

TEXNİK SPESİFİKASIYA

ISO/TS 19139-1

Birinci nəşr 2019-03

Coğrafi informasiya — XML sxeminin tətbiqi — 1-ci hissə: Kodlaşdırma qaydaları

İstinad nömrəsi

ISO/TS 19139-1:2019(E)

© ISO 2019

LaVinci

MÜƏLLİF HÜQUQLARI QORUNUR

© ISO 2019,

Bütün hüquqlar qorunur. Əksi göstərilmədiyi təqdirdə, bu nəşrin heç bir hissəsi əvvəlcədən yazılı icazə almadan, heç bir formada və ya heç bir vasitə ilə, o cümlədən elektron və ya mexaniki üsullarla, kopyalanma və ya internetdə və ya intranetdə yerləşdirilməklə çoxaldıla və ya başqa şəkildə istifadə edilə bilməz. İcazə, ya aşağıda göstərilən ünvanda ISO-dan, ya da tələb olunan ölkədə ISO-nun üzv orqanından tələb edilə bilər.

ISO müəllif hüquqları ofisi

CP 401 • Ch. de Blandonnet 8

CH-1214 Vernier, Cenevrə

Phone: +41 22 749 01 11

Fax: +41 22 749 09 47

Email: copyright@iso.org

Website: www.iso.org

İsveçrədə nəşr olunub

© ISO 2019 – Bütün hüquqlar qorunur

Mündəricat

Önsöz	4
Giriş	6
1 Miqyas	Error! Bookmark not defined.
2 Normativ istinadlar	Error! Bookmark not defined.
3 Terminlər və təriflər	Error! Bookmark not defined.
4 Simvollar və qısaldılmış terminlər	Error! Bookmark not defined.
4.1 Qısaldılmış terminlər	Error! Bookmark not defined.
4.2 Namespace qısaldılmaları	Error! Bookmark not defined.
4.3 UML model stereotipləri	Error! Bookmark not defined.
4.3.1 UML model stereotiplərinə ümumi baxış	Error! Bookmark not defined.
4.3.2 Siniflərin stereotipləri	Error! Bookmark not defined.
4.3.3 Atributların stereotipləri	Error! Bookmark not defined.
4.3.4 Əlaqələrin stereotipləri	Error! Bookmark not defined.
4.3.5 Paketlərin stereotipləri	Error! Bookmark not defined.
5 Uyğunluq	Error! Bookmark not defined.
6 Kodlaşdırma üçün tələblər	Error! Bookmark not defined.
6.1 Tələblərə ümumi baxış	Error! Bookmark not defined.
6.2 Qaydalara əsaslanan	Error! Bookmark not defined.
6.3 Keyfiyyət	Error! Bookmark not defined.
6.4 Veb tətbiqləri	Error! Bookmark not defined.
6.5 Xarici XML tətbiqlərinin istifadəsi	Error! Bookmark not defined.
6.6 Polimorfizm	Error! Bookmark not defined.
7 Kodlaşdırma qaydaları	Error! Bookmark not defined.
7.1 Kodlaşdırma qaydalarına ümumi baxış	Error! Bookmark not defined.
7.2 Standart kodlaşdırma	Error! Bookmark not defined.
7.2.1 XML sinif növü (XCT)	11
7.2.1.3 XCT qayda nümunəsi	Error! Bookmark not defined.
7.2.2 XML Sinif Qlobal Elementi (XCGE)	Error! Bookmark not defined.
7.2.3 XML Sinif xassə Növü (XCPT)	Error! Bookmark not defined.
7.3 Xüsusi hal kodlaşdırmaları	15
7.3.1 Xüsusi hal kodlaşdırmalarına ümumi baxış	15
7.3.2 Abstrakt siniflər	15
7.3.3 İrsi və alt sinif kodlaşdırmaları	Error! Bookmark not defined.
7.3.4 Enumeration kodlaşdırmaları	19
7.3.5 CodeList kodlaşdırılması	22
7.3.6 Union kodlaşdırması	25
7.3.7 MetaClass-ların kodlaşdırılması	27
7.3.8 Xaricdən müəyyən edilmiş tətbiqlərin kodlaşdırılması	27
7.4 XML namespace paketinin kodlaşdırılması	Error! Bookmark not defined.
7.5 XML sxem paketi kodlaşdırılması	31
8 Əlavə kodlaşdırmalar	33

9 Modulluq və təkrar istifadə üçün kodlaşdırma	33
9.1 UML paketləri və XML namespace-ləri	33
9.2 UML modeli üçün XML tətbiqi	33
9.3 XML paketlərinin ayrılması üçün tətbiq yanaşması	34
9.3.1 Ümumi baxış	34
9.3.2 Tətbiq yanaşmasının qaydaları	34
9.3.3 Ayrılma nümunəsi	35
İstifadə olunmuş ədəbiyyat	Error! Bookmark not defined.

Önsöz

ISO (Beynəlxalq Standartlaşdırma Təşkilatı) milli standartlaşdırma orqanlarının (ISO üzv təşkilatları) qlobal federasiyasıdır. Beynəlxalq standartların hazırlanması işi, adətən, ISO-nun texniki komitələri vasitəsilə həyata keçirilir. Bu məsələdə maraqlı olan bir üzv qurum, yaradılmış Texniki Komitədə təmsil olunmaq hüququna malikdir. ISO ilə əlaqəsi olan beynəlxalq təşkilatlar, hökumət və qeyri-hökumət təşkilatları da bu işdə iştirak edirlər. ISO bütün elektrotezlik standartlaşdırması məsələlərində Beynəlxalq Elektrotezlik Komissiyası (IEC) ilə sıx əməkdaşlıq edir.

Bu sənədin hazırlanması üçün istifadə olunan prosedurlar və onun sonrakı texniki xidməti üçün nəzərdə tutulan prosedurlar, ISO/IEC Direktivlərinin 1-ci hissəsində təsvir edilmişdir. Xüsusilə, müxtəlif növ ISO sənədləri üçün tələb olunan müxtəlif təsdiq meyarları qeyd edilməlidir. Bu sənəd ISO/IEC Direktivlərinin 2-ci Hissəsinin redaksiya qaydalarına uyğun olaraq hazırlanmışdır.
www.iso.org/directives

Bu sənədin bəzi elementlərinin patent hüquqlarının predmeti ola biləcəyi ehtimalına diqqət yetirilir. ISO bu cür patent hüquqlarının hər hansı və ya hamısının müəyyən edilməsinə görə məsuliyyət daşımır. Sənədin işlənilib hazırlanması zamanı müəyyən edilmiş hər hansı patent hüquqlarının təfərrüatları Girişdə və/və ya alınan patent bəyannamələrinin ISO siyahısında olacaq.
www.iso.org/patents

Bu sənəddə istifadə olunan hər hansı bir ticarət adı istifadəçilərin rahatlığı üçün verilmiş məlumatdır və təsdiq kimi qəbul edilmir.

ISO-nun uyğunluq qiymətləndirilməsi ilə bağlı xüsusi termin və ifadələrinin mənası haqqında izahat və ISO-nun Ticarət üçün Texniki Maneələri (TBT, Technical Barriers to Trade) üzrə ÜTT prinsiplərinə riayət etməsi ilə bağlı məlumat üçün aşağıdakı URL-ə baxın: www.iso.org/iso/foreword.html.

Bu sənədə görə məsul komitə ISO/TC 211, Coğrafi informasiya/Geomatiklərdir.

ISO/TS 19139-1-in bu ilk nəşri texniki cəhətdən yenidən işlənmişdir, ISO/TS 19139:2007-ni ləğv edir və onu əvəzləyir.

ISO 19139 seriyasının bütün hissələrinin siyahısını ISO veb saytında tapa bilərsiniz.

Bu sənədlə bağlı hər hansı rəy və ya suallar istifadəçinin milli standartlaşdırma orqanına yönəldilməlidir. Bu orqanların tam siyahısını www.iso.org/members.html saytında tapa bilərsiniz.

LaVine
Quinn

Giriş

Rəqəmsal coğrafi məlumatları təsvir edən metadata-nın əhəmiyyəti ISO 19115-1 mətnində və digər Beynəlxalq Standartlarda, məsələn, ISO 19110, ISO 19119, ISO 19157-də ətraflı izah olunur. Bu sənədlər, metadata elementlərini təyin edərək və metadata terminologiyasının, təriflərin və genişləndirmə prosedurlarının ümumi dəstini quraraq rəqəmsal coğrafi məlumatların təsvir edilməsi üçün bir struktur təmin edir. Bu standartlar həmin metadata üçün kodlaşdırmaları təyin etmir. Standartların və oxşar sahə sxemlərinin tətbiqinin standartlaşdırılmasını asanlaşdırmaq üçün bu sənəd, ISO metadata standartlarının Extensible Markup Language (XML) formatında kodlaşdırılması üçün dəqiq bir qaydalar dəsti təmin edir. Nəticədə yaranan XML sxemləri, metadata-nı təsvir etmək, yoxlamaq və mübadilə etmək üçün ümumi spesifikasiya təmin edərək qarşılıqlı əlaqəni artırmağa xidmət edir. Bu qaydalar, tətbiq sxemlərinin XML/GML formatına kodlaşdırılması üçün ISO 19136:2007, Əlavə E-də olan qaydalarla paralel istifadə edilmək üçün nəzərdə tutulmuşdur. Fərq ondadır ki, həmin qaydalar xüsusiyyətləri təmsil edən məlumatlara aiddir; bu qaydalar isə həmin məlumatlar haqqında olan metadata-ya aiddir. ISO 19118, UML sxemləri əsasında kodlaşdırma qaydalarının yaradılması tələblərini və XML əsasında kodlaşdırma qaydalarını təsvir edir, həmçinin XML-i təqdim edir. Bu sənəd, ISO 19118-də müəyyən edilmiş kodlaşdırma qaydalarından istifadə edir və digər metadata standartlarının UML modelləri üçün XML sxemlərinin yaradılması ilə bağlı konkret detallarını təmin edir. Bu qaydalar ilk dəfə ISO 19115-1-in XML formatında kodlaşdırılması kimi ISO/TS 19115-3-ün yaradılmasında istifadə edilmişdir, yəni ISO/TS 19115-3 bu sənədə uyğundur. Onlar həmçinin ISO 19157-nin kodlaşdırılması üçün ISO/TS 19157-2-nin yaradılmasında istifadə edilmişdir. Bu sənədin standartlaşdırma hədəfi metadata-nın XML tətbiqləridir. Buraya həm Coğrafi Məlumat seriyası daxilində olan digər standartlar, həm də digər təşkilatlar tərəfindən hazırlanmış modellər daxildir. Coğrafi məlumat — XML sxemi tətbiqi — 1-ci hissə: Kodlaşdırma qaydaları

1 Miqyas

Bu sənəd, coğrafi resursları təsvir edən tipləri təyin edən konseptual sxemlər üçün XML əsasında kodlaşdırma qaydalarını müəyyən edir. Kodlaşdırma qaydaları, ISO/TC 211 tərəfindən hazırlanmış standartlarda geniş istifadə olunan UML modellərində istifadə olunan UML profilini dəstəkləyir. Kodlaşdırma qaydaları, çıxış məlumat strukturu sxemi üçün XML sxemindən istifadə edir. Bu sənəddə təsvir edilən kodlaşdırma qaydaları coğrafi xüsusiyyətlər üçün UML tətbiq sxemlərinin kodlaşdırılması üçün tətbiq edilmir (bu qaydalar üçün ISO 19136-ya baxın).

2 Normativ istinadlar

Aşağıdakı sənədlər mətnə elə bir şəkildə istinad edilir ki, onların bəzi və ya bütün məzmunu bu sənədin tələblərini təşkil edir. Tarixli istinadlar üçün yalnız göstərilən nəşr tətbiq olunur. Tarixsiz istinadlar üçün istinad edilən sənədin ən son nəşri (hər hansı düzəlişlər daxil olmaqla) tətbiq olunur.

ISO 19118, Coğrafi məlumat — Kodlaşdırma W3C XMLName, "Namespaces in XML".

W3C Recommendation W3C XMLSchema-1, "XML Schema Part 1: Structures".

W3C Recommendation W3C XMLSchema-2, "XML Schema Part 2: Datatypes".

W3C Recommendation W3C XML, "Extensible Markup Language (XML) 1.0",

W3C Recommendation W3C XLink, "XML Linking Language (XLink) Version 1.0".

W3C Recommendation

3 Terminlər və təriflər

Bu sənədin məqsədləri üçün aşağıdakı terminlər və təriflər tətbiq olunur. ISO və IEC terminoloji bazalarını standartlaşdırmada istifadə üçün aşağıdakı ünvanlarda saxlayır:

— ISO Online browsing platform: mövcuddur <https://www.iso.org/obp>

— IEC Electropedia: mövcuddur <https://www.electropedia.org/>

3.1

namespace

adlar toplusu, URI istinadı ilə müəyyən edilən və XML sənədlərində element adları və atribut adları kimi istifadə olunan adlar toplusu

[MƏNBƏ: W3C XML]

3.2

package

<UML> elementləri qruplara təşkil etmək üçün ümumi məqsədli mexanizm

EXAMPLE

Məlumatın tanınması, metadata obyekt dəsti məlumatı, məhdudiyət məlumatı.

[MƏNBƏ: ISO 19103:2015, 4.27 — dəyişdirilmiş: NÜMUNƏ əlavə edilmişdir.]

3.3

polymorphism

fərqli kontekstlərdə bir şeyə fərqli mənalar və ya istifadələr vermə xüsusiyyəti, xüsusən də

dəyişən, funksiya və ya obyekt kimi bir varlığa bir neçə formada olmağa icazə vermək
Qeyd 1 : Polimorfizmin bir neçə fərqli növü var.
[MƏNBƏ: <https://searchcio.techtarget.com/>]

3.4

realization

<UML> model elementlərinin iki dəsti arasında xüsusi abstraksiya əlaqəsi, biri spesifikasiyanı (təchizatçı), digəri isə onun implementasiyasını (müşəri) təmsil edir
Qeyd 1: Realization davranışın miras alınmasını, lakin strukturların miras alınmamasını göstərir.

[MƏNBƏ: ISO 19103:2015, 4.29]

4 Simvollar və qısaltılmış terminlər

4.1 Qısaltılmış terminlər

UML Unified Modeling Language – Vahid Modelləşdirmə Dili

URI Uniform Resource Identifier – Vahid formalı Resurs İdentifikatoru

XCT XML Class Type – XML Sinif növü

XCPT XML Class Property Type – XML Sinfinin Xassə Növü

XCGE XML Class Global Element – XML Sinfinin Qlobal Elementi

XML Extensible Markup Language – Genişlənəbilən Nişanlama Dili

XSD XML Schema Definition – XML Sxem Tərif

4.2 Namespace qısaltılmaları

Cədvəl 1 bu sənəddə istifadə edilən xarici namespace-ləri təqdim edir. Sol sütunda namespace-də elementləri təsvir etmək üçün istifadə olunan ümumi namespace prefiksi göstərilir. İkinci sütunda namespace prefiksinin ingilis dilində təsviri verilir. Üçüncü sütun isə faktiki namespace-in URI-sidir. Bu URI-lər sxemlərin yerləşdiyi yeri mütləq göstərmir.

Cədvəl 1 — Bu sənəddə istifadə edilən xarici namespace-lər

Namespace prefiksi	Namespace-in ingilis dilində təsviri	Faktiki namespace-in URI-si
gml	Geography Markup Language	http://www.opengis.net/gml/3.2
xlink	XML Linking Language (XLink)	http://www.w3.org/1999/xlink
xs	W3C XML base schemas	http://www.w3.org/2001/XMLSchema

4.3 UML model stereotipləri

4.3.1 UML model stereotiplərinə ümumi baxış

UML stereotipi mövcud UML konsepsiyalarına genişləndirmə mexanizmidir. Coğrafi resursları təsvir etmək üçün artıq müəyyən edilmiş stereotiplərə əlavə olaraq, bu sənəd XML sxeminə qaydalara əsaslanan kodlaşdırma üçün zəruri olan stereotipləri müəyyən edir. Bənd 7-də təsvir edilmiş UML diaqramlarının elementləri XML tətbiqini göstərən stereotipləri daşıya bilər. Aşağıdakı altbəndlərdə sadalanan həmin stereotiplər, XML elementlərini və ya XML tiplərini, UML atributlarını, UML əlaqələrini (realizasiyalar və ya asılılıqlar) və UML paketlərini təmsil edən siniflər tərəfindən daşınır.

4.3.2 Siniflərin stereotipləri

Bu sənəddə aşağıdakı sinif stereotipləri istifadə olunur:

a) <<xs: choice>>: Sinif XML seçim bloku kimi kodlaşdırılmış tətbiq növünü təmsil edir.

Sınıfın hər bir xassəsi seçim elementləri kimi tətbiq edilir.

- b) <<xs:complexType>>: Sınıf XML kompleks növü kimi kodlaşdırılmış tətbiq növünü təmsil edir.
- c) <<xs:simpleType>>: Sınıf XML sadə növü kimi kodlaşdırılmış tətbiq növünü təmsil edir.
- d) <<xs:simpleContent>>: Sınıf XML kompleks növü kimi sadə məzmunla kodlaşdırılmış tətbiq növünü təmsil edir.

4.3.3 Atributların stereotipləri

Bu sənəddə aşağıdakı atribut stereotipləri istifadə olunur:

- a) <<xs:attribute>>: xassə XML atributu kimi kodlaşdırılır.
- b) <<xs:attributeGroup>>: xassə XML attributeGroup kimi kodlaşdırılır.
- c) <<xs:element>>: xassə XML elementi kimi ad və növ ilə kodlaşdırılır (<xs:element name="propertyName" type="propertyType"/>).

4.3.4 Əlaqələrin stereotipləri

Bu sənəddə aşağıdakı əlaqə stereotipləri istifadə olunur:

- a) <<XCT>>: (realizasiya əlaqələri tərəfindən daşınır) Abstrakt konsepsiyanın tətbiqi üçün müəyyən edilmiş xarici tətbiq ilə əvəz edilir.
- b) <<XCGE>>: (realizasiya əlaqələri tərəfindən daşınır) Abstrakt konsepsiyanın tətbiqi üçün müəyyən edilmiş xarici tətbiq ilə əvəz edilir.
- c) <<XCPT>>: (realizasiya əlaqələri tərəfindən daşınır) Abstrakt konsepsiyanın tətbiqi üçün müəyyən edilmiş xarici tətbiq ilə əvəz edilir.
- d) <<implement>>: (asılılıq əlaqələri tərəfindən daşınır) Mənbə, hədəfdə müəyyən edilmiş abstrakt konsepsiyaları həyata keçirən XML sxemini təmsil edir.
- e) <<include>>: (asılılıq əlaqələri tərəfindən daşınır) Mənbə və hədəf XML sxemlərini təmsil edir. Mənbə hədəfi daxil edir (<xs:include ... />).
- f) <<import>>: (asılılıq əlaqələri tərəfindən daşınır) Mənbə və hədəf eyni namespace daxilində qruplaşdırılmış XML obyektlərini təmsil edir. Mənbə hədəfi idxal edir (<xs:import ... />).

4.3.5 Paketlərin stereotipləri

Bu sənəddə aşağıdakı paket stereotipləri istifadə olunur:

- a) <<xmlSchema>>: Paket XML sxemini təmsil edir.
- b) <<xmlNamespace>>: Paket eyni namespace daxilində qruplaşdırılmış XML obyektlərini təmsil edir.

5 Uyğunluq

Test üçün çərçivə, konsepsiyalar və metodologiya, həmçinin uyğunluq iddia etmək üçün yerinə yetirilməsi lazım olan meyarlar ISO 19105-də müəyyən edilmişdir. Coğrafi resursların XML sxemi tətbiqi, Əlavə A-da müəyyən edilmiş test modullarından keçərsə, bu sənədə uyğundur.

6 Kodlaşdırma üçün tələblər

6.1 Tələblərə ümumi baxış

Coğrafi resurslar, adətən, digər standartlar və spesifikasiyalarda müəyyən edilir, məsələn, ISO 19115-1-ə uyğun metadata və ISO 19110-a uyğun olaraq xüsusiyyət kataloqları. Coğrafi resurslar, bir və ya daha çox UML sinifləri ehtiva edən UML paketləri dəstindən ibarət konseptual sxemlər kimi təmsil olunur. Bu konseptual sxemlər əlaqəli coğrafi resursların kodlaşdırma-müstəqil görünüşünü təmin edir.

XML, mübadilə üçün məlumatları strukturlaşdırmaq üçün çoxsaylı alternativlər təklif edir.

ISO 19118-ə uyğun olaraq Bənd 7, coğrafi resursların konseptual sxemlərinə daha dəqiq şəkildə tətbiq edilə bilən XML sxemi kodlaşdırma qaydalarını müəyyən edir.

ISO 19118-in məhdudiyətləri və bu sənəddə müəyyən edilmiş kodlaşdırma qaydaları daxilində belə, xüsusi XML sxemlərinin yaradılması üçün hələ də seçimlər mövcuddur.

Bu sənəd coğrafi resursların konseptual sxemlərinə tətbiq ediləcək kodlaşdırma qaydalarından ibarət standartlaşdırılmış qaydalar dəstində coğrafi resursların XML sxemi tətbiqləri üçün zəruri olan təməli təmin edir.

XML namespace-lərinin təfərrüatları bu sənəddə daxil edilməmişdir. Namespace, XML sənədlərində element və ya atribut adları kimi istifadə oluna bilən adlar toplusudur.

Namespace identifikatoru müəyyən bir sxem ilə adları müəyyən etmək üçün istifadə olunur.

Namespace identifikatoru URI-dir. URI oxumaq, yazmaq və insan müzakirəsinə daxil etmək üçün bəzən çətindir, ona görə də bu sənəd namespace məzmununu müəyyən edərkən ümumi namespace prefikslərinə istinad edəcəkdir.

Kodlaşdırma təfərrüatlarını təqdim etməzdən əvvəl müəyyən kodlaşdırma qaydalarının niyə istifadə edildiyini başa düşmək vacibdir. Qaydaların başa düşülməsi onların imkanlarını, məhdudiyətlərini və ən yaxşı istifadə təcürbəsini aydınlaşdıracaq. Bu kodlaşdırma qaydası hazırlanarkən nəzərə alınan əsas məqsədlərdən bəziləri digər ISO 19100 seriyası spesifikasiyaları ilə qarşılıqlı əlaqə, proqnozlaşdırıla bilənlik, genişlənmə qabiliyyəti və istifadəsi idi. Bu məqsədlərin əlavə təfərrüatları 6.2-dən 6.6-a qədər izah edilmişdir.

6.2 Qaydalara əsaslanan

Bu sənəd ISO 19118-in tələb etdiyi kimi, ISO 19100 Beynəlxalq Standartlar seriyasındakı UML modellərindən yaradılan qaydalara əsaslanan kodlaşdırmanı müəyyən edir. Qaydalara əsaslanan üsul üç vacib məqsədə nail olur:

— birincisi, yaranan XML sxemləri birbaşa konseptual modellərə əsaslanır və bu, qarşılıqlı əlaqə imkanlarını artırır;

— ikincisi, yaranan sxem proqnozlaşdırılındır, çünki hər hansı bir sinif, atribut, əlaqə və s. həmin tipin digər UML elementləri kimi kodlaşdırılır;

— üçüncüsü, bu qaydalardan istifadə edən XML sxemləri avtomatik və ya yarı-avtomatik şəkildə yaradıla bilər.

6.3 Keyfiyyət

Bu sənədin kontekstində XML sxeminin keyfiyyəti konseptual modelin bütün elementlərinin ardıcıl şəkildə tətbiq olunmasını və istifadəçinin müvafiq coğrafi resursun konseptual sxemindən istifadə edərək bir XML nümunə sənədinin məzmununu birbaşa yaratmasını və/və ya başa düşməsinə nəzərdə tutur. Əlavə olaraq, tətbiqçi konseptual sxemi və kodlaşdırma qaydalarını bilməklə coğrafi resursun XML sxemi tətbiqini müəyyən edə bilər. Keyfiyyətin digər aspekti tamlıqdır. Bu sənəd coğrafi resursun bütün konseptual sxeminin istifadə və ya tətbiqə baxılmadan kodlaşdırılmasına imkan verir.

6.4 Veb tətbiqləri

6.1-də qeyd olunan məqsədlərdən biri istifadəyə yararlılıqdır. İstifadəyə yararlılıq, coğrafi resursların dizaynına aid olduğu kimi, onların mübadiləsini veb-mühitdə baş verəcəyini nəzərə alaraq diqqət mərkəzinə qoyur. Şəbəkə üzərindən heç vaxt ötürülməyəcək nümunə sənədlərinin yaradılmasına qarşı heç bir məhdudiyət olmasa da, dizaynda coğrafi resursların internet və veb-əsaslı ötürülməsini dəstəkləyən bir çox cəhətlər nəzərdə tutulmuşdur.

6.5 Xarici XML tətbiqlərinin istifadəsi

Əlaqələndirmə qabiliyyətini və istifadəyə yararlılığı artıran başqa bir dizayn prinsipi mövcud XML sxemlərinin təkrar istifadəsidir. Əgər coğrafi metadata ilə əlaqəli olan ISO 19100 seriyasının bir hissəsini kodlaşdıran bir XML sxemi standartı artıq mövcuddursa, həmin

XML sxemi standartını daxil etmək üstünlük təşkil edir. Xarici XML sxemindən birbaşa istifadə olunarsa, qarşılıqlı əlaqə qabiliyyəti artır. Bundan əlavə, həmin xarici XML ilə uyğun nümunə sənədlərini emal edə bilən proqram təminatı da artıq mövcud ola bilər. Əlavə olaraq, xarici sxem yaxşı dizayn olunmuşsa, kodlaşdırma qaydalarına əsaslanan XML sxemindən daha səmərəli ola bilər və bu da istifadəyə yararlılıq məqsədinə nail olmağa kömək edə bilər. Mövcud tətbiqdən istifadə etmək bəzi əhəmiyyətli üstünlüklərə malik olsa da, xarici XML sxemlərinin burada müəyyən edilmiş əsas dizayn prinsiplərinə zidd olmaması tövsiyə olunur. Ətraflı məlumat və nümunələr üçün 7.3.8-ə baxın.

6.6 Polimorfizm

Polimorfizm termini 3.3-də rəsmi olaraq müəyyən edilmişdir. Ümumi olaraq, polimorfizm fərqli formalarda mövcud olmaq qabiliyyəti deməkdir, yəni xassələri onların məlumat tipi və ya hər hansı törəmə sinfi ilə tətbiq etmək deməkdir. Polimorfizm, tətbiqçilərə öz namespace-ləri daxilində xassələri n daha ümumi formatını genişləndirməyə imkan verir, lakin hələ də onların təşkilatından kənar istifadəçilər üçün istifadəyə yararlı və başa düşülən nümunə sənədləri təqdim edir. Polimorfizm əsasən 7.3-də təsvir olunan xassə növü kodlaşdırmalarından yaranır.

7 Kodlaşdırma qaydaları

7.1 Kodlaşdırma qaydalarına ümumi baxış

UML-in XML sxeminə çevrilməsi üçün ümumi qaydalar aşağıda təsvir edilmişdir və ISO 19118-də müəyyən edilmiş qaydalara uyğundur. Bəzi hallarda ISO 19118 UML-in XML sxeminə çevrilməsi üçün bir neçə metod təklif edir və burada müəyyən edilmiş qaydalar bu sənədlə uyğunlaşmaq üçün hansı metodun istifadə olunacağını aydınlaşdırır. Siniflərin məlumat mübadiləsi (və bu halda metadata mübadiləsi) üçün bütün kodlaşdırmaların əsasını təşkil etdiyini və ISO 19118-in bunu ətraflı izah etdiyini nəzərə alaraq, siniflər haqqında məlumat bu sənəddə məqsədli şəkildə verilməmişdir. Bu sənəd ISO 19118-də olan kodlaşdırma qaydalarına əsaslanır və həmin sənədlə tanışlıq, 7-ci bənddə təsvir olunan mövzuların daha yaxşı başa düşülməsinə kömək edəcəkdir.

QEYD: sxemlər.

7.2 Standart kodlaşdırma

7.2.1 XML sinif növü (XCT)

7.2.1.1 XML sinif növünə (XCT) ümumi baxış

UML-də sinif (ISO 19118) əsas modelləşdirmə konsepsiyasıdır, buna görə də əsas kodlaşdırma qaydaları UML sinifinin kodlaşdırılmasına diqqət yetirir və oradan irəli gəlir. Sinif bir və ya bir neçə xassədən ibarətdir. ISO 19118-dən xatırlatmaq lazımdır ki, xassə atributu, əlaqə, aqreqasiya və ya kompozisiyanı təmsil edə bilər (ISO 19118). Məsələn, Şəkil 1-də **Class1** sinifinin üç xassəsi var: **attr1**, **attr2** və **role1**. XML sxeminə kodlaşdırma məqsədi ilə UML atributları, əlaqələr, aqreqasiyalar və ya kompozisiyalar arasında heç bir fərq yoxdur.

ISO 19118 həmçinin XML sxemində identifikatorlar (ids) və domain unikalıdır identifikatorlar (DUIDs) kimi identifikatorların istifadə edilməsini təsvir edir. Gco namespace-də **baseTypes2014.xsd**-də xüsusi bir XML sxemi növü var, **gco:AbstractObject_Type**, bu isə lazımı identifikatorları təmin edir. Burada qeyd olunur, çünki bu, XML sinif növü kodlaşdırmasının bir hissəsidir.

7.2.1.2 XCT qaydası

Bir iş nümunəsi olaraq, Şəkil 1-dəki UML aşağıdakı tələblərdən istifadə edərək XML-ə kodlaşdırılır.

Şəkil 1 — Nümunə

Requirement /req/default/XCT

UML sinifi XML sxeminə **XML complex type (xs:complexType)** kimi, XML sinif növü (XCT) adlanan formada kodlaşdırılmalıdır.

Requirement /req/default/XCT-name

Hər bir XCT sinif adının sonuna **_Type** şəkilçisi əlavə edilən bir ad atributuna malik olmalıdır:

NÜMUNƏ 1: Şəkil 1-də göstərilən **Class1** sinifinin yaradılmasının 1-ci addımı:

```
<xs:complexType name="Class1_Type">
  (...)
</xs:complexType>
```

Requirement /req/default/XCT-complex-content

Standart kodlaşdırma qaydalarına uyğun olan bütün UML siniflərinin kompleks məzmunu olmalıdır və bu qabiliyyəti təmin etmək üçün **xs:complexType** elementi **xs:complexContent** elementini ehtiva edir.

NÜMUNƏ 2: Şəkil 1-də göstərilən **Class1** sinifinin yaradılmasının 2-ci addımı:

```
<xs:complexType name="Class1_Type">
  <xs:complexContent>
    (...)
  </xs:complexContent>
</xs:complexType>
```

Requirement /req/default/XCT-extend-abstract

Standart kodlaşdırma qaydalarına uyğun olan bütün UML sinifləri **gco: AbstractObject _Type**-i genişləndirməlidir və bu, **xs: extension** elementi əlavə edilərək və **base** atributu **gco:AbstractObject _Type**-ə bərabər olmaqla yerinə yetirilir.

NÜMUNƏ 3: Şəkil 1-də göstərilən **Class1** sinifinin yaradılmasının 3-cü addımı:

```
<xs:complexType name="Class1_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      (...)
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Requirement /req/default/XCT-sequence

Standart kodlaşdırma qaydalarına uyğun olan bütün UML sinifləri sinifin bütün xassələri ni ehtiva edən bir ardıcılığa malik olmalıdır. Bu, hər sinifin xassəsi üçün **xs: element** elementlərini ehtiva edən bir **xs: sequence** elementinin əlavə edilməsi ilə yerinə yetirilir.

Requirement /req/default/XCT-properties

xs: element elementinin atributları bunlardır:

- name** atributu, xassənin adına bərabərdir;
- type** atributu, xassənin tipi olaraq təyin edilmiş UML sinfinə uyğun gələn XCPT-nin adına bərabərdir. 7.2.3 altbəndi standart olaraq bunun sinif adı və **_PropertyType** şəkilçisi olduğunu izah edir və 7.3.8 XCPT adlandırma konvensiyasının yeganə istisnalarını müəyyən edir. Bu **type** atributu üçün dəyər kimi istifadə olunan XCPT adı müvafiq namespace ilə düzgün şəkildə prefikslə göstəriləcək.
- minOccurs** və **maxOccurs** atributları, ISO 19118:2011, Cədvəl C.5-də təsvir edilən dəyərlərdir. Əlavə olaraq, əgər sinifin hər hansı bir xassəsi 'set' və ya 'sequence' strukturundan

istifadə edən atributdursa, o zaman **minOccurs** atributu opsional atributlar üçün "0", məcburi atributlar üçün isə "1" olmalıdır və **maxOccurs** 'sərhədsiz'dir.

NÜMUNƏ 4: Şəkil 1-də göstərilən **Class1** sinifinin yaradılmasının 4-cü addımı:

```
<xs:complexType name="Class1_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="attr1" type="ns1:typeAttr1_PropertyType"/>
        <xs:element name="attr2" type="ns1:typeAttr2_PropertyType"
minOccurs="0" />
        <xs:element name="role1" type="ns1:Class2_PropertyType"
minOccurs="1"
          maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

QEYD: Bu sənəddəki nümunələrdə “**ns1**” adlı namespace prefiksi var, hansı ki UML-də hər zaman göstərilir. Bu uydurma namespace, namespace prefiksinin mövcud olması lazım olan hallarda göstərmək üçün istifadə olunur, lakin bu sənəddə müəyyən edilmiş namespace və ya müvafiq prefikslərdən heç birinə uyğun deyil.

Requirement /req/default/XCT-properties-sequence

xs: sequence daxilindəki **xs: element** elementlərinin sırası konseptual sxemi sənədləşdirən məlumat lüğətində xassənin uyğun gələn qeydinin sırası ilə eyni olmalıdır. Bu tələb standart kodlaşdırma və XCT-lər üçün bütün xüsusi hallar kodlaşdırılmalarına tətbiq olunur.

7.2.1.3 XCT qayda nümunəsi

Yuxarıdakı XCT qaydasını bu UML nümunəsinə (ISO 19157-dən) tətbiq etmək aşağıdakı XML-ə nəticə verir.

Şəkil 2 — StandAloneQualityReport məlumatı UML (ISO 19157:2013, 10.1)

Şəkil 2-yə uyğun XCT:

```
<xs:complexType name="DQ_StandaloneQualityReportInformation_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="reportReference"
type="mcc:Abstract_Citation_PropertyType"/>
        <xs:element name="abstract"
type="gco:CharacterString_PropertyType"/>
        <xs:element name="elementReport"
type="mdq:AbstractDQ_Element_PropertyType"
          maxOccurs="unbounded" minOccurs="0" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

7.2.2 XML Sinif Qlobal Elementi (XCGE)

7.2.2.1 XML Sinif Qlobal Elementinə (XCGE) ümumi baxış

Qlobal element sxemin digər bölmələrindən istinad edilə bilməsi üçün təyin olunur.

7.2.2.2 XCGE qaydası

Requirement /req/default/XCGE

Bir UML sinifi həmçinin XML Sinif Növü (XCT) adlandırılan sxem tipinə uyğun olan və adı UML sinifi ilə eyni olan bir qlobal element kimi kodlaşdırılmalıdır. Bundan sonra bu qlobal element XML Sinif Qlobal Elementi (XCGE) adlandırılacaqdır.

NÜMUNƏ 1: Şəkil 1-də göstərilən **Class1** sinifinin XCGE-si belədir:

```
<xs:element name="Class1" type="ns1:Class1_Type"/>
```

7.2.2.3 XCGE nümunəsi

NÜMUNƏ 2: Şəkil 2-də göstərilən **DQ_DataQuality**-ə uyğun olan XCGE belədir:

```
<xs:element name="DQ_StandaloneQualityReportInformation"
  substitutionGroup="gco:AbstractObject"
  type="mdq:DQ_StandaloneQualityReportInformation_Type"/>
```

7.2.3 XML Sinif xassə Növü (XCPT)

7.2.3.1 XML Sinif xassə Növünə (XCPT) ümumi baxış

XCPT qaydası, 7.2.1-də təsvir olunan xassə konsepsiyasını dəstəkləmək və bir sinifin konteyner siniflər üçün xassə növü ola biləcəyini təmin etmək məqsədi daşıyır.

7.2.3.2 XCPT qaydası

Requirement /req/default/XCPT

xassə konsepsiyasını dəstəkləmək və bir sinifin konteyner siniflər üçün xassə növü ola biləcəyini təmin etmək üçün hər bir UML sinifi həmçinin **_PropertyType** şəkilçisi ilə adlandırılan **xs**

elementi kimi təyin olunmalıdır. Bu element **XML Sinif xassə Növü (XCPT)** adlandırılacaqdır.

Requirement /req/default/XCPT-name

a) XCT-lərdən fərqləndirmək üçün **xs:complexType** elementi sinif adının sonuna **_PropertyType** şəkilçisi əlavə edilmiş ad atributuna malik olmalıdır.

NÜMUNƏ 1: Şəkil 1-də göstərilən **Class1** sinifinin XCPT-sinin yaradılmasının 1-ci addımı belədir:

```
<xs:complexType name="Class1_PropertyType">
  (...)
</xs:complexType>
```

Requirement /req/default/XCPT-sequence

b) Polimorfizm tələblərini dəstəkləmək üçün XCPT-də opsional **xs:xs:element** elementi olmalıdır ki, bu da bir **xs:sequence** elementi ilə UML sinifinin XCT-sinə istinad etməyi təmin edir.

NÜMUNƏ 2: Şəkil 1-də göstərilən **Class1** sinifinin XCPT-sinin yaradılmasının 2-ci addımı belədir:

```
<xs:complexType name="Class1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:Class1"/>
  </xs:sequence>
</xs:complexType>
```

Requirement /req/default/XCPT-reference

c) "By Ref" konteynerini təmin etmək üçün XCPT-də **xs**

elementi olmalıdır və onun **ref** atributu **gco:ObjectReference** -ə bərabər olmalıdır.

NÜMUNƏ 3: Şəkil 1-də göstərilən **Class1** sinifinin XCPT-sinin yaradılmasının 3-cü addımı belədir:

```
<xs:complexType name="Class1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:Class1"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
</xs:complexType>
```

Requirement /req/default/XCPT-nilReason

d) Zəruri hallarda boş dəyər izahları təmin etmək üçün XCPT-də **xs**

elementi olmalıdır və onun **ref** atributu **gco:nilReason** -a bərabər olmalıdır.

NÜMUNƏ 4: Şəkil 1-də göstərilən **Class1** sinifinin XCPT-sinin yaradılmasının 4-cü addımı belədir:

```
<xs:complexType name="Class1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:Class1"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.2.3.3 XCPT nümunəsi

NÜMUNƏ 5: Şəkil 2-də göstərilən **DQ_StandAloneReportInformation**-a uyğun XCPT belədir:

```
<xs:complexType name="DQ_StandAloneQualityReportInformation_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="mdq:DQ_StandAloneQualityReportInformation"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.3 Xüsusi hal kodlaşdırmaları

7.3.1 Xüsusi hal kodlaşdırmalarına ümumi baxış

7.2-də təsvir olunan kodlaşdırma mexanizmləri ISO 19100 seriyasının UML-də olan siniflərin əksəriyyətini əhatə etsə də, bəzi xüsusi hallar var ki, fərqli kodlaşdırma qaydaları tətbiq olunur. İstisnalar ümumiyyətlə siniflərə tətbiq edilən xüsusi stereotiplərə əsaslanır. Stereotipi olmayan və ya **Type** və ya **Data Type** stereotipi olan siniflər standart kodlaşdırmalara uyğun gəlir. Digər siniflər 7.3.2-dən 7.3.6-ya qədər olan müvafiq kodlaşdırma qaydalarına uyğun gələcək.

Bəzi xüsusi XCT kodlaşdırmaları burada təsvir olunub ki, burada XCT **xs: simpleType**

kimi, 7.2-də təsvir olunan **xs:complexType** əvəzinə istifadə olunur. XCT sadə olduqda, müvafiq XCPT "By Ref" qabiliyyətlərini ehtiva etmir. Sadə növlərdə istinadlardan istifadəni məhdudlaşdırmaq, bütün xassə növlərinin istinadla həyata keçirildiyi təqdirdə yaranacaq potensial mürəkkəbliyi aradan qaldırır.

7.3.2 Abstrakt siniflər

7.3.2.1 Abstrakt sinifə ümumi baxış

Abstrakt siniflərin tətbiq edənlərin belə növlərdə elementlərin mövcud ola bilməyəcəyini anlamaları üçün ardıcıl şəkildə müəyyən edilməsi tövsiyə olunur.

7.3.2.2 Abstrakt sinif qaydası

Abstrakt siniflər standart qaydalarla kodlaşdırılır, lakin əlavə tələblər mövcuddur.

Şəkil 3 — Abstrakt sinif üçün UML nümunəsi

Requirement /req/abstract/XCT-name

Abstrakt sinifin adının qarşısına müvafiq XCT-də "Abstract" sözü əlavə olunmalıdır.

Requirement /req/abstract/XCT-abstract

XCT-nin **xs:complexType** elementinə **abstract="true"** atributu əlavə edilməlidir.

NÜMUNƏ 1: Şəkil 3-də göstərilən **AbsClass1** sinifinə uyğun olan XCT:

```
<xs:complexType name="AbstractAbsClass1_Type" abstract="true">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="attrX" type="ns1:typeAttrX_PropertyType"/>
        <xs:element name="attrY" type="ns1:typeAttrY_PropertyType"
minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Requirement /req/abstract/XCGE-name

Abstrakt sinifin adının qarşısına XCGE-də "Abstract" sözü əlavə olunmalıdır.

Requirement /req/abstract/XCGE-abstract

XCGE-nin **xs:element** elementinə **abstract="true"** atributu əlavə edilməlidir.

NÜMUNƏ 2: Şəkil 3-də göstərilən **AbsClass1** sinifinə uyğun olan XCGE:

```
<xs:element name="AbstractAbsClass1" type="ns1:AbstractAbsClass1_Type"
abstract="true"/>
```

Requirement /req/abstract/XCPT-ref

Müvafiq sinif adı **xs:element** -in **ref** atributunda istifadə olunmalıdır (Qeyd: XCPT-nin **name** atributuna "Abstract" sözü əlavə edilmir, çünki xassə növü XML sxemində abstrakt olmur).

NÜMUNƏ 3: Şəkil 3-də göstərilən **AbsClass1** sinifinə uyğun olan XCPT:

```
<xs:complexType name="AbsClass1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:AbstractAbsClass1"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.3.2.3 Abstrakt sinif nümunəsi

Yuxarıdakı qaydaları ISO 19115-1-dən olan UML nümunəsinə tətbiq etmək aşağıdakı XML-ə nəticə verir.

Şəkil 4 — ISO 19115-1-dən olan EX_GeographicExtent sinifinin UML-i

EX_GeographicExtent sinifinə uyğun olan XCT:

```

<xs:complexType name="AbstractEX_GeographicExtent_Type" abstract="true">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="extentTypeCode" type="gco:Boolean_PropertyType"
minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

EX_GeographicExtent sinifinə uyğun olan XCGE:

```

<xs:element name="AbstractEX_GeographicExtent"
type="gex:AbstractEX_GeographicExtent_Type"
abstract="true"/>

```

EX_GeographicExtent sinifinə uyğun olan XCPT:

```

<xs:complexType name="EX_GeographicExtent_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="gex:AbstractEX_GeographicExtent"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>

```

7.3.3 İrsi və alt sinif kodlaşdırmaları

7.3.3.1 İrsi və alt sinif kodlaşdırmasına ümumi baxış

İrsi konsepsiyası genişləndirmə olaraq sadə genişlənmələrlə həyata keçirilir. Hərçənd bu, irsi və alt sinifləndirmə konsepsiyasının sadələşdirilməsidir, çünki məhdudiyyətləri təmin edən ümumiləşdirmə əlaqələri və çoxlu irsi təmin edən ümumiləşdirmə əlaqələri nəzərə alınmır. Bu genişləndirmə yalnız bir dizayn xüsusiyyətidir. XML sxeminə gəldikdə isə çoxlu irsi dəstəklənmir və yalnız simulyasiya edilə bilər.

Varsayılan UML sinifləri kimi, alt siniflər üçün də uyğun XCT, XCGE və XCPT mövcuddur.

7.3.3.2 İrsi və alt sinif qaydası

Alt sinifin XCT-si varsayılan sinifdən aşağıdakı fərqlərə malikdir:

Requirement /req/inheritance/XCT

a) XCT-nin **xs:complexContent** elementi **xs:extension** elementini ehtiva etməlidir, onun **base** atributu isə əsas (yuxarı) sinifin namespace-ə uyğunlaşdırılmış XCT-sinə bərabər olmalıdır. (Bu, varsayılan UML sinifinin **gco:AbstractObject_Type** bazasına malik **xs:extension** elementindən fərqlənir). Əsas sinif abstrakt sinifdirsə, onun müvafiq XCT adına "**Abstract**" prefiksi əlavə olunacaq.

NÜMUNƏ 1: Şəkil 5-də göstərilən **SubClass1** sinifinin yaradılmasının a) addımı belədir:

```

<xs:complexType name="SubClass1_Type">
  <xs:complexContent>
    <xs:extension base="ns1:Class1_Type">
      (... )
    </xs:extension>

```

```
</xs:complexContent>
```

```
</xs:complexType>
```

Requirement /req/inheritance/XCT-sequence

b) **xs:extension** elementi sinfin bütün xassələri ni ehtiva edən **xs:sequence** elementini ehtiva etməlidir. Bu ardıcılıq yalnız alt sinfə xas olan xassələri ehtiva edir.

NÜMUNƏ 2: Şəkil 5-də göstərilən **SubClass1** sinifinin yaradılmasının a) addımı belədir:

```
<xs:complexType name="SubClass1_Type">
```

```
<xs:complexContent>
```

```
<xs:extension base="ns1:Class1_Type">
```

```
<xs:sequence>
```

```
<xs:element name="attr3" type="ns1:typeAttr3_PropertyType"/>
```

```
</xs:sequence>
```

```
</xs:extension>
```

```
</xs:complexContent>
```

```
</xs:complexType>
```

Requirement /req/inheritance/XCGE

Alt sinfin XCGE-sinin varsayılan UML sinifindən fərqi odur ki, o, **substitutionGroup** atributunu daxil etməlidir və bu, əsas sinfin namespace-ə uyğunlaşdırılmış XCGE-sinə bərabər olmalıdır.

NÜMUNƏ 3: Şəkil 5-də göstərilən **SubClass1** sinifinin XCGE-si belədir:

```
<xs:element name="SubClass1" type="ns1:SubClass1_Type" substitutionGroup="ns1:Class1"/>
```

Alt sinfin XCPT-si varsayılan UML sinifinin XCPT-si ilə eyni şəkildə kodlaşdırılır.

NÜMUNƏ 4: Şəkil 5-də göstərilən **SubClass1** sinifinin XCPT-si belədir:

```
<xs:complexType name="SubClass1_PropertyType">
```

```
<xs:sequence minOccurs="0">
```

```
<xs:element ref="ns1:SubClass1"/>
```

```
</xs:sequence>
```

```
<xs:attributeGroup ref="gco:ObjectReference"/>
```

```
<xs:attribute ref="gco:nilReason"/>
```

```
</xs:complexType>
```

7.3.3.3 İrsi və alt sinif nümunəsi

NÜMUNƏ 5: Faktiki ISO 19115-1 nümunəsindən istifadə edərək XML sxemində alt sinif nümunəsi.

Şəkil 6 — ISO 19115-1-dən olan EX_GeographicExtent və EX_GeographicDescription siniflərinin UML-i

NÜMUNƏ 6: EX_GeographicDescription sinifinə uyğun olan XCT:

```
<xs:complexType name="EX_GeographicDescription_Type">
  <xs:complexContent>
    <xs:extension base="gex:AbstractEX_GeographicExtent_Type">
      <xs:sequence>
        <xs:element name="geographicIdentifier" type="mcc:MD_Identifier_PropertyType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

NÜMUNƏ 7: EX_GeographicDescription sinifinə uyğun olan XCGE:

```
<xs:element name="EX_GeographicDescription" type="gex:EX_GeographicDescription_Type"
substitutionGroup="gex:AbstractEX_GeographicExtent"/>
```

NÜMUNƏ 8: EX_GeographicDescription sinifinə uyğun olan XCPT:

```
<xs:complexType name="EX_GeographicDescription_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="gex:EX_GeographicDescription"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.3.4 Enumeration kodlaşdırmaları

7.3.4.1 Enumeration kodlaşdırmalarına ümumi baxış

ISO 19100 seriyasının konseptual sxem dili iki növ sadalanmış tip dəstəkləyir ki, bunlar "bəyanatın etibarlı mnemonic identifikatorların siyahısını müəyyən edir (ISO 19103)." Bu iki sadalanmış növ **Enumeration** və **CodeList**-dir və bu iki tip arasındakı fərqi anlamaq vacibdir.

Enumeration açar sözü ilə olan sinif "müəyyən adlandırılmış ədədi dəyərlərin sabit siyahısıdır. Sadalanmış növə malik atributlar yalnız bu siyahıdan dəyərləri qəbul edə bilər." (ISO 19103).

<<enumeration>> sinfi yalnız dəyərlərin siyahısına heç bir genişlənmə tələb olunmadığı halda istifadə olunur; əks halda **<<CodeList>>** sinfi istifadə olunur.

Şəkil 7 — Enumeration stereotipli sinif üçün UML nümunəsi

7.3.4.2 Enumeration qaydası

Requirement/req/enumeration/XCT

<<enumeration>> sinfinin kodlaşdırılması **xs:simpleType** elementi ilə başlayan bir XCT-nin yaradılması ilə başlamalıdır, **name** atributu UML-dəki sinif adına bərabər olmalı və **_Type** şəkilçisi əlavə edilməlidir.

NÜMUNƏ 1: Şəkil 7-də göstərilən **EnumClass1** sinfinin yaradılmasının a) addımı belədir:

```
<xs:simpleType name="EnumClass1_Type">
```

```
(...)
```

```
</xs:simpleType>
```

Requirement /req/enumeration/XCT-string

xs:simpleType elementi bir **xs:restriction** elementi ehtiva etməlidir, onun **base** atributu **xs:string**-ə bərabər olmalıdır.

NÜMUNƏ 2: Şəkil 7-də göstərilən **EnumClass1** sinfinin yaradılmasının b) addımı belədir:

```
<xs:simpleType name="EnumClass1_Type">
```

```
<xs:restriction base="xs:string">
```

```
(...)
```

```
</xs:restriction>
```

```
</xs:simpleType>
```

Requirement /req/enumeration/XCT-enumerations

Hər **xs:restriction** elementi, UML modelində sadalanan dəyərlərə bərabər olan **xs:enumeration** elementlərinin bir sıra ehtiva etməlidir.

NÜMUNƏ 3: Şəkil 7-də göstərilən **EnumClass1** sinfinin yaradılmasının c) addımı belədir:

```
<xs:simpleType name="EnumClass1_Type">
```

```
<xs:restriction base="xs:string">
```

```
<xs:enumeration value="enumVal1"/>
```

```
<xs:enumeration value="enumVal2"/>
```

```
<xs:enumeration value="enumVal3"/>
```

```
</xs:restriction>
```

```
</xs:simpleType>
```

Requirement /req/enumeration/XCGE

<<enumeration>> sinfinin XCGE-si **/req/default/XCGE** qaydalarına uyğun olmalıdır və əlavə olaraq **substitutionGroup** atributu **"gco:CharacterString"** XCGE-yə bərabər olmalıdır.

NÜMUNƏ 4: Şəkil 7-də göstərilən **EnumClass1** sinfinin XCGE-si belədir:

```
<xs:element name="EnumClass1" type="ns1:EnumClass1_Type"
substitutionGroup="gco:CharacterString"/>
```

Requirement /req/enumeration/XCPT

<<enumeration>> sinfinin XCPT-si 7.3.1-də təsvir olunan sadə tip XCT-lər üçün standart kodlaşdırmaya uyğun olmalıdır.

NÜMUNƏ 5: Şəkil 7-də göstərilən **EnumClass1** sinfinin XCPT-si belədir:

```
<xs:complexType name="EnumClass1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:EnumClass1"/>
  </xs:sequence>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.3.4.3 Enumeration nümunəsi

Yuxarıdakı Enumeration kodlaşdırma qaydasını ISO 19115-1-dən olan bu UML nümunəsinə tətbiq etmək aşağıdakı XML-ə nəticə verir.

Şəkil 8 — ISO 19115-1-dən olan MD_ObligationCode sinfi

NÜMUNƏ 6: MD_ObligationCode sinfinə uyğun olan XCT:

```
<xs:simpleType name="MD_ObligationCode_Type">
  <xs:restriction base="xs:string">
    <xs:enumeration value="mandatory"/>
    <xs:enumeration value="optional"/>
    <xs:enumeration value="conditional"/>
  </xs:restriction>
</xs:simpleType>
```

NÜMUNƏ 7: MD_ObligationCode sinfinə uyğun olan XCGE:

```
<xs:element name="MD_ObligationCode" type="mex:MD_ObligationCode_Type"
substitutionGroup="gco:CharacterString"/>
```

NÜMUNƏ 8: MD_ObligationCode sinfinə uyğun olan XCPT:

```

<xs:complexType name="MD_ObligationCode_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="mex:MD_ObligationCode"/>
  </xs:sequence>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>

```

7.3.5 CodeList kodlaşdırılması

7.3.5.1 CodeList kodlaşdırılmasına ümumi baxış və əsas elementlər

CodeList stereotipi olan sinif **enumeration** açar sözü ilə olan sinif kimi sadalanmış bir tiptir. Fərq ondadır ki, <<**CodeList**>> sinfi genişlənə bilər.

<<**CodeList**>> sinifləri bəzən məlumat lüğətinin bir hissəsi kimi, iki sütunlu cədvəllərdə təsvir edilir: **Ad** və **Tərif**.

Cədvəl 2 — ISO 19115-1-dən olan CI_TelephoneTypeCode <<CodeList>> cədvəli

Ad	Tərif
CI_TelephoneTypeCode	Telefon növü
səs	Telefon səs xidməti göstərir
faks	Telefon faks xidməti göstərir
sms	Telefon SMS xidməti göstərir

CodeList cədvəlindəki hər bir sətir **CodeList**-in tərifinin məqsədinə xidmət edən unikal konsepsiya və ya variantı təsvir etmək üçün nəzərdə tutulmuşdur. **CodeList** stereotipinin xüsusi kodlaşdırılması yaradıcılar tərəfindən nəzərdə tutulan bəzi xüsusiyyətləri dəstəkləmək üçündür. Bu xüsusiyyətlər bunlardır:

- CodeList**-lər və onların əlaqəli tərifləri reyestrlərdə idarə olunur. Məlumat lüğəti **CodeList**-lərin əsas konsepsiya reyestridir və istifadəçi icmaları üçün reyestrlərin yaradılması mənbəyidir.
- CodeList** cədvəllərinin **Ad** sütunu, bir **CodeList** cədvəlindəki sətirdə müəyyən edilmiş hər bir unikal konsepsiyayı və ya variantı tanımaq üçün bütün proqram təminatı tərəfindən istifadə ediləcək dəyərləri ehtiva edir. Nəticə etibarilə, 'Ad' dəyəri hər hansı bir dil daxil olmaqla (ingilis dili daxil olmaqla) xüsusi ad kimi qəbul edilməməlidir.
- İstifadəçi icmaları **CodeList** yaradıcıları tərəfindən nəzərə alınmayan yeni konsepsiyalar və ya variantlar əlavə etmək məcburiyyətində ola bilərlər və bu genişlənmələrin istifadəçi icmaları reyestrlərində idarə edilməsi zəruridir. (Bu, məlumat lüğətində **CodeList**-ə əlavə sətirlər əlavə etməyə bənzəyir).

7.3.5.2 CodeList kodlaşdırma qaydası

Requirement /req/codelist/XCT

CodeList-in nəzərdə tutulmuş xüsusiyyətlərini təmin etmək üçün **gco** namespace-də iki atributlu xüsusi "**CodeListValue_Type**" XCT təqdim olunur. Aşağıda onun fraqmenti göstərilib:

```

<xs:complexType name="CodeListValue_Type">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="codeList" type="xs:anyURI" use="required"/>
      <xs:attribute name="codeListValue" type="xs:anyURI" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

```

Bu xüsusi XCT-nin tərifini hər hansı CodeList-in **CharacterString**-ə çevrilə biləcəyini təmin edir. 7.2.2-də təsvir edildiyi kimi, hər CodeList üçün bu xüsusi XCT-yə əsaslanan XCGE və XCPT müəyyən olunur. Nəticədə hər hansı CodeList nümunəsi XML elementidir. Onun adı XCGE-sinin adıdır və o, bir mətn dəyərini və xüsusi XCT tərəfindən müəyyən edilmiş iki XML atributunu ehtiva edir.

codeList atributu CodeList-in təfərrüatlarına, reyestrə və ya CodeList kataloqunda yerləşən URL-ə istinad etmək üçün nəzərdə tutulub.

NÜMUNƏ 1: Əgər Cədvəl 2 ISO veb saytında olan bir CodeList kataloqunda saxlanılırsa, **codeList** atributu üçün mümkün dəyər belə ola bilər:

```
codeList = "http://standards.iso.org/iso/19115/-1/cit/1.0/codelists.xml#CI_TelephoneTypeCode"
```

codeListValue atributu CodeList dəyərini təyin olunmuş identifikatorunu ehtiva etmək üçün nəzərdə tutulub. Bu identifikator, məlumat lüğəti cədvəllərinin **Ad** sütununda ifadə olunan dəyərdir. CodeList kataloqu (və ya reyestri) dəyərin defolt metadata dilində açıq adını və tərifini, eləcə də müxtəlif dillərdə ifadələri ehtiva edəcəkdir.

NÜMUNƏ 2: Əgər metadata nümunə sənədi "səs" telefon növünü göstərməli olarsa, **codeListValue** üçün dəyər (Cədvəl 2-yə əsasən) belə olardı:

```
codeListValue="voice"
```

Elementin məzmununu metadata-nın defolt dilində olan CodeList dəyərini adır.

NÜMUNƏ 3: Aşağıda **CI_TelephoneTypeCode**-dan bir CodeList dəyəri istifadə olunan metadata tətbiq nümunəsinin daha tam bir fraqmenti göstərilib:

```

<CI_TelephoneTypeCode codeList="http://standards.iso.org/iso/19115/-1/cit/1.0/codelists.xml#CI_TelephoneTypeCode"
codeListValue="voice">voice</CI_TelephoneTypeCode>

```

XML elementinin dəyəri istifadəçiyə CodeList-in faktiki dəyərini etibarlı defolt ifadəsinə çıxış imkanı verir, iki atribut isə tətbiqə CodeList-in və onun dəyərlərinin tam tərifinə çıxış imkanı verir.

7.3.5.3 CodeList kodlaşdırmasının təfərrüatları

CodeListValue_Type 7.3.5.2-də təqdim olunmuşdu və bu **complexType** bütün <<CodeList>> sinifləri üçün XCT olaraq xidmət edir.

Requirement /req/codelist/XCGE

<<CodeList>> sinfinin XCGE-si global element kimi kodlaşdırılmalı və onun **name** atributu UML sinfinin adına bərabər olmalıdır, növü isə **CodeListValue_Type**-ə bərabər olmalıdır.

<<enumeration>> sinfi kimi, bu XCGE də "**gco:CharacterString**" XCGE-yə bərabər **substitutionGroup** atributunu ehtiva etməlidir.

Şəkil 9 — CodeList stereotipli sinif üçün UML nümunəsi

NÜMUNƏ 1: Şəkil 9-da göstərilən **CodeListClass1** sinfinə uyğun olan XCGE belədir:

```
<xs:element name="CodeListClass1" type="gco:CodeListValue_Type"
substitutionGroup="gco:CharacterString"/>
```

NÜMUNƏ 2: Şəkil 9-da göstərilən **CodeListClass1** sinfinə uyğun olan XCPT belədir:

```
<xs:complexType name="CodeListClass1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:CodeListClass1"/>
  </xs:sequence>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

NÜMUNƏ 3: Faktiki ISO 19115-1 nümunəsindən istifadə edərək CodeList sinfinin kodlaşdırılması.

Şəkil 10 — ISO 19115-1-dən CI_TelephoneTypeCode sinfi

NÜMUNƏ 4: **CI_TelephoneTypeCode**-a uyğun olan XCGE belədir:

```
<xs:element name="CI_TelephoneTypeCode" type="gco:CodeListValue_Type"
substitutionGroup="gco:CharacterString"/>
```

NÜMUNƏ 5: **CI_TelephoneTypeCode**-a uyğun olan XCPT belədir:

```
<xs:complexType name="CI_TelephoneTypeCode_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="cit:CI_TelephoneTypeCode"/>
  </xs:sequence>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

</xs:complexType>

7.3.6 Union kodlaşdırması

7.3.6.1 Union kodlaşdırma qaydasına ümumi baxış

Union kodlaşdırma qaydası bir neçə element arasında seçim təklif edir.

7.3.6.2 Union kodlaşdırma qaydası

Requirement /req/union/XCT

Stereotipi **Union** olan bütün siniflər, Şəkil 11-də göstəriləyi kimi aşağıdakı şəkildə emal edilməlidir: — sinif, UML-dəki sinif adına bərabər **name** atributu və **_Type** şəkilçisi ilə **xs:complexType** vasitəsilə bir XCT-yə çevriləcəkdir.

xs:complexType elementinin daxilində hər bir UML sinif atributu üçün bir **xs:choice** ehtiva edən **xs:element** elementi olacaq.

Hər **xs:element** üçün:

- **name** atributu UML sinifindəki atributun adına bərabər olacaq; və
- **type** atributu atributun tipinə bərabər olacaq, lakin müvafiq namespace ilə prefikslənmiş və "**_PropertyType**" şəkilçisi ilə suffikslənmiş olacaq.

Şəkil 11 — Union stereotipli sinif üçün UML nümunəsi

NÜMUNƏ 1: Şəkil 11-də göstərilən **UnionClass1** sinfinə uyğun olan XCT belədir:

```
<xs:complexType name="UnionClass1_Type">
  <xs:choice>
    <xs:element name="attr4" type="ns1:typeAttr4_PropertyType"/>
    <xs:element name="attr5" type="ns1:typeAttr5_PropertyType"/>
  </xs:choice>
</xs:complexType>
```

Requirement /req/union/XCGE

<<**Union**>> sinfinin XCGE-si 7.2-də təsvir olunan standart kodlaşdırma qaydalarına uyğun gəlməlidir.

NÜMUNƏ 2: Şəkil 11-də göstərilən **UnionClass1** sinfinin XCGE-si belədir:

```
<xs:element name="UnionClass1" type="ns1:UnionClass1_Type"/>
```

Requirement /req/union/XCPT

Union istinadın istənilmədiyini başqa bir hal olduğuna görə, XCPT sadə növlər üçün standart kodlaşdırma qaydalarına uyğun gəlməlidir (7.3.1-də təsvir olunduğu kimi).

NÜMUNƏ 3: Şəkil 11-də göstərilən **UnionClass1** sinfinin XCPT-si belədir:

```
<xs:complexType name="UnionClass1_PropertyType">
  <xs:sequence minOccurs="0">
```

```
<xs:element ref="ns1:UnionClass1"/>
</xs:sequence>
<xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.3.6.3 Union kodlaşdırma nümunəsi

Yuxarıdakı Union kodlaşdırma qaydasını ISO 19115-1-dən olan UML Union nümunəsinə (Şəkil 12) tətbiq etmək aşağıdakı XML-ə nəticə verir.

Şəkil 12 — ISO 19115-1-dən MD_Resolution sinfi

NÜMUNƏ 5: MD_Resolution sinfinə uyğun olan XCT:

```
<xs:complexType name="MD_Resolution_Type">
  <xs:choice>
    <xs:element name="equivalentScale" type="mri:MD_RepresentativeFraction_PropertyType"/>
    <xs:element name="distance" type="gco:Distance_PropertyType"/>
    <xs:element name="vertical" type="gco:Distance_PropertyType"/>
    <xs:element name="angularDistance" type="gco:Angle_PropertyType"/>
    <xs:element name="levelOdDetail" type="gco:CharacterString_PropertyType"/>
  </xs:choice>
</xs:complexType>
```

NÜMUNƏ 6: MD_Resolution sinfinə uyğun olan XCGE:

```
<xs:element name="MD_Resolution" type="mri:MD_Resolution_Type"/>
```

NÜMUNƏ 7: MD_Resolution sinfinə uyğun olan XCPT:

```
<xs:complexType name="MD_Resolution_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="mri:MD_Resolution"/>
  </xs:sequence>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.3.7 MetaClass-ların kodlaşdırılması

MetaClass-lar yalnız onların reallaşdırılmalarından biri vasitəsilə tətbiq edilə bilər. Aşağıda reallaşdırılmaların kodlaşdırılması və istifadəsi daha ətraflı müzakirə edilir.

Şəkil 13 — MetaClass stereotipli sinif üçün UML nümunəsi

<<MetaClass>> reallaşdırılması, standart kodlaşdırılmış XCT-nin ardıcılığında bir element üçün **type** atributunu təyin edərkən istifadə olunur. Başqa sözlə, Şəkil 13-dən olan **Class1** sinfinin XCT-sində "attr1" elementinin tipi **MetaClass1_PropertyType** deyil, **ns1:externalImpl** olardı, çünki **MetaClass1** ilə **ns1:externalImpl** arasındakı reallaşdırma göstərilmişdir.

NÜMUNƏ: Şəkil 13-də göstərilən **Class1** sinfinə uyğun olan XCT belədir:

```
<xs:complexType name="Class1_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="attr1" type="ns1:externalImpl_PropertyType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

7.3.8 Xaricdən müəyyən edilmiş tətbiqlərin kodlaşdırılması

7.3.8.1 Xaricdən müəyyən edilmiş tətbiqlərə ümumi baxış

Xaricdən müəyyən edilmiş tətbiqlərin istifadəsi, bu sənədin qarşılıqlı əlaqə qabiliyyətini və istifadəyə yararlığını artırmaq məqsədilə 6.5-də təqdim edilmişdi. Artıq mövcud olan ISO 19100 seriyasının Beynəlxalq Standartlarınınin tətbiqlərindən istifadə etmək və ya müəyyən bir kodlaşdırma texnologiyasına xas olan xarici tətbiqlərdən istifadə etmək mümkündür.

7.3.8.2 Kodlaşdırma seçimləri

ISO 19100 seriyasının UML modellərində göstərilən konsepsiyalara xarici tətbiqləri daxil etməyin üç yolu vardır. ISO 19100 seriyasının UML siniflərinin adları tətbiq sxeminə giriş nöqtəsi kimi qorunur və bu, reallaşdırmanın stereotipinə əsaslanır (ISO 19103-də təsvir edildiyi kimi).

7.3.8.2.1 XCPT vasitəsilə kodlaşdırma

Requirement /req/external/XCPT

Xarici kodlaşdırmadan istifadə etməyin ən sadə halı, mövcud tətbiqin ISO 19100 seriyasının UML siniflərinə uyğun olan sinif növləri, qlobal elementlər və sinif xassə növləri təqdim etdiyi zamandır. Bu UML-də, reallaşdırma əlaqəsində bir XCPT stereotipinin olması ilə göstərilir (Şəkil 14-də olduğu kimi).

Şəkil 14 — ISO 19103 tarixinin gco:Date_PropertyType ilə reallaşdırılması

Bu vəziyyətdə, hədəf sinif üçün XCT-lər, XCGE-lər və ya XCPT-lər yaradılmır. Hədəf sinif bir XCT-nin xassəsi üçün növ olarsa, mənbə sinifinin adı **type** atributunun dəyəri kimi istifadə olunmalıdır.

Şəkil 15 — CI_Date sinfi (ISO 19115-1-dən)

NÜMUNƏ: Şəkil 15-də göstərilən **CI_Date** sinfinə uyğun olan XCT belədir:

```
<xs:complexType name="CI_Date_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="date" type="gco:Date_PropertyType"/>
        <xs:element name="dateType" type="cit:CI_DateTypeCode_PropertyType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

7.3.8.2.2 XCGE vasitəsilə kodlaşdırma

Requirement /req/external/XCGE

Xarici kodlaşdırmadan istifadə etməyin növbəti halı, mövcud tətbiq sinif növləri və qlobal elementlər təqdim etdiyi, lakin xarici tətbiqin irsi ağaclarının qorunmasının vacib olduğu zamandır.

Bu halda, aşağıdakı qaydalar tətbiq edilməlidir:

a) **XCPT** ISO 19100 seriyasının UML modellərindən sinif adı ilə yaradılacaq. Bu XCPT tətbiq sxeminə, xarici tətbiqin XCGE-si vasitəsilə giriş nöqtəsi kimi xidmət edir. Bu UML-də, reallaşdırma əlaqəsində bir **XCGE** stereotipinin olması ilə göstərilir (aşağıda Şəkil 16-da göstərilirdiyi kimi).

Şəkil 16 — ISO 19107 GM_Point-in gml:Point ilə reallaşdırılması

b) Reallaşdırmada **XCGE** stereotipi olduğda, hədəf sinif üçün XCPT, **ref** atributunun dəyəri kimi reallaşdırmanın mənbə sinifinin adını istifadə edərək kodlaşdırılacaqdır.

NÜMUNƏ 1: Şəkil 16-ya əsaslanan **GM_Point** üçün XCPT belədir:

```
<xs:complexType name="GM_Point_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="gml:Point"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

Requirement /req/external/XCGE-element

Bəzi hallarda, xarici tətbiqi təmsil edən UML **xs:element** stereotipi olan siniflərin xassələri ni ehtiva edə bilər. Bu halda, UML-də göstərilən sinifə uyğun olan XCGE olacaq və bu sinifin adında istifadə edilən namespace prefiksi ilə göstərilən namespace-də yerləşəcəkdir.

NÜMUNƏ 2: Xarici tətbiqdən bir **xs:element** əsasında XCGE nümunəsi.

Şəkil 17 — gco:Unlimited -in xs:element stereotipli sinif kimi göstərildiyi XCGE

NÜMUNƏ 3: **UnlimitedInteger**-a uyğun olan XCGE belədir:

```
<xs:element name="UnlimitedInteger" type="gco:UnlimitedInteger_Type" nillable="true"/>
```

7.3.8.2.3 XCT vasitəsilə kodlaşdırma

7.3.8.2.3.1 Ümumi qayda

Tez-tez, xarici tətbiqdən yalnız bir XCT təmin olunur. Bu, XML sxemi **simpleTypes**-dan istifadə edildiyi hallarda baş verir. Bu vəziyyətdə, ISO 19100 seriyasından sinif ilə XML sxemi obyekt arasında olan reallaşdırma **XCT** stereotipini daşıyır.

7.3.8.2.3.2 xs stereotipi

Şəkil 18 — CharacterString-in xs: string ilə reallaşdırılması

7.3.8.2.3.4 xs:complexType stereotipi

Requirement /req/external/XCT-complexType

Reallaşdırma bir **XCT** stereotipinə malik olduqda və mənbə sinif bir **xs:complexType** stereotipinə malik olduqda, mənbə sinifə uyğun bir **XCT** yaradılmalıdır. **XCT** aşağıdakı tələblərə uyğun olmalıdır:

- xs:complexType** yaradılmalı və **name** atributu XML sinfinin adından namespace prefiksini çıxararaq müəyyən edilməlidir. Əgər XML sinfi abstrakt olarsa, **xs:complexType** elementi **abstract="true"** atributunu ehtiva etməlidir;
- xs:complexType** bir **xs:complexContent** ehtiva edəcək;
- xs:complexContent** bir **xs:extension** ehtiva edəcək və onun **base** atributu xarici tətbiqin yuxarı sinfinin adına uyğun olmalıdır. Əgər xarici tətbiq üçün yuxarı sinif yoxdursa, **base** atributu **gco:AbstractObject**-a bərabər olmalıdır;
- xs:extension** **xs:element** stereotipi olan XML sinfinin xassələrinə əsaslanan **xs:elements**-dən ibarət **xs:sequence** ehtiva edəcək.

xs:element stereotipi olan hər hansı xüsusiyyət **xs:sequence** ilə **xs:element**-in olmasını göstərir:

- 1) xassənin adına bərabər olan ad atributu və
- 2) tip atributunun xassə növünə bərabər olması.

7.2-dən standart XCT kodlaşdırmalarında olduğu kimi, **minOccurs** və **maxOccurs** dəyərləri ISO 19118:2011, Cədvəl A.5-ə əsaslanır;

e) **xs:sequence** elementindən sonra, həmçinin **xs:atribut** və ya **xs:attributeGroup** stereotipi olan XML sinfinin hər hansı xassələrinə əsaslanan **xs:atribut** və ya **xs:attributeGroup** baş verə bilər. Əgər UML

elementində bir növ dəyəri varsa, o zaman xs:atribut və ya xs:attributeGroup aşağıdakılara malik olacaq:

- 1) UML-dəki elementin adına bərabər ad atributu və
- 2) UML-də göstərilən tipə bərabər tip atributu.

Əgər UML-də element üçün heç bir növ mövcud deyilsə, o zaman xs:atribut və ya xs:attributeGroup UML-dəki elementin adına bərabər dəyərə malik ref atributunu ehtiva etməlidir.

7.2-dən olan standart XCT kodlaşdırmalarında olduğu kimi, minOccurs və maxOccurs dəyərləri ISO 19118:2011, Cədvəl A.5-ə əsaslanır.

NÜMUNƏ: Şəkil 20-də göstərilən **PT_FreeText** sinfinin **XCT**-si:

```
<xs:complexType name="PT_FreeText_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="textGroup" type="lan:LocalisedCharacterString_PropertyType"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

7.3.8.2.3.5 xs

stereotipi

Requirement /req/external/XCT-union

Bir sinif **xs:union** stereotipinə malik olduqda, aşağıdakı qaydalar tətbiq edilməlidir:

- b) XCT UML sinfinin adına bərabər ad atributuna malik xs:simpleType olacaq.
- c) xs:simpleType daxilində, boşluqlarla ayrılmış UML sinfindəki elementlərin adlarına bərabər üzv atributuna malik bir xs:union olacaq.

NÜMUNƏ GCO ad məkanında xs:union üçün XCT nümunəsi

NÜMUNƏ: Şəkil 21-də göstərilən **Date** üçün **XCT** belədir:

```
<xs:simpleType name="Date_Type">
  <xs:union memberTypes="xs:date xs:gYearMonth xs:gYear"/>
</xs:simpleType>
```

7.3.8.2.4 XCT vasitəsilə kodlaşdırmada XCGE-nin yaradılması

Requirement /req/external/XCGE

Xarici tətbiq **XCT**-ləri üçün bir **XCGE** yaradılmalıdır. **XCGE**-lər, abstrakt siniflər üçün (7.3.2-də) və alt siniflər üçün (7.3.3-də) təsvir edilən standart kodlaşdırma qaydalarına uyğun gəlir.

7.3.8.2.5 XCPT-nin yaradılması

Requirement /req/external/XCPT

Bir **XCPT** də yaradılmalıdır və o, aşağıdakı hallarda 7.2.3-dəki kodlaşdırma qaydalarına uyğun gəlməlidir:

- reallaşdırmanın mənbə sinfi **xs:complexType** stereotipinə malikdirsə, və ya
- (7.3.1-dəki kodlaşdırma qaydalarına uyğun olaraq), reallaşdırmanın mənbə sinfi **xs:simpleType** və ya **xs:simpleContent** stereotipinə malikdirsə.

NÜMUNƏ 1: Şəkil 32-yə əsaslanan **CharacterString** üçün **XCPT** belədir:

```
<xs:complexType name="CharacterString_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="gco:CharacterString"/>
  </xs:sequence>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

Şəkil 22 — xs

stereotipli sinif üçün UML nümunəsi

NÜMUNƏ 2: Şəkil 22-yə əsaslanan **Date_PropertyType** üçün **XCPT** belədir:

```
<xs:complexType name="Date_PropertyType">
  <xs:choice minOccurs="0">
    <xs:element ref="gco:Date"/>
    <xs:element ref="gco:DateTime"/>
  </xs:choice>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

7.4 XML namespace paketinin kodlaşdırılması

xmlNamespace stereotipi 4.3-də bir namespace daxilində qruplaşdırılmış XML obyektləri dəsti kimi təqdim edilmişdir. Bir UML paketi iki modelləşdirmə elementi arasında bir əlaqə kimi təyin olunan asılılıqlara malik ola bilər. **xmlNamespace** paketləri arasında **implement** və **import** asılılıqları mövcud ola bilər.

7.5 XML sxem paketi kodlaşdırılması

xmlSchema stereotipli paketlər XML sxemini təmsil edən paketlərdir. Hər hansı bir **<<xmlSchema>>** paketi üçün, UML-dəki bu paketlə eyni ada malik bir XML sxem faylı mövcud olmalıdır. **<<xmlSchema>>** paketi **<<implement>>** və **<<include>>** asılılıqlarından istifadə edə bilər. **<<include>>** asılılığı eyni namespace-dən bir XML sxem faylını istinad etmək üçün istifadə olunur. Hər **<<xmlSchema>>** paketi iki növdən biridir.

Birinci növ, birbaşa <<xmlNamespace>> paketinə uyğun gələn və .xsd genişlənməsinin əlavə olduğu <<xmlNamespace>> paketi ilə eyni ada malik olan kök <<xmlSchema>> paketidir. Bu o deməkdir ki, Şəkil 24-də göstərilən mri <<xmlNamespace>> paketi olduğundan xmlSchema stereotipinə malik müvafiq mri.xsd paketi və müvafiq olaraq bu sənəddə müəyyən edilmiş kodlaşdırma qaydalarından istifadə etməklə yaradılmış mri.xsd faylı da olacaqdır. .

Hər bir XML sxem faylı bu XML bəyannaməsi ilə başlayır: <?xml version="1.0" encoding="utf-8"?>. Eyni fayldakı kök element xs:schema-dır və Cədvəl 3-də göstərilən atributlara malik olmalıdır.

Cədvəl 3 — Kök <<xmlSchema>> paketi üçün xs:schema elementinin atributları

Atributun baş verməsi

Atributun ad sahəsi

Atributun adı

Atributun dəyəri

Nümunə atribut-dəyər cütü

QEYD 1 Cədvəl 3-dəki paketAdı kök <<xmlSchema>> paketinin müvafiq <<xmlNamespace>> paketinin adına istinad edir.

NÜMUNƏ Şəkil 25-də göstərilən UML aşağıdakı xs:schema bəyannaməsi ilə mri.xsd adlı faylın mövcudluğu ilə nəticələncək:

```
<xs:schema targetNamespace=http://standards.iso.org/iso/19115/-3/mri/1.0
```

```
xmlns:lan="http://standards.iso.org/iso/19115/-3/lan/1.0"
```

```
xmlns:mcc=http://standards.iso.org/iso/19115/-3/mcc/1.0
```

```
xmlns:mri="http://standards.iso.org/iso/19115/-3/mri/1.0" version="1.0"
```

```
xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

İkinci növ <<xmlSchema>> paketi adı <<xmlSchema>> paketi adlanır və o, birbaşa ISO 19100 seriyasından olan mövcud paketə uyğun gəlir.

stereotip tətbiqindən asılılıq. Şəkil 25-dəki nümunə adı <<xmlSchema>> paketini və onun hədəf ISO 19115-1 paketini göstərir.

Şəkil 25 — Adı xmlSchema paketi

Bütün adı <<xmlSchema>> paketlərinin müvafiq bir kök <<xmlSchema>> paketi var ki, bu ya adı <<xmlSchema>> paketi ilə <<include>> asılılığı arasında birbaşa mənbədir, ya da maraq doğuran adı <<xmlSchema>> üçün <<include>> asılılığının birbaşa mənbəyi olan adı <<xmlSchema>>-nin kök <<xmlSchema>>-sidir. Şəkil 26-da kök <<xmlSchema>> root.xsd-dir və adı <<xmlSchema>>-lər isə schema1.xsd və schema2.xsd-dir. root.xsd <<xmlSchema>> iki adı <<xmlSchema>>-nin köküdür.

Şəkil 26 — <<xmlSchema>> paketlərinin iyerarxiyası
Hər bir adi <<xmlSchema>> paketinə əsasən yaradılmış XML sxem faylındakı XML bəyanatı belədir:

```
<?xml version="1.0" encoding="utf-8"?>
```

Eyni fayldakı kök element isə **xs**

olacaq və bu elementin atributları Cədvəl 4-də göstərilmişdir.

Cədvəl 4 — Adi <<xmlSchema>> paketi üçün xs:schema elementinin atributları

Qeyd 2: Cədvəl 4-dəki **rootPackageName** kök <<xmlSchema>> paketinə uyğun gələn <<xmlNamespace>> paketinin adına istinad edir.

8 Əlavə kodlaşdırmalar

Mövcud yanaşma və texnologiyalardan maksimum istifadə etmək üçün bəzi siniflər ISO və ya digər təşkilatların müvafiq tətbiqlərindən istifadə edərək xüsusi olaraq kodlaşdırılıb. Bu kodlaşdırmalar aşağıdakı ünvanda mövcuddur:

<http://standards.iso.org/iso/19139/resources>.

9 Modulluq və təkrar istifadə üçün kodlaşdırma

9.1 UML paketləri və XML namespace-ləri

UML modellərini XML sxemlərinə kodlaşdırarkən UML paketləri ilə XML namespace-ləri arasında əlaqələr haqqında qərarlar verilməlidir. Bu prosesdə aşağıdakı qaydalar tətbiq olunur:

1. XML tətbiqi konseptual modeldə hər UML paketi üçün minimum bir namespace ehtiva etməlidir, yəni bir neçə UML paketi bir namespace-də birləşdirilməməlidir.
2. Zərurət yarandıqda, UML paketləri müxtəlif komponentlərin idarə olunmasını asanlaşdırmaq üçün bir neçə namespace-ə bölünə bilər.
3. Namespace-lər arasında asılılıqları minimuma endirmək və dövrü asılılıqları aradan qaldırmaq üçün qayda 1-ə istisnalar tətbiq oluna bilər.

9.2 UML modeli üçün XML tətbiqi

Fərqli UML paketlərindəki siniflər arasındakı əlaqələr, paketlər arasında asılılıqlara səbəb olur ki, bu da iki paketin birlikdə daxil edilmədən təkrar istifadəsini mümkün deyil edir. Bu əlaqələri idarə etmək və modulizasiya və avtomatlaşdırılmış sxem yaradılmasını asanlaşdırmaq üçün UML modelinə XML-ə xas bir tətbiq təbəqəsi əlavə edilə bilər, semantikaya təsir etmədən. Bu təbəqə aşağıdakılardan ibarətdir:

1. Model paketlərini bir-birindən ayırmağa imkan verən abstrakt siniflər,
2. UML-dən XML-ə çevirmə proqram təminatının tələb etdiyi etiketlenmiş dəyərlərin və stereotiplərin əlavə edilməsi,

3. XML namespace-ləri arasında dövrü asılılıqları aradan qaldırmaq üçün bəzi model paketlərinin refaktoru,
4. Digər ISO modellərinin XML tətbiqləri tərəfindən istifadə edilən və ya istifadə oluna bilən siniflər üçün əvəzetmə qruplarını təyin etmək üçün yaradılmış abstrakt siniflər.

Abstrakt siniflər bir namespace-də yerləşdirilir.

Requirement /req/modularity/

Bu tətbiq modelinə əsasən XML sxemi bu sənəddə müəyyən edilmiş qaydalara uyğun olaraq avtomatik şəkildə yaradılmalıdır.

9.3 XML paketlərinin ayrılması üçün tətbiq yanaşması

9.3.1 Ümumi baxış

Bu tətbiq yanaşması, opsional bir xassənin və ya əlaqənin xassə növünün, yerləşdirici elementdən fərqli bir paketdən olan bir sinif olduğu hallarda tələb olunur. XML nümunə sənədləri, opsional xassə növü sinfini həyata keçirən namespace-i idxal etmədən də doğrulanmalıdır. Bu, sxemdə xassə dəyəri növü üçün abstrakt sinifdən istifadə etməklə və həmin xassə dəyəri sinfinin real tətbiqini təmin edən uşaq sinfi üçün müvafiq əvəzetmə yaratmaqla həyata keçirilir.

9.3.2 Tətbiq yanaşmasının qaydaları

Requirement /req/modularity/decouple

Opsional xassə digər bir namespace-dən olan bir sinif növündə olduqda, namespace-lər ayrılmalıdır. Şəkil 27-də **Class1 ns1** namespace-də yerləşir və opsional xassə olaraq **Class2** sinfini ehtiva edir, bu sinif isə **ns2** namespace-də müəyyən edilir.

9.3.2.1 Abstrakt ümumi siniflər namespace-i

Tətbiqdə ilk addım modulluq üçün zəruri olan abstrakt sinifləri saxlayan yeni bir namespace yaratmaqdır.

Requirement /req/modularity/abstractCommonClassNamespace

Modulluq üçün tələb olunan abstrakt elementlər **Metadata Common Classes (mcc)** namespace-də müəyyən edilməlidir. Bu namespace abstrakt sinifləri birləşdirən sxemləri müəyyən edir və XML nümunə sənədləri bu paketdən heç bir element ehtiva etməyəcəkdir.

Abstrakt sinif üçün XCGE:

```
<xs:element abstract="true" name="AbstractClass2"
  substitutionGroup="gco:AbstractObject" type="mcc:AbstractClass2_Type"/>
```

Abstrakt sinif üçün XCT:

```
<xs:complexType abstract="true" name="AbstractClass2_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence/> <!-- no content -->
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Abstrakt sinif üçün XCPT:

```
<xs:complexType name="AbstractClass2_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="mcc:AbstractClass2"/>
  </xs:sequence>
```

```

    <xs:attributeGroup ref="gco:ObjectReference"/>
    <xs:attribute ref="gco:nilReason"/>
</xs:complexType>

```

9.3.2.2 ns1 üçün sxem

ns1 namespace-nin sxemi **mcc** namespace-i idxal edir və opsional xassə növünü abstrakt sinif növü kimi müəyyən edir. **Class1** üçün XCT:

```

<xs:complexType name="Class1_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element maxOccurs="unbounded" minOccurs="0" name="property"
          type="mcc:AbstractClass2_PropertyType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

9.3.2.3 ns2 üçün sxem

ns2 namespace-nin sxemi, abstrakt sinif üçün konkret XML elementini müəyyən edir:

```

<xs:element name="Class2" substitutionGroup="mcc:AbstractClass2"
  type="ns2:Class2_Type"/>
<xs:complexType name="Class2_Type">
  <xs:complexContent>
    <xs:extension base="mcc:AbstractClass2_Type">
      <xs:sequence>
        <!--definition of elements in concrete instance... -->
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

9.3.3 Ayrılma nümunəsi

Bu nümunə ISO 19115-1-dən olan **MD_Metadata** sinfi ilə izah olunur. **MD_Metadata** sinfi opsional **resourceLineage** xassəsinə malikdir, bu xassənin növü **LI_Lineage** sinfi ilə müəyyən edilir. Bu əlaqə abstrakt sinifdən istifadə edərək həyata keçirilir ki, bu da namespace-lərin tam ayrılmasını təmin edir.

Şəkil 28 və Şəkil 29-da bu nümunə daha ətraflı izah edilmişdir.

Şəkil 28 — Fərqli bir paketdə təyin edilmiş məlumat növü sinfinə malik opsional xassə nümunəsi

Bu nümunə **MD_Metadata** sinfi üçün olan XML nümunə sənədlərinin xassə növü sinfinə (burada **LI_Lineage**) ehtiyac olduğu hallar üçün modulluq və reuse-u asanlaşdırmaq üçün təqdim edilmişdir. **MD_Metadata** sinfi **Metadata Base (mdb)** namespace-ə daxil edilir və **Metadata Common Classes (mcc)** namespace-dən abstrakt sinif olan **_LineageInformation** sinfini idxal edir. Bu abstrakt sinfin konkret tətbiqi isə **Metadata for Resource Lineage (mrl)** namespace-də müəyyən edilmiş **LI_Lineage** sinfidir.

Şəkil 29 — Fərqli bir paketdə olan məlumat növü sinfinə malik opsional xassələr üçün tətbiq modeli

MD_Metadata sinfi (şəkildə göstərilirdiyi kimi) opsional **resourceLineage** xassəsinə malikdir, və bu xassə **mcc** namespace-dən olan abstrakt siniflə əlaqələndirilir. **LI_Lineage** sinfi isə **mrl** namespace-də həmin abstrakt sinfin yerinə əvəz edici olaraq istifadə olunur.

XML nümunə sənədləri yalnız **LI_Lineage** məlumatlarını ehtiva etdikləri halda **mrl** namespace-ni idxal etməlidirlər.

NÜMUNƏ 1: Əsas metadata sinfi ilə abstrakt xassə növü arasındakı əlaqə:

```
<xs:import namespace="http://standards.iso.org/iso/19115/-3/mcc/1.0/"
  schemaLocation="../../mcc/1.0/mcc.xsd"/>
<xs:element name="MD_Metadata" substitutionGroup="gco:AbstractObject"
  type="mdb:MD_Metadata_Type"/>
<xs:complexType name="MD_Metadata_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element maxOccurs="unbounded" minOccurs="0"
name="resourceLineage"
          type="mcc:Abstract_LineageInformation_PropertyType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

NÜMUNƏ 2: Abstrakt sinif üçün abstrakt elementin təyini:

```
<xs:element abstract="true" name="Abstract_LineageInformation"
  substitutionGroup="gco:AbstractObject"
  type="mcc:Abstract_LineageInformation_Type"/>
```

NÜMUNƏ 3: Abstrakt sinif üçün XML sinif növü (XCT):

```
<xs:complexType abstract="true" name="Abstract_LineageInformation_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence/> <!-- no content -->
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

NÜMUNƏ 4: Abstrakt sinif üçün XML sinif xassə növü (XCPT):

```
<xs:complexType name="Abstract_LineageInformation_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="mcc:Abstract_LineageInformation"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

NÜMUNƏ 5: Abstrakt sinif üçün əvəzedici konkret sinfin (mrl-də) təyini:

```
<xs:element name="LI_Lineage"
  substitutionGroup="mcc:Abstract_LineageInformation"
  type="mrl:LI_Lineage_Type"/>
<xs:complexType name="LI_Lineage_Type">
  <xs:complexContent>
    <xs:extension base="mcc:Abstract_LineageInformation_Type">
      <xs:sequence>
        <!--definition of elements in concrete instance... -->
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
```

</xs:complexType>

Əlavə A: Normativ Abstrakt Test Dəsti

UML modellərindən yaradılan sxemlər, bu sənəddə təsvir edilmiş kodlaşdırma qaydalarına uyğun olaraq bütün siniflər və xassələri ehtiva etməlidir. Bu qaydalar avtomatik test üçün uyğun deyil, ancaq minimum uyğunluq tələbi, coğrafi resurslar üçün sxemlərin XSD sənədləri kimi etibarlı olmasını tələb edir.

Əlavə B: Məlumatların Uyğunluğu

Bu əlavə, sənədin 2006-cı il nəşrindən olan əsas dəyişiklikləri izah edir. Dəyişikliklər ISO 19115-1:2014 ilə əlaqəli yeniliklər, tələb olunan qaydalar və ISO/TS 19115-3:2016-da təkmilləşdirilmiş sxemlər ilə bağlıdır.

- 2006-cı il nəşrinin 7-ci bəndində təqdim edilmiş “ISO 19100 seriyasının Beynəlxalq Standartlarındakı UML modellərinə əlavələr” və 2006-cı il nəşrinin **B Əlavəsi**, "Əlavələr üçün Məlumat Lüğəti" ISO 19115-1:2014 tərəfindən əvəz edilmişdir.
- 2006-cı il nəşrinin 8-ci bəndində təsvir edilmiş kodlaşdırma qaydaları təsviri xarakter daşıyırdı. Onlar bu sənədin 7-ci bəndində tələblərə çevrilmişdir.
- 2006-cı il nəşrinin 9-cu bəndində təsvir edilmiş XSD fayllar ISO/TS 19115-3:2016-nın XML sxemi və **C Əlavəsi** tərəfindən əvəz edilmişdir və bu məlumatlar standards.iso.org/iso/19115/-3-da mövcuddur.
- 2006-cı il nəşrinin **C Əlavəsində** təsvir edilmiş XML sxemi və digər XML resursları ISO 19115-3:2016 ilə əvəz olunmuşdur.
- 2006-cı il nəşrinin **D Əlavəsində** təqdim edilmiş tətbiq nümunələri ISO 19115-3:2016-nın **D Əlavəsində** olan nümunələrlə əvəz edilmişdir.

İstifadə olunmuş ədəbiyyat

- [1] ISO 19103:2015, Geographic information — Conceptual schema language
- [2] ISO 19105, Geographic information — Conformance and testing
- [3] ISO 19107, Geographic information — Spatial schema
- [4] ISO 19110, Geographic information — Methodology for feature cataloguing
- [5] ISO 19115-1, Geographic information — Metadata — Part 1: Fundamentals
- [6] ISO/TS 19115-3, Geographic information — Metadata — Part 3: XML schema implementation for fundamental concepts
- [7] ISO 19119, Geographic information — Services
- [8] ISO 19136, Geographic information — Geography Markup Language (GML)
- [9] ISO 19157, Geographic information — Data quality
- [10] ISO/TS 19157-2, Geographic information — Data quality — Part 2: XML schema implementation

ICS 35.240.70

Qiymət 40 səhifə üçün hesablanır

© ISO 2019 – Bütün hüquqlar qorunur

Laqvinə Əliyeva