prefix ‘skos:’ will mean <http://www.w3.org/2004/02/skos/core#> in a URI. So skos:definition is equivalent to the full URI of the definition predicate <http://www.w3.org/2004/02/skos/core#definition>. All prefixes used in this document are listed at the end of the document.

The example also shows that predicates are also URIs. This makes it possible to express information about the properties, such as giving them a human-readable name.

## Links and graphs

The object in a triple can be a URI as well. In this instance, the predicate is used to link two resources together. For instance, say there are two people. If you want to express that they know each other, you could use a triple that has the form: URIperson1 URIknows URIperson2.

By linking resources together in this way, a large collection of interconnected nodes is created. The term ‘graph’ is used for such interlinked collections of triples.

## Types

Resources can be of a certain type. The type of a resource reflects the kind of entity from the real world that should be captured by the linked data resource. Based on the type, the resources often have different properties associated with them (e.g. a “person” would have a name and an email address, but an “office” would have a location and possibly a code). A resource can be of multiple types and have the properties of both (e.g. a tropical bird may be a “bird” and a “tropical animal”).An example of a type in ESCO is the <http://data.europa.eu/esco/model#Occupation type, which represents occupations.

## Ontologies

The set of types, predicates and their meaning is expressed in ontologies, like the SKOS ontology[10]. This popular RDF ontology is used to represent knowledge organisation systems like thesauri, taxonomies and classification schemes. It is used to describe and organise concepts in general.

## Concept scheme

SKOS defines a set of predicates to describe a concept as well. For instance, skos:prefLabel provides the preferred way humans can name a concept (SKOS properties will be explained in more detail later). SKOS concepts are organised in a

[10] SKOS: http://www.w3.org/2004/02/skos/

concept scheme. Concept schemes are collections of concepts that describe a particular knowledge area.

This document often contains example triples to illustrate a section. In such a case, only the triples that are relevant to that section are shown. The subject, predicate and object may have additional information connected to them at any time, even if that information is left out to keep this document readable. Furthermore, examples of resources used in this document are not guaranteed to be actual ESCO concepts.

**Make sure you understand the following concepts from this section**

URI, resource, triple (subject, predicate and object), graph, links, types, ontologies, concept schemes and prefixes.

## 2.2 ESCO pillar concepts

The resources contained in the pillars of ESCO are called ESCO pillar concepts. The ESCO ontology is an extension of the SKOS ontology, meaning that these concepts are specialisations of SKOS concepts, with some extra information added to them. This section will show different types of ESCO concepts, list their extra properties and show the different kinds of relations the concepts have.

### *Types of ESCO pillar concepts*

There are three pillars of ESCO concepts, the occupation pillar, the skill/competence pillar and the qualification pillar. Each pillar defines its own type of concept – an esco:Occupation, an esco:Skill and an esco:Qualification – all of which are more specific versions of the esco:Concept type. Each concept can define some specific properties. For a list of all properties and types used in ESCO, please see the technical specification.

### *ESCO pillar concepts labels*

ESCO pillar concepts have human readable names, which are called ‘labels’. There are two types of labels for ESCO concepts: the “preferred” term (skosxl:prefLabel) and the “non-preferred” terms (skosxl:altLabel). There are also the hidden labels (skosxl:hiddenLabel), which are never shown to the user, but can be used for searching purposes.

The preferred term provides the way a concept is preferably represented in ESCO interfaces. It should always be of the neutral gender and there is only one preferred term per language. Non-preferred terms provide other ways of labelling a concept and can be of the neutral, male or female grammatical gender. These terms usually provide synonyms for the described concept, but can also be spelling variants, declensions, abbreviations, etc. A final type of label used for ESCO concepts is the hidden label. Hidden labels are used for defining search results, but will not be shown to the user when describing the ESCO concept. They can be used for common spelling mistakes or politically incorrect terms relating to a concept.

It is important to note that labels of ESCO pillar concepts are resources themselves. This is because, apart from their actual text, the labels also need gender information and the language of the label. As a result, labels also have a URI and when an ESCO pillar concept defines its label, it does so by placing a link to a label resource. This is shown in the image below:

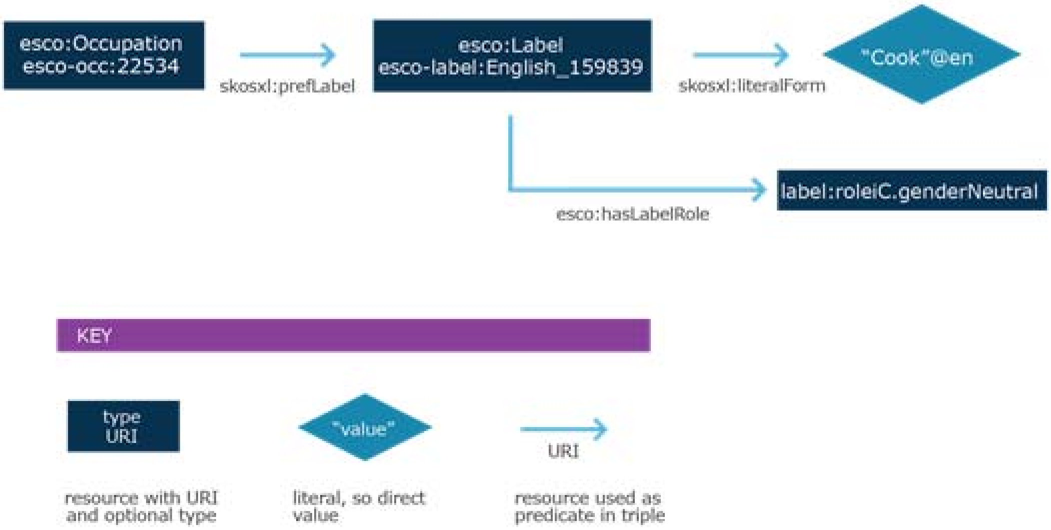


**Figure 5: ESCO concept labels**

## Properties per concept type

The table below lists all the properties per type of ESCO concepts. It is important to note that all of these properties are therefore also shared with the more specific types of ESCO concepts, e.g. occupations, skills and qualifications. Relations to ESCO concepts, hierarchical relations, labels and mapping relations are described further in the document.

| ***esco:Concept*** | |
|---|---|
| ***dct:contributor*** | An agent (person or organisation) that is responsible for contributing to the concept. |
| ***dct:created*** | Date at which the concept was created. |
| ***dct:creator*** | An agent (person or organisation) that created the concept. |
| ***dct:description*** | An informal description of the concept. |
| ***dct:identifier*** | An unambiguous reference to this concept within some context. A concept can have multiple identifiers, depending on the context. The format is "{qualifier} : {identifier}" – for instance, ESCO-ITM: 1213. |
| ***dct:issued*** | Date of official issuance (e.g. publication) of the concept. |
| ***dct:modified*** | Date at which the concept was last modified. |
| ***dct:owner*** | An agent (person or organisation) that possesses the |

| ***esco:Concept*** | |
|---|---|
| | intellectual ownership of the concept. |
| ***esco:editorialStatus*** | The stage at which the concept is in the editing flow. |
| ***esco:hasNaceCode*** | The NACE code of an industry sector that the concept belongs to. |
| ***esco:releasedWithVersion*** | The version number of the ESCO taxonomy that the concept was *first* released with. |
| ***iso-thes:status*** | The stage at which the concept is in the publishing flow. |
| ***rdf:type*** | The type of the concept (a concept can have multiple types). |
| ***skos:changeNote*** | Smaller changes to the concept, like spelling corrections. Mostly filled in by a human being. |
| ***skos:definition*** | The formal definition of the concept, as you would find it in e.g. a dictionary. |
| ***skos:historyNote*** | Significant changes to the meaning of a concept. Often filled in automatically. |
| ***skos:inScheme*** | The ESCO pillar the concept belongs to. |
| ***skos:notation*** | The unique identifier of the ESCO concept within its pillar. |
| ***skos:scopeNote*** | Practical instructions on when to apply the concept and where its limits lie. |

The ESCO occupations have the following extra predicates:

| ***esco:Concept*** | |
|---|---|
| ***esco:regulatedProfessionNote*** | Describes the regulation applying to the occupation. |
| ***esco:WorkContext*** | The possible work contexts of the occupation. |

Skills and competencies in ESCO have the following extra properties:

| ***esco:Concept*** | |
|---|---|
| ***esco:skillReuseLevel*** | How broadly the skill is reused in the ESCO taxonomy:<br>• *skill-reuse:iC.CrossSector*: the skill is used in a very |

| ***esco:Concept*** | |
|---|---|
| | broad range of sectors;<br>• *skill-reuse:iC.Transversal*: the skill is used in a few different sectors, e.g. the agriculture and veterinary sectors only;<br>• *skill-reuse:iC.SectorSpecific*: the skill is used in different occupations of the same sector;<br>• *skill-reuse:iC.OccupationSpecific*: the skill is used in one occupation only;<br>• *skill-reuse:iC.JobSpecific*: the skill is used only in specific jobs that are part of one occupation. |
| ***Esco:skillType*** | The sub-type of the skill, either skill/competence (http://data.europa.eu/esco/SkillCompetenceType#iC.Skill) or knowledge (http://data.europa.eu/esco/SkillCompetenceType#iC.Knowledge). |

**Make sure you understand the following concepts from this section**

ESCO pillar concepts (esco:Occupation, esco:Skill and esco:Qualification), concept labels (preferred, non-preferred and hidden) and properties per concept type.

## 2.3 ESCO supporting concept

Apart from the dedicated ESCO concepts, ESCO also contains supporting concepts. These concepts do not fall into the ESCO pillars and as the name suggests, they fulfil a more supporting role in the ESCO taxonomy. Some concepts are created in the context of ESCO; others are reused from existing taxonomies.

Supporting concepts are grouped into supporting taxonomies. Every taxonomy handles a certain domain and are a few examples of supporting taxonomies in ESCO are shown below:

- esco-cs:CEFR/ConceptScheme: Common European Framework of Reference for Languages: Learning, Teaching, Assessment;
- esco-cs:country: list of countries as per http://www.geonames.org/countries;
- esco-cs:EQF2012/ConceptScheme: European Qualifications Framework for lifelong learning - (2008/C 111/01);
- esco-cs:FoET2013/ConceptScheme: Fields of Education and Training from the International Standard Classification of Education, version 2013;
- esco-cs:ISCO1988/cs: International Standard Classification of Occupations, 1988 version;
- esco-cs:ISCO2008/cs: International Standard Classification of Occupations, 2008 version;
- esco-cs:ISO639/cs: List of languages as per the ISO 639-1/2 standard;
- esco-cs:NACErev2/cs: Statistical Classification of Economic Activities in the European Community, Rev. 2 (2008);
- esco-cs:NUTS2008/cs: Nomenclature of territorial units for statistics, 2008 version.

The supporting taxonomies that are used in ESCO were not available in Linked Data format. Therefore, they were transformed to a skos:ConceptScheme and new URIs were created for the concept scheme and its concepts.

The image below shows how supporting taxonomies are used in ESCO, with the kinds of relationships that connect them to ESCO. To keep the image simple, not all supporting taxonomies have been shown. The relationships are further detailed in the next section.

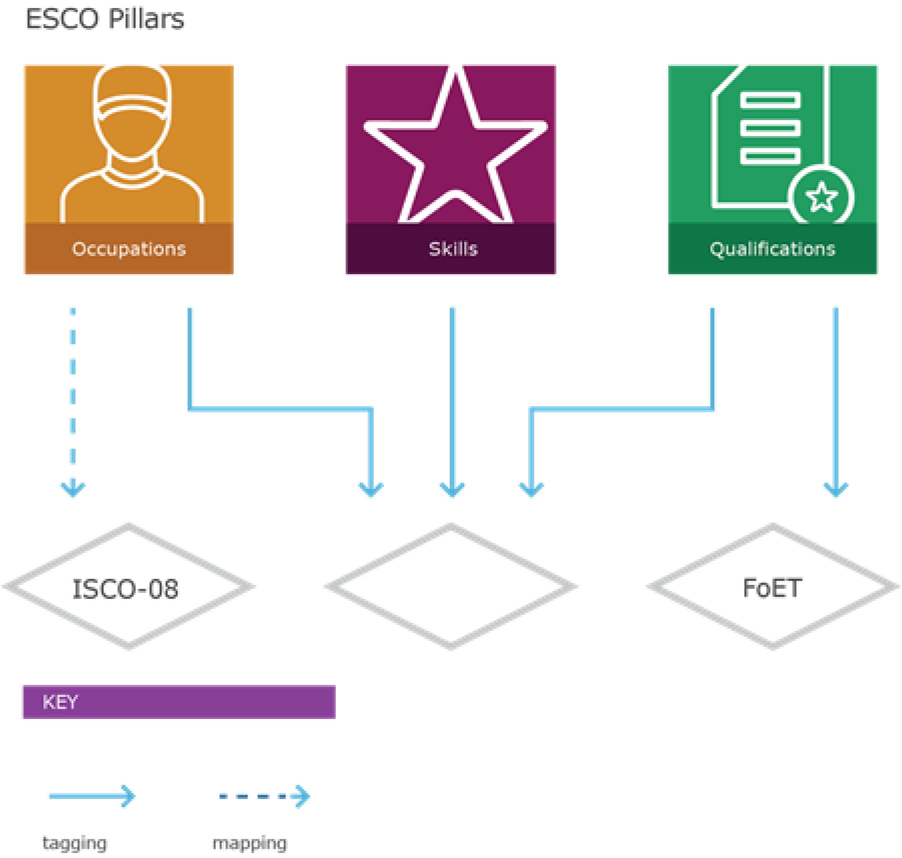


**Figure 6: Use of supporting taxonomies**

## 2.4 ESCO concept relations

As the name suggests, a Linked Data concept is only truly complete if it is linked with other concepts. Such a connection can be made easily in RDF; it is enough to introduce a new triple that uses the origin of the link as a subject, the target as an object and the kind of link as the predicate. For instance, the following triple defines that the cook occupation defined earlier is part of the occupations pillar in ESCO:

*esco-occ:22534 skos:inScheme esco-cs:ESCO_Occupations*

There are four different types of relations in ESCO:

- relations between ESCO pillars;
- tagging the relations of ESCO concepts with concepts outside of ESCO;
- mapping relations that connect ESCO concepts to concepts from other taxonomies with comparable meanings;
- hierarchical relations.

Each of these relations are presented in this chapter.

### Relations between ESCO pillars

The different types of ESCO concepts are connected to one another through specific ESCO relationships. These relationships are always sub-properties (so more specific versions) of the skos:related property and they are described in this section.

Relations between ESCO concepts are actually ‘rich’ relations. This means that the relations themselves are represented by their own resource, so information can be expressed about the relation itself (e.g. the relation can receive a description and a qualifier signaling the kind of relation). The type of these relations is always esco:Relationship.

### Relation to skills

ESCO occupations are related to ESCO skills through two kinds of relations, esco:RelationshipType#iC.essentialSkill and esco:RelationshipType#iC.optionalSkill, which provide the semantics for this instance of esco:Relationship. 'Essential skills' are those skills that that are usually required when working in an occupation, independent of the work context or the employer. 'Optional skills' refer to skills that may be required or occur when working in an occupation depending on the employer, working context or country. The example below illustrates the use of esco:Relationships as rich relations:

*example:relation/1 rdf:type esco:Relationship* (1)

*example:relation/1 esco:isRelationshipFor example:occupation/1* (2)

*example:relation/1 esco:refersConcept example:skill/1* (3) *example:relation/1 esco:hasRelationshipType esco:RelationshipType#iC.essentialSkill* *(4)*

*example:relation/1 dcterms:description "description for this relation"@en* *(5)*

*example:occupation/1 esco:relatedEssentialSkill example:skill/1* *(6)*

*example:skill/1 esco:isEssentialSkillFor example:occupation/1* *(7)*

The relation is made explicit. It receives the URI example:relation/1 and additional information is coupled to this URI: the fact that it is a relationship (1), the origin of the relationship (2), the target of the relationship (3), the kind of relationship (4), and a description for the relationship (5). For convenience, two more triples are inferred from the rich relationship that provide a direct (but not rich) relationship between the occupation and the skill (6) and vice-versa (7).

The case for esco:RelationshipType#iC.optionalSkill is similar to this one; the inferred relation is esco:relatedOptionalSkill and its inverse is the esco:isOptionalSkilllFor predicate.

This same type of relation can be used to have an esco:Skill refer to optional or essential skills. In that case, the origin of the relation is simply an esco:Skill as well, while the other properties and values will be the same as in the example above. There is one additional constraint: a skill/competence typed esco:Skill cannot be an essential/optional skill for a knowledge typed esco:Skill. The same procedure applies to esco:Qualifications.

The drawing below provides a graphical representation of the triples above, to clarify which resources are involved in rich ESCO relationships.

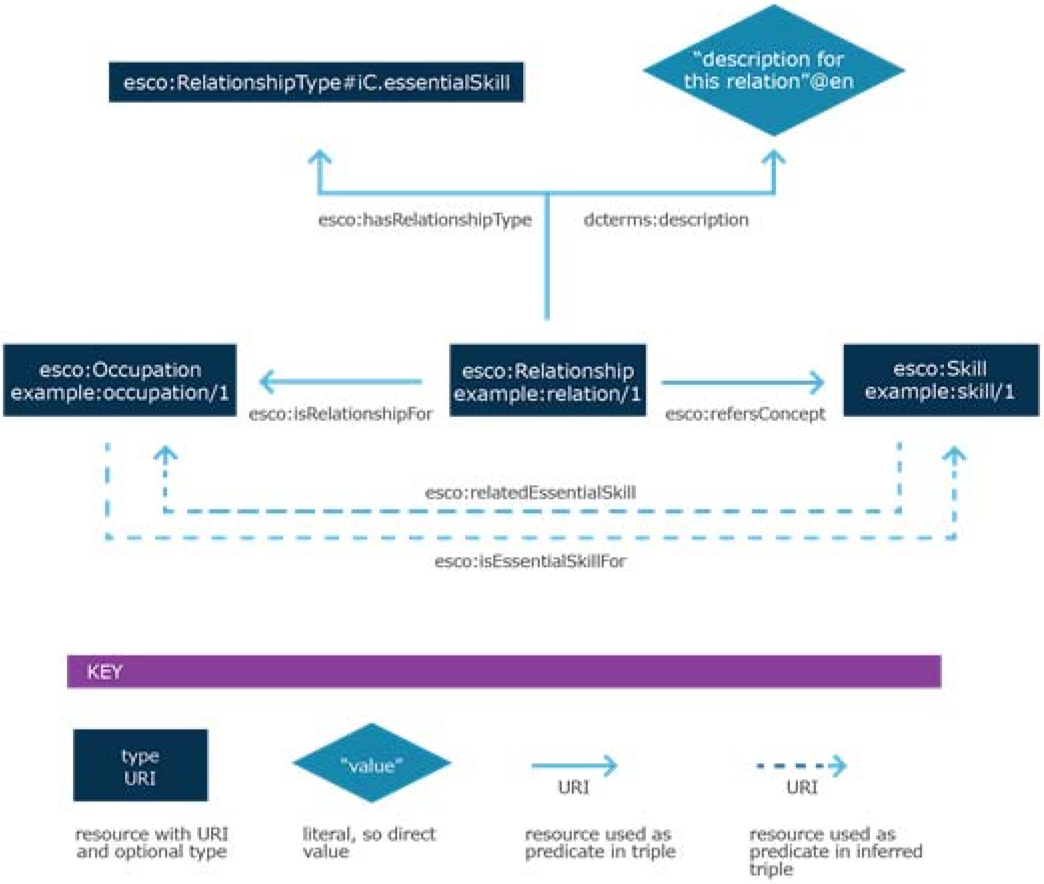

esco:RelationshipType#iC.essentialSkill
"description for this relation"@en
esco:hasRelationshipType
dcterms:description
esco:Occupation example:occupation/1
esco:isRelationshipFor
esco:Relationship example:relation/1
esco:refersConcept
esco:Skill example:skill/1
esco:relatedEssentialSkill
esco:isEssentialSkillFor
KEY
type URI
resource with URI and optional type
"value"
literal, so direct value
URI
resource used as predicate in triple
URI
resource used as predicate in inferred triple

**Figure 7: ESCO's rich relations**

## *Tagging relations*

To enrich their content, ESCO concepts also contain links to resources that are not part of the ESCO taxonomy. Such relations pointing to resources outside of the ESCO taxonomy are called either “tagging” or “mapping” relations. The difference is that where a mapping relation always talks about a *resource with a comparable meaning*, tagging relations are between *any kind of resource*. An example of a tagging relation is the ISCO-08 tagging of ESCO.

Tagging relations are not only useful for enriching ESCO with additional information, they also come in handy for indexing purposes. For instance, triples like those shown in the example above allow searches like: ‘give me all occupations that are relevant for the following ISCO-08 code’. Tagging relations in ESCO are more specific versions of the Dublin Core dcterms:subject property.

## *Mapping relations*

Mapping relations are used for aligning different taxonomies that describe the same type of content. This way, a translation is made between the two taxonomies. The predicates that are used to link the two resources are sub-properties of the skos:mappingRelations, which have the following meanings[11]:

- **skos:relatedMatch:** used to state an associative mapping link between two resources. Note that SKOS does not specify the way the two resources are related.
- **skos:closeMatch:** indicates that two resources are sufficiently similar that they can be used interchangeably in applications that consider the two resource schemes they belong to. However, skos:closeMatch is not defined as transitive, which prevents such similarity assessments to propagate beyond these two schemes. If a resource (ex1:A) is a close match for another resource (ex2:B), which is itself a close match for ex3:C, it does not follow from the SKOS data model that ex1:A is a close match for ex3:C.
- **skos:exactMatch:** also indicates semantic similarity—it is a sub-property of skos:closeMatch. However, it denotes an even higher degree of closeness: the two resources have equivalent meaning, and the link can be exploited across a wider range of applications and schemes. skos:exactMatch is indeed transitive: if a resource (ex1:A) is an exact match for another resource (ex2:B), which is itself an exact match for ex3:C, it does follow from the SKOS data model that ex1:A is an exact match for ex3:C.
- **skos:broadMatch:** this is a sub-property of skos:broader. It is used to define a hierarchical relation between two resources where the object is more specific than the subject. The inverse of this relation is the skos:narrowMatch relation.
- **skos:narrowMatch:** this is a sub-property of skos:narrower. It is used to define a hierarchical relation between two resources where the object is more generic than the subject. The inverse of this relation is the skos:broadMatch relation.

ESCO allows direct use of the SKOS mapping relations, however it is possible that a certain relation has a high enough business relevance that it merits a specific property. This is currently only the case for the esco:memberOfISCOGroup property, which is used to refer from an esco:Occupation (MemberConcepts only, see later) to a ISCO-08 or ISCO-88 code. This property is a more specific variant of the skos:broadMatch property and it is very important because it gives a different hierarchical view on ESCO, according to ISCO occupation groups. In ESCO v0, the ISCO occupation groups even provided the only hierarchy for occupations as ESCO had no esco:OccupationGroups yet.

Through the use of mapping properties, ESCO works as a hub vocabulary that allows other thesauri to translate their contents first to ESCO, and then from ESCO to

[11] See the SKOS primer.

another taxonomy that is also mapped to ESCO. For instance, if we know that an occupation in ROME, used in France, is connected by skos:exactMatch to an ESCO occupation, and this occupation is also connected to an occupation from the Dutch SBC classification, then the ROME occupation can be translated to the SBC occupation through ESCO.

## Hierarchical relations

The ESCO pillars are organised according to a hierarchy. The hierarchy orders the concepts in a pillar from generic (top) to specific (bottom).

### GroupConcept vs. MemberConcept

As described before, there are currently three pillars in ESCO, each with their own type of concept. The concepts in these pillars are organised in a hierarchy and to clarify their position in the hierarchy, they receive an additional type. The hierarchy of a pillar is illustrated in Figure 8. The pyramid represents an ESCO pillar; the top of an ESCO pillar is made up of esco:GroupConcepts, while the bottom consists of esco:MemberConcepts. Take, for example, the ESCO occupation pillar. The MemberConcepts are the core components of the ESCO pillars; they are what the ESCO pillars are meant to describe. The GroupConcepts simply organise the MemberConcepts.

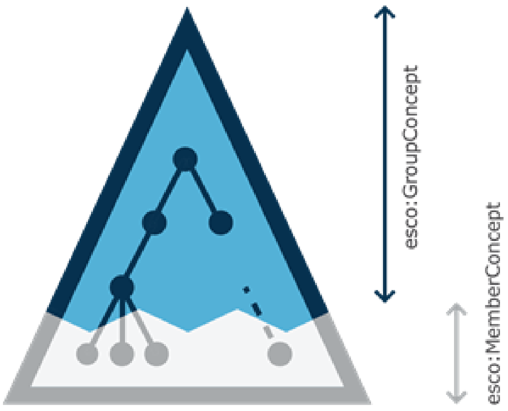


Figure 8: ESCO pillar hierarchy

The GroupConcepts in this pillar create logical groupings or clusters of the occupations, which become more fine-grained with every level toward the bottom of the tree. A more specific version of a GroupConcept is a subgroup of its more general grouping. This means that all of its members are also part of the more general group.

At some level in the tree, the GroupConcepts do not have any more fine-grained group concepts, but only MemberConcepts. This type of GroupConcept is called a esco:LeafGroupConcept. The level at which the LeafGroupConcepts are located is not fixed; it can be different for different branches within the same pillar.

### Hierarchical relation properties

To create the hierarchy of pillars, ESCO makes use of the skos:broader property. An extra distinction is made for the relation of the most course-grained MemberConcept and the LeafGroupconcept that it is linked to. Here esco:broaderInstance is used, but this is simply a more specific version of skos:broader. It is used to better illustrate the breaking line between GroupConcepts and MemberConcepts in the pillar. For instance, take the following triples:

*esco-oc:occupation esco:broaderInstance esco-oc:occLeafGroup* *(1)*

*esco-oc:occLeafGroup skos:broader esco-oc:occGroupTL* *(2)*

*esco-oc:occupation esco:memberOfGroup esco-oc:occLeafGroup* *(3)*

*esco-oc:occupation esco:memberOfGroup esco-oc:occGroupTL* *(4)*

These triples express that there is some member occupation, esco-oc:occupation, that has an immediate, more general, occupation group with URI esco-oc:occLeafGroup (triple 1). This group is part of a more general group, esco-oc:occGroupTL (triple 2). The occupation member concept also denotes all the occupation groups that it is part of, in every level of the hierarchy (triples 3 and 4).

The skos:broader property has an inverse, called skos:narrower. So this means that when there is a link as in triple 2 of the previous example, there is also a link in the inverse direction, using the skos:narrower predicate. This means that the top level occupation group has the leaf occupation group as a narrower occupation group.

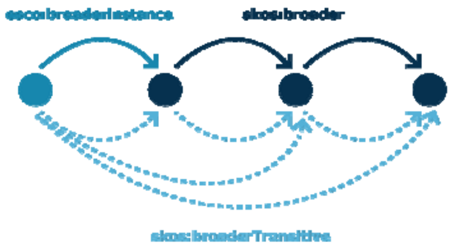


**Figure 9: skos:broaderTransitive relation**

All hierarchical relations in ESCO are created using the skos:broader and skos:narrower relations, or more specific versions of them. SKOS defines that every skos:broader predicate implies there is also a skos:broaderTransitive relation. As the name implies, this last property is the transitive version of skos:broader. This gives the picture shown in Figure 9. Because there is a broader relation from the first concept to the second, and a broader relation from the second to the third, there is also a broaderTransitive relation between them. And because of the transitivity of this relation, there is also a skos:broaderTransitive relation from the first to the third concept.

The same principle goes for the skos:narrower relation. Furthermore, this relation is used to infer an extra property that shows group membership inside an ESCO pillar. In Figure 10, the broader and narrower properties are shown for a path from a top level group concept to the most fine-grained member concept. The left side shows the inferred skos:narrowerTransitive relations for the top level occupation group. These are inferred in the same way the broaderTransitive relations were earlier. An extra rule is used on the right side of the image to infer the esco:hasGroupMember properties for this concept. The rule is that such a predicate is created from the top-level group to all *member concepts* for which there is a narrowerTransitive relation as well.

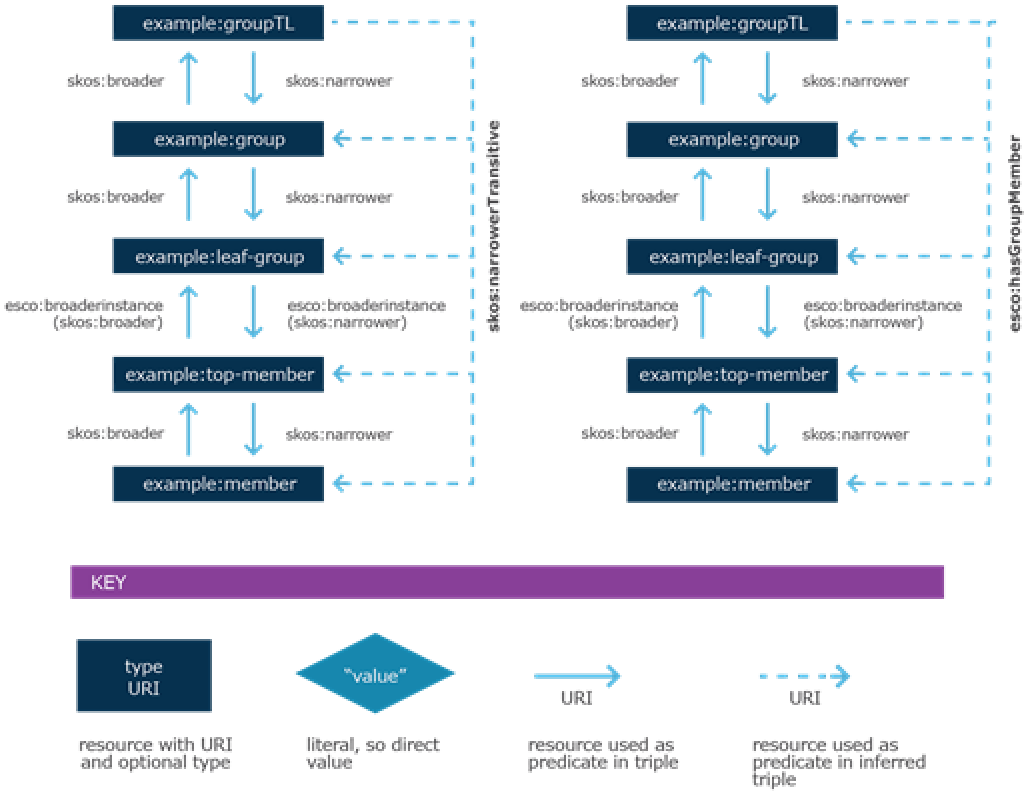


**Figure 10: Inferred group membership properties**

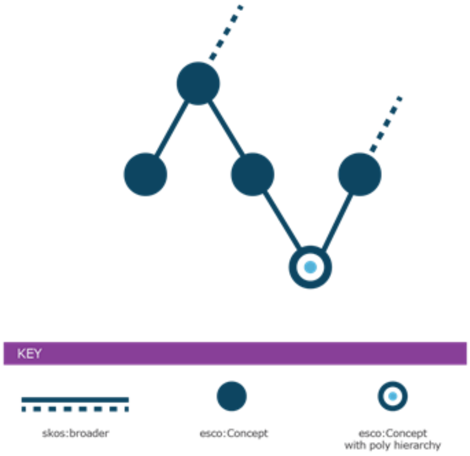
Through the esco:hasGroupMember relation, there is a direct relation from a group concept to all of the member concepts that are part of the group. The inverse property, pointing from the members to the group, is esco:memberOfGroup.

The hierarchical relations of ESCO are restricted to between concepts of the same pillar. For example, an occupation (be it a MemberConcept or a GroupConcept) can only have broader or narrower relations to other occupations.

### Poly-hierarchy

It is possible to have skos:broader relations to different concepts. This makes the structure of the occupation pillar a little bit more complex. as there are now multiple paths from the top of the hierarchy to a concept in the pillar. This is illustrated in Figure 11 below.

Note that the restriction still applies that skos:broader and esco:broaderInstance relations remain within an ESCO pillar.



The fact that there are multiple paths to an ESCO concept can seem hard to grasp at first, but it offers much greater modelling potential, especially when combined with inheritance (next section).

**Figure 11: Poly-hierarchy**

### Inheritance

What does it mean when a concept is more general than another concept? In ESCO, a more specific concept automatically receives a set of properties from the more general concept. This is called inheritance, and it is best illustrated with an example. Say we are looking at a BMW M3; this is a car, so it inherits the property that it has four wheels and an engine. As every car is also a vehicle, we know our particular M3 can be used to carry people or goods from one place to another. In short, wherever we talk about a car, we are free to use the term BMW M3 instead, as it has inherited all the properties of cars (and vehicles for that matter). This is called the Liskov Substitution principle[12].

ESCO pillar concepts work in much the same way. ESCO relationships (relations mentioned in the 'relations between concepts' section) from any broader transitive concept are inherited by their more specific concepts. There are some restrictions though:

- in case a relationship with the same skill is typed as essential as well as optional, e.g. because of a poly-hierarchical relationship, only the essential skill relationship is inherited;
- in case a relationship with the same qualification is typed as required as well as legally required, only the legally required qualification relationship is inherited.

Inherited relations and properties do not need to be repeated in the more specific concepts. They can be inferred by looking at the relations and properties of the more general concept.

Other relations that can be inherited are the NACE, FoET, SkillType and SkillReuseLevel tagging relations. However, there is an additional restriction here that

[12] http://en.wikipedia.org/wiki/Liskov_substitution_principle

when a broader concept is tagged with such a code, the narrower concepts of that concept can only be tagged with the same or more specific versions of the code used for tagging. This means that if occupation group 'alpha' is tagged with NACE code 'A', then the narrower occupations and occupation groups of 'alpha' can only be tagged with NACE code 'A' or more specific versions of this code, like '01' or '03.22'. If a narrower concept is already tagged with a more specific code, tagging is not inherited any more. Mapping relations are never inherited. Note that inheritance rules are applied before releasing and publishing the ESCO thesaurus, so users need not apply these ESCO business rules.

The strength of inheritance is that it avoids repeating the same information repeatedly. Following the previous example with the BMW M3, there is no need to say that the M3 has four wheels, an engine and that it drives on the road, because you already know it is a car and these properties are already specified for cars. This becomes obvious when we represent it graphically. Say that we want to model different kinds of musicians in ESCO. Without using inheritance, this would present the following diagram:

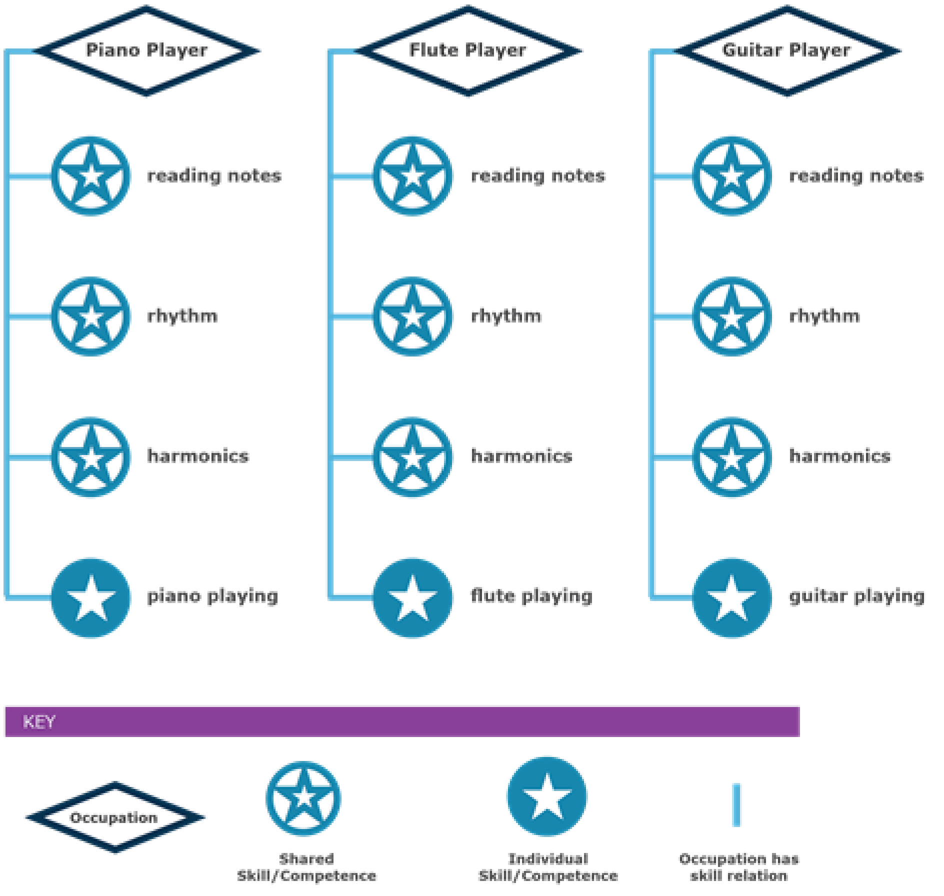


**Figure 12: Not using inheritance causes repetition**

Clearly, there is a large amount of repetition in this diagram. The three skills or competences 'reading notes', 'rhythm' and 'harmonics' are repeated for every occupation. By using inheritance, this problem is solved. The next figure shows the same occupations, but adds a more general occupation as well ("musician"). All the shared skills are set to the musician occupation, and so there is no need to repeat them anymore. Because of inheritance, we can derive that all the specific types of musicians, like a piano player, also have the skills of a musician.

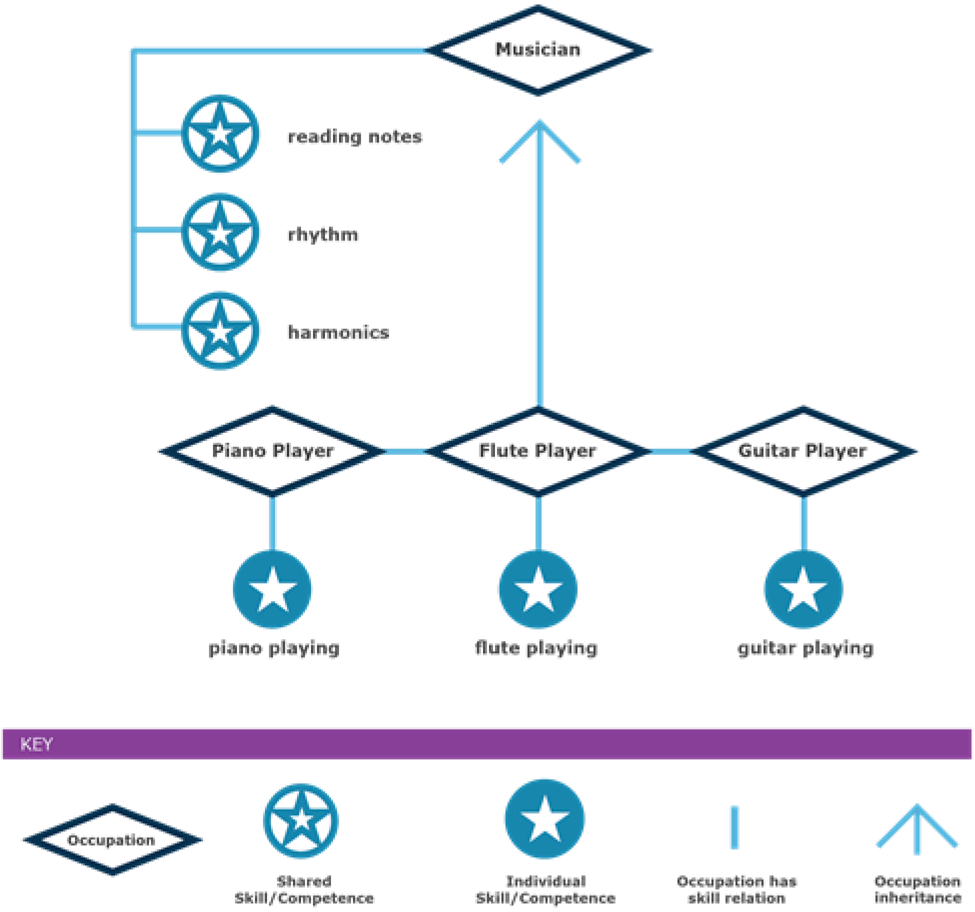


**Figure 13: Using inheritance to avoid repetition**

### Make sure you understand the following concepts from this section

ESCO concept relations, relations between ESCO pillars, relation to skills (essential and optional), tagging and mapping relations, hierarchical relations and inheritance.

# 3. Appendix

## Prefixes

The following prefixes are used in this document:

| Prefix | URL |
|---|---|
| **dct:** | http://purl.org/dc/terms/ |
| **esco:** | http://data.europa.eu/esco/model# |
| **esco-cs:** | http://data.europa.eu/esco/ConceptScheme/ |
| **esco-occ:** | http://data.europa.eu/esco/occupation/ |
| **esco-label:** | http://data.europa.eu/esco/label/ |
| **label-role:** | http://data.europa.eu/esco/LabelRole# |
| **esco-q:** | http://data.europa.eu/esco/qualification/ |
| **esco-sk:** | http://data.europa.eu/esco/skill/ |
| **iso-thes:** | http://purl.org/iso25964/skos-thes# |
| **owl:** | http://www.w3.org/2002/07/owl# |
| **rdf:** | http://www.w3.org/1999/02/22-rdf-syntax-ns# |
| **rdfs:** | http://www.w3.org/2000/01/rdf-schema# |
| **skill-reuse:** | http://data.europa.eu/esco/SkillReuseLevel# |
| **skos:** | http://www.w3.org/2004/02/skos/core# |
| **skos-xl:** | http://www.w3.org/2008/05/skos-xl# |

Publications Office