

BEYNƏLXALQ
STANDART

ISO
19118

İkinci nəşr
2011-10-15

Coğrafi məlumat — Kodlaşdırma

İnformasiya coğrafiyası — Kodaj



İstinad nömrəsi
ISO 19118:2011(E)

© ISO 2011

Latvian



MÜƏLLİF HÜQUQU QORUNAN SƏNƏD

© ISO 2011

Bütün hüquqlar qorunur. Başqa cür göstərilməyi təqdirdə, bu nəşrin heç bir hissəsi aşağıdakı ünvanda ISO-nun və ya aşağıdakı ölkədəki ISO-nun üzv orqanının yazılı icazəsi olmadan hər hansı formada və ya hər hansı üsulla, elektron və ya mexaniki, o cümlədən fotokopiya və mikrofilmlə çoxalda və ya istifadə edilə bilməz. tələb edən.

ISO müəllif hüquqları ofisi
Case postale 56 - CH-1211 Cenevrə 20
Tel. + 41 22 749 01 11
Faks + 41 22 749 09 47 E-
poçt copyright@iso.org
Veb www.iso.org

İsveçrədə nəşr edilmişdir

İçindəkilər

Səhifə

Ön söz.....	iv
Giriş.....	1
Əhatə dairəsi.....	1
2 Uyğunluq.....	1
Giriş.....	1
2.1 Kodlaşdırma qaydaları ilə bağlı uyğunluq sinifləri.....	1
2.2 Kodlaşdırma xidmətləri ilə bağlı uyğunluq sinifləri.....	1
2.3 Normativ istinadlar.....	2
3 Terminlər və təriflər.....	2
4 Simvollar və qısaldılmış terminlər.....	6
5 Əsas anlayışlar və fərziyyələr.....	7
6 Konsepsiyalar.....	7
6.1 Məlumat mübadiləsi.....	7
6.2 Tətbiq sxemi.....	8
6.3 Kodlaşdırma qaydası.....	9
6.4 Kodlaşdırma xidməti.....	10
6.5 Transfer xidməti.....	10
6.6	
7 Xarakter repertuarı.....	11
8 Ümumi nümunə modeli.....	11
8.1 Giriş.....	11
8.2 UML ilə nümunə modeli arasında əlaqə.....	14
9 Kodlaşdırma qaydaları.....	14
9.1 Giriş.....	14
9.2 Ümumi kodlaşdırma tələbləri.....	15
9.3 Daxil edilən məlumatların strukturu.....	17
9.4 Çıxış məlumatlarının strukturu.....	17
9.5 Dönüşüm qaydaları.....	18
9.6 Nümunələr.....	18
10 Kodlaşdırma xidməti.....	18
Əlavə A (normativ) XML əsaslı kodlaşdırma qaydası.....	20
Əlavə B (normativ) Abstrakt test kostyumu.....	21
Əlavə C (məlumatlandırıcı) İcmalar tərəfindən istifadə edilən XML əsaslı kodlaşdırma qaydası.....	25
Bibliografiya.....	68

Ön söz

ISO (Beynəlxalq Standartlaşdırma Təşkilatı) milli standartlar orqanlarının (ISO üzv qurumlarının) ümumdünya federasiyasıdır. Beynəlxalq Standartların hazırlanması işi adətən ISO-nun texniki komitələri vasitəsilə həyata keçirilir. Texniki komitənin yaradıldığı mövzu ilə maraqlanan hər bir üzv qurum həmin komitədə təmsil olunmaq hüququna malikdir. ISO ilə əlaqəli beynəlxalq təşkilatlar, dövlət və qeyri-hökumət təşkilatları da işdə iştirak edirlər. ISO bütün elektrotexniki standartlaşdırma məsələlərində Beynəlxalq Elektrotexniki Komissiya (IEC) ilə sıx əməkdaşlıq edir.

Beynəlxalq standartlar ISO/IEC Direktivlərinin 2-ci hissəsində verilmiş qaydalara uyğun olaraq hazırlanır.

Texniki komitələrin əsas vəzifəsi Beynəlxalq Standartları hazırlamaqdır. Texniki komitələr tərəfindən qəbul edilmiş Beynəlxalq Standartların layihələri səsvermə üçün üzv orqanlara göndərilir. Beynəlxalq Standart kimi dərc edilməsi üçün səs verən üzv qurumların ən azı 75%-nin təsdiqi tələb olunur.

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.

ISO 19118 Texniki Komitə ISO/TC 211 tərəfindən hazırlanmışdır, *Coğrafi məlumat/Geomatika*.

Bu ikinci nəşr texniki cəhətdən yenidən işlənmiş birinci nəşri (ISO 19118:2005) ləğv edir və əvəz edir.

Giriş

Bu Beynəlxalq Standart "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar dəsti daxilində coğrafi məlumatların mübadiləsi üçün istifadə olunan kodlaşdırma qaydalarının müəyyən edilməsi üçün tələbləri müəyyən edir. Kodlaşdırma qaydası tətbiq sxemləri və standartlaşdırılmış sxemlər ilə müəyyən edilmiş coğrafi məlumatı nəqliyyat və saxlama üçün uyğun olan sistemdən asılı olmayan məlumat strukturuna kodlamağa imkan verir. Kodlaşdırma qaydası kodlaşdırılan məlumatların növlərini və nəticədə əldə edilən məlumat strukturunda istifadə olunan sintaksisi, strukturu və kodlaşdırma sxemlərini müəyyən edir. Yaranan məlumat strukturu rəqəmsal mediada saxlanıla və ya ötürmə protokollarından istifadə etməklə ötürülə bilər. Verilənlərin kompüterlər tərəfindən oxunması və şərh edilməsi nəzərdə tutulur, lakin məlumatlar insan tərəfindən oxuna bilən formada ola bilər.

Tətbiqdən müstəqil məlumat mübadiləsi üçün bir kodlaşdırma qaydasının seçilməsi proqram domenlərini və ayrı-ayrı millətləri məlumatların ölçüsü və ya emal mürəkkəbliyi ilə bağlı platformadan asılı və ya daha effektiv ola biləcək öz kodlaşdırma qaydalarını müəyyən etmək və istifadə etmək imkanını istisna etmir. XML ISO/IEC 8879-un alt dəstidir və hesablama platformasından asılı olmadığı və Ümumdünya Şəbəkəsi ilə qarşılıqlı əlaqədə olduğu üçün seçilmişdir.

Bu Beynəlxalq Standart üç məntiqi hissəyə bölünür. UML sxemləri əsasında kodlaşdırma qaydalarının yaradılması üçün tələblər 6-9-cu bəndlərdə göstərilmişdir. Kodlaşdırma xidmətinin yaradılması üçün tələblər 10-cu bənddə, XML əsaslı kodlaşdırma qaydalarına dair tələblər isə Əlavə A-da göstərilmişdir.

XML əsaslı kodlaşdırma qaydası neytral məlumat mübadiləsi kimi istifadə üçün nəzərdə tutulub. Genişləndirilə bilən İşarələmə Dili (XML) və ISO/IEC 10646 simvol dəsti standartlarına əsaslanır.

Coğrafi məlumat standartları "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar dəsti daxilində təşkil edilmişdir. Bu Beynəlxalq Standartlar seriyasının əsası və ümumi strukturu və əsas təsvir üsulları ISO 19101, ISO/TS 19103 və ISO/TS 19104-də müəyyən edilmişdir.

Bu Beynəlxalq Standartın istifadəçiləri coğrafi məlumatı rəsmi şəkildə təsvir etmək üçün proqram sxemləri hazırlaya bilərlər. Tətbiq sxemi digər standartlaşdırılmış konseptual sxemlərdən (məsələn, ISO 19107) elementləri birləşdirməklə tərtib edilir. Bu inteqrasiyanın necə baş verməsi ISO 19109-da təsvir edilmişdir. "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar toplusu, həmçinin coğrafi məlumat proqramlarının işlənilib hazırlanması zamanı mövcud olan ümumi xidmətlər toplusunu müəyyən edir. Ümumi xidmətlər ümumiyyətlə ISO 19119-da müəyyən edilir və ümumi məlumat modelinə uyğun olaraq coğrafi məlumatlara giriş və emalı əhatə edir. Bu Beynəlxalq Standart icra məsələlərini əhatə edir.

Əlavə

Coğrafi məlumat — Kodlaşdırma

1 Əhatə dairəsi

Bu Beynəlxalq Standart "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar toplusunda coğrafi məlumatlara uyğun olan məlumatların mübadiləsi üçün istifadə üçün kodlaşdırma qaydalarının müəyyən edilməsi üçün tələbləri müəyyən edir.

Bu Beynəlxalq Standart müəyyən edir

- UML sxemləri əsasında kodlaşdırma qaydalarının yaradılması tələbləri,
- kodlaşdırma xidmətləri yaratmaq üçün tələblər və
- məlumatların neytral mübadiləsi üçün XML əsaslı kodlaşdırma qaydalarına tələblər.

Bu Beynəlxalq Standart heç bir rəqəmsal medianı müəyyən etmir, heç bir ötürmə xidmətlərini və ya ötürmə protokollarını müəyyən etmir, nə də daxili böyük şəkillərin kodlaşdırılmasını müəyyən etmir.

2 Uyğunluq

2.1 Giriş

Bu Beynəlxalq Standart üçün iki uyğunluq sinifləri dəsti müəyyən edilmişdir.

2.2 Kodlaşdırma qaydaları ilə bağlı uyğunluq sinifləri

Bütün kodlaşdırma qaydaları B.1-də mücərrəd test paketinin bütün sınaq işlərindən keçməlidir. Bütün kodlaşdırma qaydaları B.2 və/və ya B.3-də mücərrəd test paketinin bütün sınaq işlərindən keçməlidir.

Cədvəl 1 — Kodlaşdırma qaydaları ilə bağlı uyğunluq sinifləri

Uyğunluq sinfi	yarımbəndi abstrakt test dəsti
Bütün kodlaşdırma qaydaları	B.1
Nümunə çevrilməsi ilə kodlaşdırma qaydası	B.2
Sxemaya çevrilmə ilə kodlaşdırma qaydası	B.3

2.3 Kodlaşdırma xidmətləri ilə bağlı uyğunluq sinifləri

Bütün kodlaşdırma xidmətləri B.4-də mücərrəd test paketinin bütün sınaq işlərindən keçməlidir. Kodlaşdırma xidmətinin imkanlarından asılı olaraq, o, Cədvəl 2-yə uyğun olaraq əlavə uyğunluq siniflərinin bütün sınaq işlərindən keçməlidir.

Cədvəl 2 — Kodlaşdırma xidmətləri ilə bağlı uyğunluq sinifləri

Uyğunluq sinfi	yarımbəndi abstrakt test dəsti
Bütün kodlaşdırma xidmətləri	B.4
Ümumi kodlaşdırma xidməti	B.5
Məlumatları kodlayan xidmət	B.6
Məlumatları deşifrə edən xidmət	B.7
Çıxış verilənləri strukturu sxemini yaradan xidmət	B.8

3 Normativ istinadlar

Aşağıdakı istinad sənədləri bu sənədin tətbiqi üçün zəruridir. Tarixli istinadlar üçün yalnız istinad edilən nəşr tətbiq edilir. Tarixsiz istinadlar üçün istinad edilən sənədin ən son nəşri (hər hansı düzəlişlər daxil olmaqla) tətbiq edilir.

ISO 8601:2004, *Məlumat elementləri və mübadilə formatları — İnformasiya mübadiləsi — Tarixlərin və vaxtların təmsil olunması*

ISO/IEC 10646:2011, *İnformasiya texnologiyası — Universal Kodlu Simvol Dəsti (UCS)*

ISO/TS 19103:2005, *Coğrafi məlumat — Konseptual sxem dili*

ISO 19109:2005, *Coğrafi məlumat — Tətbiq sxemi üçün qaydalar*

Genişlənən İşarələmə Dili (XML) 1.0, W3C Tövsiyəsi. < ünvanında mövcuddur <http://www.w3.org/TR/REC-xml> >

4 Terminlər və təriflər

Bu sənədin məqsədləri üçün aşağıdakı terminlər və təriflər tətbiq edilir.

4.1 tətbiq sxemi

konseptual sxem(4.5) üçündata(4.8) bir və ya bir neçə proqram tərəfindən tələb olunur

[ISO 19101:2002, 4.2]

QEYD Tətbiq sxemi məzmunu, strukturu və tətbiq olunan məhdudiyətləri təsvir edirməlumat (4.22) xüsusi proqram domenində.

4.2 xarakter

təmsil etmək, təşkil etmək və ya nəzarət etmək üçün istifadə edilən elementlər toplusunun üzvüdata(4.8)

[ISO/IEC 2382-1:1993, 01.02.11]

4.3 kod

müəyyən edilmiş sxem üzrə etiketin təsviri

4.4**konseptual model**

model(4.27) anlayışlarını müəyyən edən **adanışıq aləmi**(4.33)

[ISO 19101:2002, 4.4]

4.5**konseptual sxem**

formal təsviri **akonseptual model**(4.4)

[ISO 19101:2002, 4.5]

4.6**konseptual sxem dili**

təmsil etmək məqsədi ilə konseptual formalizmə əsaslanan formal dil**konseptual sxemlər**(4.5)

[ISO 19101:2002, 4.6]

NÜMUNƏLƏR UML, EXPRESS, IDEF1X.

QEYD Konseptual sxem dili leksik və ya qrafik ola bilər.

4.7**çevrilmə qaydası**

girişdəki nümunələri çevirmək üçün qayda**data**(4.8) çıxış verilənləri strukturunda instansiyalara struktur

4.8**data**

yenidən şərh edilə bilən təmsil**məlumat**(4.22) ünsiyyət, şərh və ya emal üçün uyğun olan rəsmiləşdirilmiş şəkildə

[ISO/IEC 2382-1:1993, 01.01.02]

4.9**məlumat mübadiləsi**

çatdırılması, qəbulu və tərcüməsi**data**(4.8)

4.10**məlumat ötürülməsi**

hərəkət**data**(4.8) a üzərində bir nöqtədən digərinə**orta**(4.26)

QEYD Transfer**məlumat**(4.22) məlumatların ötürülməsini nəzərdə tutur.

4.11**məlumat növü**

spesifikasiyası **adəyər domeni**(4.34) bu domendeki dəyərlər üzərində icazə verilən əməliyyatlarla

[ISO/TS 19103:2005, 4.1.5]

NÜMUNƏLƏR Tam, Real, Boolean, Sətir və Tarix.

QEYD Məlumat növü terminlə müəyyən edilir, məsələn, Tam ədəd. Məlumat növlərinin dəyərləri müəyyən edilmiş dəyər sahəsinə aiddir, məsələn -65537 və 65536 arasında olan bütün tam ədədlər. Əməliyyatlar toplusu +, -, * və / ola bilər və semantik cəhətdən yaxşı müəyyən edilmişdir. Məlumat növü sadə və ya mürəkkəb ola bilər. Sadə məlumat növü müəyyən kontekstdə dəyərlərin atom hesab edildiyi dəyər sahəsinə müəyyən edir, məsələn, Tam ədəd. Müəkkəb məlumat növü bir araya toplanmış məlumat növləri toplusudur. Müəkkəb məlumat növü obyekt təmsil edə bilər və beləliklə, şəxsiyyətə malik ola bilər.

4.12

verilənlər toplusu

müəyyən edilə bilən kolleksiyad^{data}(4.8)

[ISO 19115:2003, 4.2]

4.13

kodlaşdırma

çevrilməsidad^{data}(4.8) bir sırakodlar(4.3)

4.14

kodlaşdırma qaydası

müəyyən edilə bilən kolleksiyaçevirmə qaydaları(4.7) müəyyən edirkodlaşdırma(4.13) xüsusi üçündata(4.8) quruluş

NÜMUNƏLƏR XML, ISO 10303-21, ISO/IEC 8211.

QEYD Kodlaşdırma qaydası çevrilən məlumatların növlərini, həmçinin sintaksisi, strukturunu vəkodlar(4.3)

əldə edilən məlumat strukturunda istifadə olunur.

4.15

kodlaşdırma xidməti

olan proqram komponentikodlaşdırma qaydası(4.14) həyata keçirilir

4.16

xüsusiyyət

real dünya hadisələrinin abstraksiyası

[ISO 19101:2002, 4.11]

QEYD Xüsusiyyət bir növ və ya nümunə kimi baş verə bilər. Xüsusiyyət növü və ya xüsusiyyət nümunəsi yalnız biri nəzərdə tutulduqda istifadə olunur.

4.17

fayl

vahid kimi saxlanılan və ya emal edilən qeydlər toplusu

[ISO/IEC 2382-1:1993, 01.08.06]

4.18

coğrafi məlumatlar

data(4.8) Yer kürəsinə nisbətən bir yerə açıq və ya gizli istinadla

[ISO 19109:2005, 4.12]

4.19

coğrafi məlumat

məlumat(4.22) Yərə nisbətən yerlə bağlı və ya açıq şəkildə əlaqəli olan hadisələrə dair

[ISO 19101:2002, 4.16]

4.20

identifikator

linqvistik cəhətdən müstəqil ardıcılığıpersonajlar(4.2) əlaqəli olanı unikal və daimi olaraq müəyyən etməyə qadirdir

[ISO 19135:2005, 4.1.5]

4.21**identifikasiya konvensiyası**

yaradılması qaydaları toplusu **identifikatorlar**(4.20)

4.22**məlumat**

obyektlər, məsələn, faktlar, hadisələr, əşyalar, proseslər və ya ideyalar, o cümlədən anlayışlar haqqında müəyyən kontekstdə xüsusi mənə daşıyan biliklər

[ISO/IEC 2382-1:1993, 01.01.01]

4.23**nümunə modeli**

təmsil **model**(4.27) saxlamaq üçün **data**(4.8) a uyğun olaraq **tətbiq sxemi**(4.1)

4.24**interfeys**

-UML- elementin davranışını xarakterizə edən əməliyyatlar toplusu

[ISO/IEC 19501]

4.25**qarşılıqlı fəaliyyət**

ünsiyyət qurmaq, proqramları icra etmək və ya ötürmək qabiliyyəti **data**(4.8) müxtəlif funksional vahidlər arasında istifadəçidən həmin bölmələrin unikal xüsusiyyətləri haqqında çox az və ya heç bir məlumata malik olmamasını tələb edən tərzdə

[ISO/IEC 2382-1:1993, 01.01.47]

4.26**orta**

saxlama və ya ötürmə üçün maddə və ya agentlik **data**(4.8)

NÜMUNƏLƏR Kompakt disk, internet^[1], radio dalğaları və s.

4.27**model**

reallığın bəzi aspektlərinin abstraksiyası

[ISO 19109:2005, 4.14]

4.28**sxem**

formal təsviri **amodel**(4.27)

[ISO 19101:2002, 4.25]

4.29**sxem modeli**

təmsil **model**(4.27) saxlamaq üçün **sxemlər**(4.28)

NÜMUNƏ Sxem anbarı üçün təmsil modeli.

4.30**stereotip**

-UML- metamodelin semantikasını genişləndirən modelləşdirmə elementinin yeni növü

[ISO/IEC 19501]

ISO 19118:2011(E)

QEYD Stereotiplərin metamodeldə mövcud olan müəyyən növlərə və ya siniflərə əsaslanması zəruridir. Stereotiplər əvvəlcədən mövcud olan növ və siniflərin strukturunu deyil, semantikasını genişləndirə bilər. Bəzi stereotiplər UML-də əvvəlcədən müəyyən edilmişdir, digərləri istifadəçi tərəfindən müəyyən edilə bilər. Stereotiplər UML-də üç genişlənmə mexanizmindən biridir; digərləri məhdudiyət və etiketli dəyərdir.

4.31

köçürmə protokolu

paylanmış sistemlər arasında qarşılıqlı əlaqənin müəyyən edilməsi üçün ümumi qaydalar toplusu

4.32

köçürmə vahidi

kolleksiyası **data**(4.8) a məqsədi **iləməlumat ötürülməsi**(4.10)

QEYD Köçürmə vahidinin a kimi müəyyən edilə bilən olması lazım deyil **verilənlər toplusu**(4.12).

4.33

danişiq aləmi

maraq doğuran hər şeyi ehtiva edən real və ya hipotetik dünyaya baxış

[ISO 19101:2002, 4.29]

4.34

dəyər domeni

qəbul edilmiş dəyərlər toplusu

[ISO/TS 19103:2005, 4.1.15]

NÜMUNƏ 3-28 diapazonu, bütün tam ədədlər, istənilən simvol, bütün qəbul edilmiş dəyərlərin sadalanması (yaşıl, mavi, ağ).

5 Simvollar və qısaldılmış terminlər

DCE	Paylanmış hesablama mühiti
DUID	Domenin unikal identifikatoru
HTML	Hipermetn işarələmə dili
MODIS	Orta dəqiqlikli görüntüləmə spektrometri
POSC	Neft Açıq Standartlar Konsorsiumu
TIFF	Tagged şəkil fayl formatı
UCS	Universal çoxoktetli kodlu simvol dəsti
UML	Vahid modeləşdirmə dili
UTF	UCS Transfer formatı
UUID	Universal unikal identifikator
XML	Genişləndirilə bilən işarələmə dili

6 Əsas anlayışlar və fərziyyələr

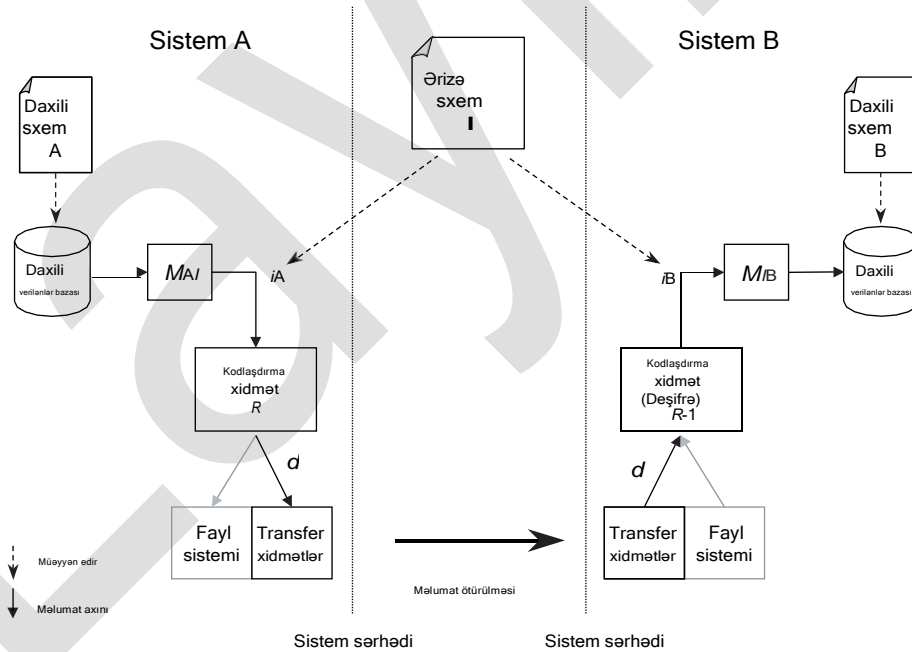
6.1 Konsepsiyalar

“ISO 19100 seriyası” kimi tanınan Beynəlxalq Standartlar toplusunun məqsədi heterojen coğrafi informasiya sistemləri arasında qarşılıqlı əlaqəni təmin etməkdir. Heterojen sistemlər arasında qarşılıqlı fəaliyyətə nail olmaq üçün iki əsas məsələni müəyyən etmək lazımdır. Birinci məsələ coğrafi məlumatların məzmununun semantikasını və məntiqi strukturlarını müəyyən etməkdir. Bu, tətbiq sxeminə edilməlidir. İkinci məsələ proqram sxeminə uyğun olan məlumatları təmsil edə bilən sistemdən və platformadan asılı olmayan məlumat strukturunun müəyyən edilməsidir.

Məlumat mübadiləsinin fundamental anlayışları, yəni coğrafi məlumatların kodlaşdırılması, çatdırılması, qəbulu və şərh üçün tətbiq sxeminə əsaslanan prosedur 6.2-6.6-da təsvir edilmişdir. Məlumat mübadiləsi prosesinin icmalı 6.2-də təsvir edilmişdir; 6.3 coğrafi məlumatların şərhinə imkan verən tətbiq sxemlərini təqdim edir; 6.4 sistemdən müstəqil məlumat strukturlarının yaradılması üçün kodlaşdırma qaydasının əhəmiyyətini təsvir edir; 6.5 kodlaşdırma qaydasını yerinə yetirmək üçün kodlaşdırma xidməti adlanan proqram komponentini təsvir edir; və 6.6 köçürmə xidməti adlanan çatdırılma və qəbul prosedurunu təsvir edir.

6.2 Məlumat mübadiləsi

Məlumat mübadiləsinin icmalı Şəkil 1-də göstərilmişdir. Sistem A verilənlər toplusunu B sistemində göndərmək istəyir. Uğurlu mübadilə təmin etmək üçün A və B üç şey üzərində qərar verməlidir: yeni ümumi tətbiq sxemil, hansı kodlaşdırma qaydası R müəyyənləşdirmək və hansı ötürmə protokolundan istifadə etmək. Tətbiq sxemi uğurlu məlumat ötürülməsinin əsasını təşkil edir və ötürülən məlumatların mümkün məzmununu və strukturunu müəyyən edir, kodlaşdırma qaydası isə verilənlərin sistemdən asılı olmayan məlumat strukturuna necə kodlaşdırılmasına dair çevrilmə qaydalarını müəyyən edir.



Şəkil 1 — İki sistem arasında məlumat mübadiləsinə baxış

Hər iki sistem, A və B məlumatları daxili verilənlər bazasında daxili sxemə uyğun saxlayır, lakin sxemlər adətən fərqli olur, yəni A sxemi B sxeminə bərabər deyil. A-nın daxili verilənlər bazasından B-nin daxili verilənlər bazasına verilənlər toplusu.

- a) A sistemi üçün ilk addım onun daxili məlumatlarını ümumi tətbiq sxeminə uyğun olan məlumat strukturuna çevirməkdir. Burada bu, daxili sxem anlayışlarından tətbiq sxeminə müəyyən edilmiş anlayışlara xəritəçəkmənin müəyyən edilməsi və məlumat nümunələrinin tərcüməsi üçün müvafiq xəritəçəkmə proqramının yazılması ilə həyata keçirilir. Şəkil 1-də bu xəritələşdirmə kimi işarələnmişdir *MA*. Nəticə tətbiq sxeminə xas məlumat strukturudur, *IA*. Məlumat strukturu yaddaşda və ya ara faylda saxlanılır və sistemdən asılıdır və buna görə də ötürülmək üçün uyğun deyil.
- b) Növbəti addım kodlaşdırma qaydasını tətbiq edən kodlaşdırma xidmətindən istifadə etməkdir *R* sistemdən asılı olmayan və buna görə də ötürülməyə uyğun olan məlumat strukturu yaratmaq. Bu kodlanmış verilənlər toplusu adlanır *dvə* fayl sisteminə saxlanıla və ya köçürmə xidmətindən istifadə etməklə ötürülə bilər.
- c) Sistem A sonra kodlaşdırılmış məlumat dəstinə göndərmək üçün ötürmə xidmətini işə salır *dB* sistemində. Köçürmə xidməti qablaşdırmanın necə aparılması və faktiki daşınmanın on-line və ya off-line rabitə vasitəsi ilə necə baş verməsi ilə bağlı ötürmə protokoluna əməl edir. Hər iki tərəfin istifadə edilən köçürmə protokolu ilə bağlı razılığa gəlməsi lazımdır.
- d) B sistemindəki ötürmə xidməti ötürülən verilənlər toplusunu qəbul edir və protokola uyğun olaraq verilənlər toplusu paketdən çıxarılır və kodlaşdırılmış verilənlər toplusu kimi saxlanılır. *dməsələn*, ara faylda.
- e) Tətbiq sxeminə xas məlumat strukturunu əldə etmək üçün *B*, B sistemi tərs kodlaşdırma qaydasını tətbiq edir *R-1* kodlanmış məlumatları şərh etmək.
- f) Verilənlər dəstindən istifadə etmək üçün B proqramın sxemə aid olan məlumat strukturunu tərcümə etməlidir. *B* daxili məlumat bazasına daxil edilir. Bu, tətbiq sxemindən onun daxili sxeminə xəritəçəkmənin müəyyən edilməsi və faktiki tərcüməni həyata keçirən proqram təminatının yazılması ilə həyata keçirilir. Şəkil 1-də bu xəritələşdirmə qeyd edilmişdir *M/B*.

Bu Beynəlxalq Standart bütün məlumat mübadiləsi prosesini deyil, yalnız kodlaşdırma qaydaları və kodlaşdırma xidmətləri yaratmaq üçün tələbləri müəyyən edir. Beləliklə, yalnız b) və e) addımları standartlaşdırılır. a), c), d) və f) addımları ümumi informasiya texnologiyası xidmətlərindən istifadə edir.

6.3 Tətbiq sxemi

Tətbiq sxemi oxşar məlumat tələbləri olan tətbiqlər üçün konseptual sxemdir. Tətbiq sxemi uğurlu məlumat mübadiləsinin əsasını təşkil edir və məlumatların mümkün məzmununu və strukturunu müəyyən edir. O, həmçinin məlumatların lokal saxlanması üçün proqram-sxemaya xas məlumat strukturlarının həyata keçirilməsi üçün əsasdır.

Bu Beynəlxalq Standarta uyğun olaraq kodlaşdırma üçün istifadə edilən proqram sxemi ISO/TS 19103 və ISO 19109-a uyğun olaraq UML konseptual sxem dilində yazılmalıdır. Bu Beynəlxalq Standartlar tətbiq sxemlərinin necə yazılması üçün çərçivəni müəyyən edir. Qaydalara xüsusiyyət növlərini müəyyən etmək üçün standartlaşdırılmış sxemlərdən necə istifadə ediləcəyinə dair spesifikasiyalar daxildir. Həm məlumat göndərən, həm də qəbul edən proqram sxeminə çıxışı olmalıdır.

Tətbiq sxemi uğurlu nəticəni təmin etmək üçün məlumat mübadiləsinin hər iki ucuna əlçatan olmalıdır. Məlumat mübadiləsi baş verməzdən əvvəl tətbiq sxeminin ötürülməsi lazımdır ki, həm alıcı, həm də göndərən tətbiq sxeminə uyğun olaraq xəritələr və məlumat strukturlarını həyata keçirərək sistemlərini hazırlaya bilsinlər. O, verilənlər bazası ilə birlikdə ötürülə bilər və ya ictimai yerdə saxlanıla və verilənlər bazasından istinad edilə bilər.

Tətbiq sxemi kağız və ya elektron əsaslı üsullarla dəyişdirilə bilər.

6.4 Kodlaşdırma qaydası

6.4.1 Konsepsiya

Kodlaşdırma qaydası, müəyyən bir məlumat strukturu üçün kodlaşdırmanı təyin edən, müəyyən edilə bilən çevrilmə qaydalarının toplusudur. Kodlaşdırma qaydası çevrilən məlumat növlərini, həmçinin nəticədə əldə edilən məlumat strukturunda istifadə olunan sintaksisi, strukturu və kodlaşdırma sxemlərini müəyyən edir. Nəqliyyat və ya saxlama üçün uyğun olan sistemdən asılı olmayan məlumat strukturları yaratmaq üçün proqram-sxemativ məlumat strukturlarına kodlaşdırma qaydası tətbiq edilir. Kodlaşdırma qaydasını müəyyən etmək üçün üç mühüm aspektin müəyyən edilməsi lazımdır: giriş məlumat strukturu, çıxış məlumat strukturu və giriş və çıxış məlumat strukturlarının elementləri arasında çevrilmə qaydaları. Həm giriş, həm də çıxış məlumat strukturları konseptual sxem dili ilə yazılır və dillərdəki anlayışlar kodlaşdırma qaydasını müəyyən etmək üçün istifadə olunur.

6.4.2 Daxil olan verilənlərin strukturu

Daxil olan verilənlərin strukturu proqrama aid sxemə xas məlumat strukturudur. Məlumat strukturu məlumat nümunələri toplusu kimi düşünülə bilər, yəni $\{I_1, \dots, I_n\}$; Şəkil 1-ə baxın. Hər bir məlumat nümunəsi, I_k , konsepsiyanın bir nümunəsidir, I , tətbiq sxemində müəyyən edilmişdir. Tətbiq sxemi tətbiq sxemində müəyyən edilmiş anlayışlar toplusunu müəyyən edir $\{I_1, \dots, I_n\}$.

Tətbiq sxemi konseptual bir sxemdir, C konseptual sxem dilində yazılmış, C . Konseptual sxem anlayışlar toplusunu müəyyən edir $\{C_1, \dots, C_m\}$ konseptual sxem dilinin anlayışlarını təcəssüm etdirərək $C - \{C_1, \dots, C_m\}$. Tətbiq sxemi konseptual sxem olduğundan, $C-I$.

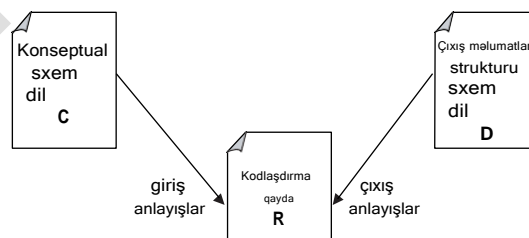
6.4.3 Çıxış məlumatlarının strukturu

Çıxış məlumatlarının strukturu bir sxem ilə müəyyən edilir, $D - \{D_1, \dots, D_n\}$. Çıxış strukturu üçün sxemdir və Şəkil 1-də göstərilir. Çıxış verilənləri strukturu məlumat nümunələri toplusu kimi düşünülə bilər, yəni $\{d_1, \dots, d_n\}$ burada hər bir məlumat nümunəsi, d_k , konsepsiyanın bir nümunəsidir, D .

Sxem, D , çıxış verilənlər strukturunun sintaksisini, strukturunu və kodlaşdırma sxemlərini müəyyən edir.

6.4.4 Konversiya qaydaları

Dönüşüm qaydası, giriş məlumatı strukturunda məlumat nümunəsinin çıxış məlumat strukturunda sifra, bir və ya daha çox nümunəyə necə çevriləcəyini müəyyən edir. Dönüşüm qaydaları konseptual sxem dilinin anlayışlarına əsasən müəyyən edilir və C , və çıxış məlumat strukturu sxeminin anlayışları haqqında, D . Dönüşüm qaydasını müəyyən etmək lazımdır, R , konseptual sxem dilində anlayışların hüquqi birləşmələrinin hər biri üçün. Dönüşüm qaydaları toplusudur $R - \{R_1, \dots, R_n\}$, harada R_i i -ci çevrilmə qaydası və C_i i -şema dilindən nümunələrin hüquqi birləşməsi. Mümkün olan hər kəs üçün dönüşüm cədvəli C hər biri harada qurula bilər C_i çıxış məlumat strukturunda nümunələrin istehsalına xəritələr, D . Şəkil 2 giriş və çıxış konseptual sxem dili ilə kodlaşdırma qaydası arasındakı əlaqəni göstərir.



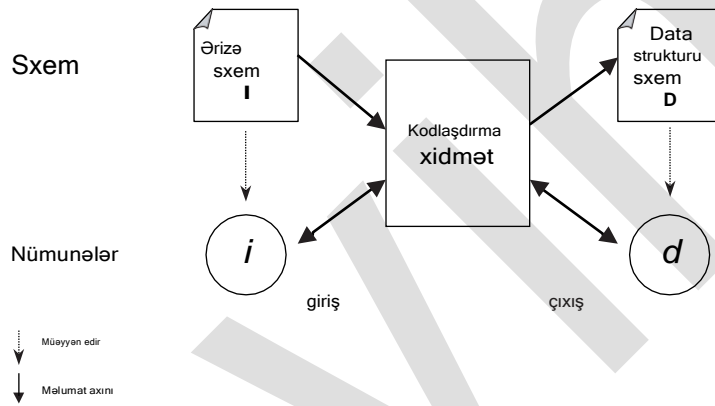
Şəkil 2 — Kodlaşdırma qaydası giriş anlayışlarından çıxış anlayışlarına çevrilmə qaydalarını müəyyən edir

QEYD Dönüşüm qaydaları hər hansı xüsusi tətbiqə görə deyil, iki sxem dili əsasında müəyyən edilir sxem. Bu, tərtibatçılara proqram-sxemdən asılı olmayan kodlaşdırma xidmətləri yazmağa imkan verən ümumi yanaşmadır, sxemlər eyni konseptual sxem dilində müəyyən edildiyi müddətdə müxtəlif proqram sxemləri üçün istifadə edilə bilər.

6.5 Kodlaşdırma xidməti

Kodlaşdırma xidməti kodlaşdırma qaydasını həyata keçirən və kodlaşdırma və deşifrə funksiyası üçün interfeys təmin edən proqram komponentidir. Bu, məlumat mübadiləsinin inteqrasiya olunmuş hissəsidir.

Şəkil 3 kodlaşdırma xidmətinin təfərrüatlarını və onun mühüm spesifikasiya sxemləri ilə əlaqələrini təqdim edir. Kodlaşdırma xidməti giriş verilənləri strukturunu oxuya və nümunələri çıxış məlumat strukturuna çevirə bilməlidir və əksinə. O, həmçinin tətbiq sxemi bəyannamələrini oxuya və müvafiq çıxış məlumat strukturunu sxemini yazıya bilməlidir. Giriş məlumatlarının strukturu tətbiq sxemi ilə müəyyən edilir. Tətbiq sxemi konseptual sxem dilinin anlayışlarından istifadə etməklə müəyyən edilir. Çıxış verilənləri strukturunu, həmçinin çıxış verilənlər strukturunun mümkün məzmununu, strukturunu və kodlaşdırma sxemlərini təyin edən verilənlər strukturunu sxemi adlanan sxemlə təsvir edilir. Məlumat strukturunu sxemi sxem dili ilə təsvir edilmişdir. Kodlaşdırma qaydası iki səviyyədə çevirmə qaydalarını müəyyən edir: birincisi sxem səviyyəsində, ikincisi isə instansiya səviyyəsindədir. Sxem səviyyəsində, çevirmə qaydaları tətbiq sxemində müəyyən edilmiş anlayışların hər biri üçün verilənlər strukturunu sxemindəki uyğun anlayışlara uyğunlaşmanı müəyyən edir. Nümunə səviyyəsində, çevirmə qaydaları giriş məlumatı strukturundakı nümunələrin hər biri üçün çıxış məlumatı strukturunda müvafiq nümunələr üçün xəritələşdirməni müəyyən edir. Nümunələrin çevrilməsi qaydaları adətən şemanın çevrilməsi qaydalarından çıxarılır.



Şəkil 3 — Kodlaşdırma prosesinin icmalı

Kodlaşdırma xidməti ən azı kodlaşdırma və dekodlaşdırma funksionallığı üçün interfeyslər təmin etməlidir. Bu cür interfeyslərin nümunələri kodlaşdırma üçün (i, I) kodlaşdırmaq və deşifrə üçün (d, I) deşifrə. Burada, i proqram-sxemaya xüsusi verilənlər strukturuna istinaddır; I tətbiq sxemində istinaddır; d sistemdən müstəqil məlumat strukturuna istinaddır.

6.6 Transfer xidməti

Köçürmə xidməti, paylanmış informasiya sistemləri arasında oflayn və ya onlayn rabitə vasitələri ilə məlumatların ötürülməsinə imkan verən bir və ya bir neçə ötürmə protokolunu həyata keçirən proqram komponentidir. İki sistem arasında məlumatı uğurla ötürmək üçün göndərən və qəbul edən istifadə olunan ötürmə protokolu barədə razılığa gəlməsi lazımdır.

Müxtəlif ötürmə protokolları müəyyən edilə bilər. Bir nümunə, məlumatların optik və ya maqnit daşıyıcılarında saxlandığı və poçt xidmətləri və ya digər xüsusi çatdırılma xidmətlərindən istifadə edilərək çatdırıldığı oflayn ötürmə protokollarıdır. Başqa bir nümunə, verilənlərin sıxıldığı və e-poçt əlavəsi kimi daxil edildiyi, fayl ötürmə protokolundan istifadə edilərək çatdırıldığı və ya əsas şəbəkə xidmətinə əsaslanan digər paylanmış informasiya texnologiyaları xidmətlərindən istifadə edilərək ötürüldüyü onlayn ötürmə protokollarıdır.

Bu Beynəlxalq Standart heç bir üstünlük verilən ötürmə protokollarını təyin etmir.

7 personajların repertuarı

ISO/IEC 10646 Universal Character Set (UCS) adlı simvolların beynəlxalq səviyyədə tanınmış repertuarını və onun simvol kodlaşdırma sxemlərini müəyyən edir. Bu Beynəlxalq Standartın həyata keçirilməsində ISO/IEC 10646-da müəyyən edilmiş beynəlxalq xarakter dəsti standartlarından istifadə edilməlidir.

Bu Beynəlxalq Standartın beynəlxalq profilləri tərəfindən dəstəklənə bilən simvol kodlaşdırma sxemləri aşağıdakılardır:

- a) 8 bit dəyişən ölçülü UCS Transfer Format UTF-8;
- b) 16-bit dəyişən ölçüsü UCS Transfer Format UTF-16;
- c) 16-bit sabit ölçülü Universal Xarakterlər Dəsti UCS-2 (köhnəlmişdir);
- d) 32-bit sabit ölçülü Universal xarakter dəsti UCS-4.

Bu Beynəlxalq Standarta uyğunluğu iddia edən beynəlxalq kodlaşdırma qaydaları bu simvol kodlaşdırma sxemlərindən birini və ya bir neçəsini dəstəkləməlidir. Milli profillər və sistem tətbiqləri çərçivəsində müxtəlif simvol kodlaşdırma sxemlərindən istifadə edilə bilər. Sabit ölçülü simvol kodlaşdırma sxemləri tez-tez verilənlər bazası tətbiqlərində istifadə olunur və dəyişən ölçülü tez-tez məlumat mübadiləsi məqsədləri üçün istifadə olunur.

ISO/IEC 10646 yalnız personajların repertuarını təyin edir və əslində hansı dilin istifadə edildiyini göstərmir.

QEYD 1 Mətn sətirlərində müxtəlif dilləri ayırmağın vacib olduğu hallarda, xüsusi mexanizmlər istifadə oluna biləcəyini göstərin.

ISO/IEC 10646 kompozit simvolların yaradılması mexanizmlərini müəyyən edir. Mürəkkəb simvollar bir və ya bir neçə əlavə simvolun əsas simvolun üzərinə qoyulması nəticəsində yaranan simvollarıdır. ISO/IEC 10646 əvvəlcədən tərtib edilmiş simvollar toplusunu və onların müəyyən edilmiş parçalanmasını müəyyən edir. Mürəkkəb simvolların əvvəlcədən tərtib edilmiş ekvivalentləri ilə qarışdırılması şərh problemlərinə səbəb ola biləcəyi üçün, əvvəlcədən tərtib edilmiş simvol varsa, mürəkkəb simvoldan istifadə köhnəlmişdir, yəni əvvəlcədən tərtib edilmiş simvol həmişə istifadə edilməlidir.

Xülasə etmək üçün kodlaşdırma qaydası olmalıdır

- bir və ya bir neçə simvol kodlaşdırma sxemini dəstəkləyin və
- ekvivalent əvvəlcədən tərtib edilmiş simvollar varsa, kompozit simvoldan istifadə etməyin.

NÜMUNƏ Əvvəlcədən tərtib edilmiş δ simvolu müəyyən edilmiş σ parçalanmaya malikdir

QEYD 2 Xarakterin normallaşdırılmasının daha ətraflı təsviri üçün baxın <http://www.unicode.org/reports/tr15/> və <http://www.w3.org/TR/charmod-norm/>.

QEYD 3 UTF-16, UCS-2 və UCS-4 tələb edir məlumat haqqında necə üçün bəydləşmə ilə bayt sifariş vermək, bax http://www.unicode.org/faq/utf_bom.html.

8 Ümumi nümunə modeli

8.1 Giriş

Ümumi nümunə modeli 8-ci bənddə müəyyən edilmişdir. Nümunə modeli kodlaşdırma xidmətlərini inkişaf etdirərkən məlumatların rahat ümumi təsviridir. Nümunə modeli UML-də ifadə olunan tətbiq sxemləri ilə təsvir edilən məlumatları təmsil etməyə qadirdir. Nümunə modeli 6-cı bənddə (məlumat strukturları) müəyyən edilmiş tətbiq sxemində xas məlumat strukturunu təmsil edir (A və B Şəkil 1-də). Nümunə modeli obyektlərin ardıcılığını (IM_Object) ehtiva edən verilənlər toplusundan (IM_Dataset) ibarətdir, burada obyekt xassələr ardıcılığından (IM_Property) ibarətdir. Bu kontekstdəki xüsusiyyətlər ya atributlar, ya da assosiasiyalardır; əməliyyatlar deyil.

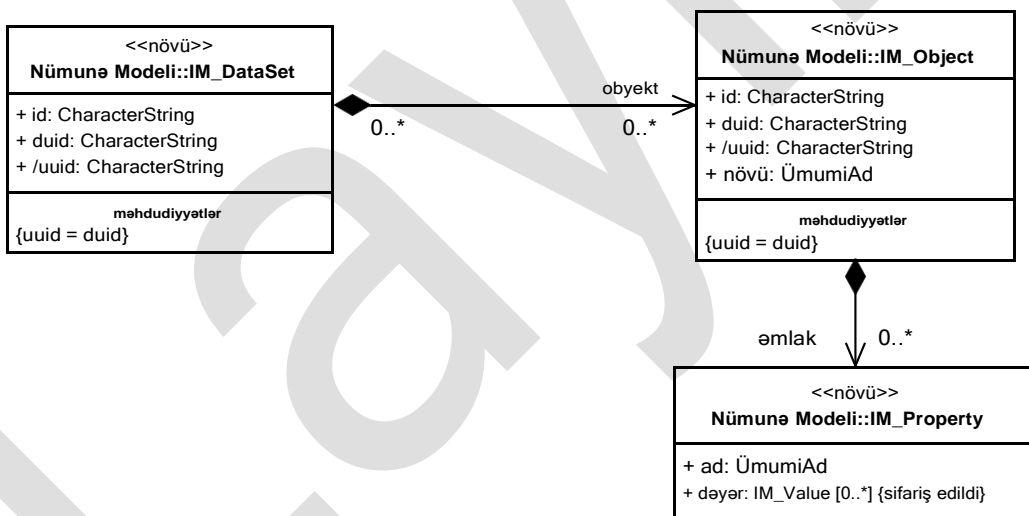
ümumi nümunə modelinə daxildir. Hər bir əmlak məlumat növünə görə kodlanır. Nümunə modeli Şəkil 4 və 5-də göstərilmişdir.

Tətbiq sxemi bir sıra sinifləri və onların atributlarını və assosiasiyalarını müəyyən edir və məlumat təqdimatlarının yaradılması üçün əsasdır. Məlumat təqdimatı (verilənlər toplusu) sinif təriflərinə uyğun olaraq strukturlaşdırılmış və kodlaşdırılmış bir və ya bir neçə obyektəndən ibarətdir. 8-ci bənddə obyektlərin, onların atributlarının və obyektlər arasında assosiasiyaların necə təmsil olunması prinsipləri təsvir olunur.

Verilənlər toplusunda əsas məlumat vahidi obyektədir. Obyekt vahid konkret sinfin nümunəsi olmalıdır. Mücərrəd siniflərin və interfeys kimi stereotipləşdirilmiş siniflərin heç bir nümunəsi yoxdur. Beləliklə, bu cür siniflər tərəfindən müəyyən edilmiş xüsusiyyətlər onları miras alan və ya reallaşdıran konkret siniflərin bir hissəsi kimi kodlanır. Tətbiq sxemində hər bir sinfin unikal adı olmalıdır. Tətbiq sxemi standartlaşdırılmış sxemlərdə və ya digər tətbiq sxemlərində müəyyən edilmiş siniflərə istinad edə və ya istifadə edə bilər. Bu siniflərin bəyannaməsi ya proqram sxemini ehtiva edən UML modelinə daxil edilməlidir, ya da proqram sxemini ayrıca fayl kimi müşayiət etməlidir.

Obyekt əmlak qiymətləri toplusunu ehtiva etməlidir. Obyektin sinfi xassələri müəyyən edir və onlar ya "sinif" supertipləri vasitəsilə miras alınır, ya da sinfin özündə müəyyən edilə bilər. Fərqli xassələri bir-birindən fərqləndirmək üçün hər bir əmlakın öz sinfində unikal adı olmalıdır. Mülkün məlumat tipi mümkün dəyərləri idarə edir və çoxluq ifadəsi yaradılmış obyektə atributun nümunələrinin sayını göstərir.

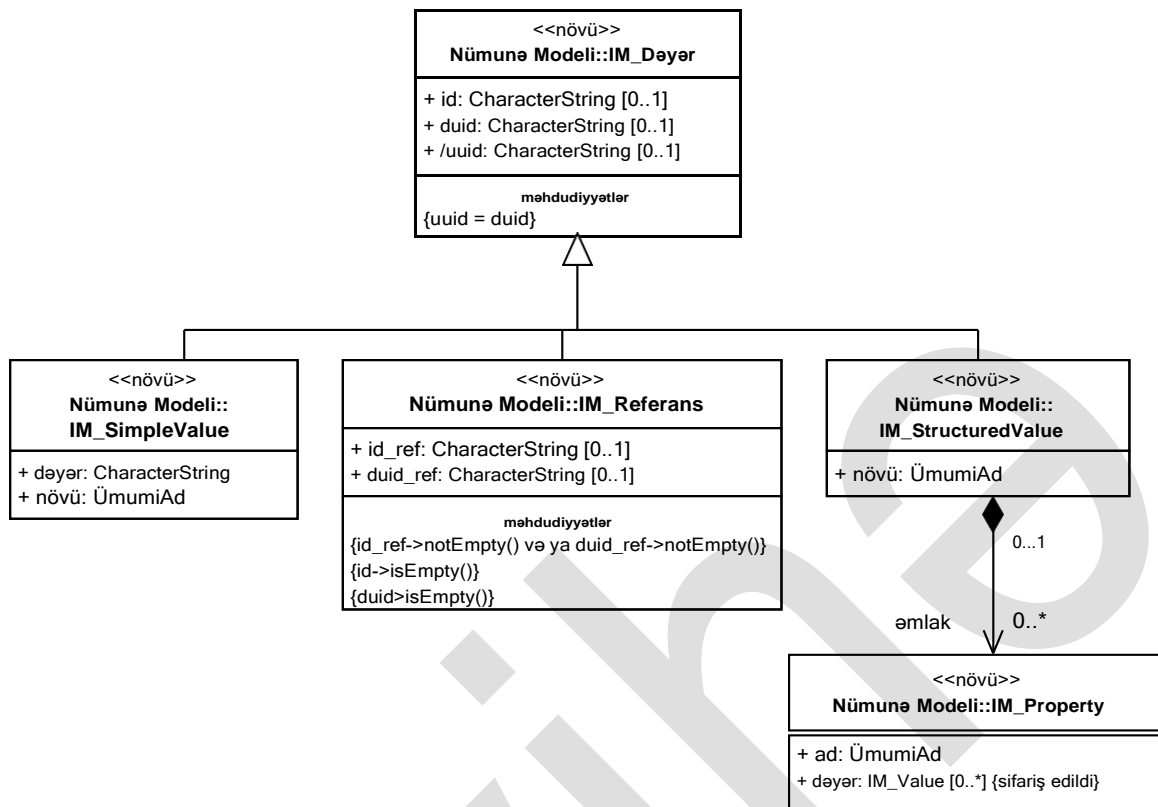
Obyektin, obyektin vəziyyətini təmsil etmək üçün zəruri olan mümkün atributları və assosiasiyaları müəyyən edən tətbiq sxemində və ya standartlaşdırılmış sxemdə müəyyən edilmiş müvafiq sinif var. IM_Object "sinif" atributu ilə öz sinfinə istinad edir, o, verilənlər bazası kontekstində özünəməxsus identifikatoru "id" ilə müəyyən edilməlidir və müəyyən edilmiş kainat, proqram domeni və ya ad məkanı daxilində universal olaraq unikal şəkildə müəyyən edilə bilər. "duid" atributu.



Şəkil 4 — Nümunə modeli—Dataset, obyekt və mülkiyyət

Sinif tərəfindən təyin olunan atributlar və sinifdən naviqasiya edilə bilən assosiasiya ucları xassələr dəstinə uyğunlaşdırılır. Əmlak (IM_Property) sifarişli dəyərlər toplusu olan adı təmsil edir. O, bir atribut və ya assosiasiya sonunu təmsil edə bilər. Əmlak adı assosiasiyanın atribut adına və ya hədəf rolunun adına uyğun olmalıdır. Dəyər (IM_Value) mülkiyyət dəyərini təmsil edir.

Null dəyərlər açıq və ya gizli şəkildə verilə bilər. Açıq null dəyəri verilmiş nilReason dəyəri ilə müvafiq IM_Property nümunəsi ilə göstərilməlidir. Müvafiq IM_Property nümunəsi yoxdursa, gizli null dəyəri göstərilir.



Şəkil 5 — Nümunə modeli—Dəyər növləri

Üç dəyər növü aşağıdakı kimi müəyyən edilir.

- IM_SimpleValue sadə məzmunun dəyərini təmsil edir.

NÜMUNƏ Tam ədəd və ya simvol sətri.

- IM_Reference hədəf obyektinə keçid və ya istinadı təmsil edir. Hədəf obyektı eyni ötürmə vahidində və ya başqa birində yerləşə bilər. Unikal identifikator (id_ref) eyni ötürmə vahidində yerləşən obyektı hədəfləyir. Domenin unikal identifikatoru (duid_ref) proqram domeninin kontekstində yerləşən obyektı hədəfləyir.
- IM_StructuredValue mürəkkəb məzmunlu məlumat növü dəyərini təmsil edir [xassələr ardıcılığı (IM_Property)].

Obyekt öz assosiasiyaları vasitəsilə bir və ya bir neçə obyektə əlaqələndirilə bilər (və ya istinad edə bilər). UML müxtəlif sémantika ilə assosiasiya, aqreqasiya və kompozisiya adlanan üç müxtəlif növ assosiasiyaları müəyyən edir; ətraflı məlumat üçün ISO/TS 19103-ə baxın. Ümumilikdə, assosiasiyaları təmsil etmək üçün iki strategiya var: ya obyektlərin bir hissəsi kimi əhatə olunmuş, ya da ayrı-ayrı assosiasiya obyektləri və ya əlaqə cədvəlləri kimi obyektlərdən ayrılmış.

İnkapsullaşdırılmış təmsil strategiyası assosiasiyaları mənəbe obyektı xassəsinə və hədəf obyekt xassəsinə bölür. Bu iki əlaqə xüsusiyyəti sonra hər biri digər obyektə işarə edir. Bağlantı xassəsi hədəf obyektlərinə və ya kompozisiya vəziyyətində hədəf obyektlərin özünə istinadları ehtiva edir. Bağlantı xassəsi hədəf sinfinə yaxın rol adı ilə müəyyən edilir və müvafiq çoxluğa malikdir. Əgər rolun adı yoxdursa və ya assosiasiya mənəbe obyektindən naviqasiya edilə bilməzsə, onda heç bir keçid xüsusiyyəti olmayacaq. Referensial tamliq məhdudiyətinin tətbiq olunduğu halda iki keçid mülkiyyətinin dəyəri uyğun olmalıdır. Yəni, mənəbe obyektı keçid xassəsi vasitəsilə başqa hədəf obyektinə istinad edirsə və hədəf obyektin mənəbe obyektı ilə iki istiqamətli assosiasiyası varsa, hədəf obyektin mənəbe obyektinə istinad edən müvafiq keçid xassəsinə malik olmalıdır. Əgər keçid xassəsi istinad və ya daxil edilmiş obyekt kimi kodlaşdırılıbsa və ya ümumiyyətlə kodlaşdırılmayıbsa, o, konkret kodlaşdırma qaydası ilə müəyyən edilir (məsələn, əlavə A).

8.2 UML və nümunə modeli arasında əlaqə

Cədvəl 3 UML və nümunə modeli arasındakı əlaqənin xülasəsini verir.

Supertipləri olan siniflərə əsaslanan obyektlər öz sinfinin və supertiplərinin bütün xassələrini və assosiasiya sonlarını ehtiva etməlidir. Beləliklə, bütün atributlar və assosiasiya ucları supertiplərdən kopyalanmalı və obyektin bir hissəsi hesab edilməlidir. Atribut və assosiasiyanın son adları atributların dəyərlərinə çıxış yolu olmalıdır və buna görə də onlar sinif daxilində unikal olmalıdır.

Əməliyyatlar və məhdudiyətlər nümunə modelinə uyğunlaşdırılmamalıdır.

Cədvəl 3 — UML ilə nümunə modeli arasındakı əlaqənin xülasəsi

UML konsepsiyası	Nümunə modeli
Paket	Yoxdur ^a
Sınıf Stereotip <<İnterfeys>> <<Məlumat Tipi>> <<Birlik>> <<Sadalama>> <<Kod Siyahısı>> <<Növ>> <<Feature Type>> YOX istifadə olunan UML profili ilə müəyyən edilmiş hər hansı digər stereotip	IM_Obyekt IM_SimpleValue və ya IM_StructuredValue IM_StructuredValue IM_SimpleValue IM_SimpleValue və ya IM_StructuredValue IM_Object IM_Obyekt IM_Obyekt kodlaşdırma qaydası spesifikasiyası ilə müəyyən edildiyi kimi
Atribut	Atribut növünə görə IM_Value ilə IM_Property; ya IM_SimpleValue, IM_Reference və ya IM_StructuredValue
Assosiasiya	IM_Reference ilə IM_Property
Toplama	IM_Reference ilə IM_Property
Tərkibi	Hədəf növünə görə IM_Value ilə IM_Property; ya IM_Reference (hədəf növü bir sinifdir) və ya IM_SimpleValue və ya IM_StructuredValue (hədəf növü məlumat növüdür)
Ümumiləşdirmə	Alt sinifə uyğun olan obyekt, alt sinfin super siniflərdən miras aldığı bütün xüsusiyyətləri daşıyır.
Əməliyyat	Yoxdur
Məhdudiyət	Yoxdur
^a	N/A tətbiq edilə bilməz deməkdir.

9 Kodlaşdırma qaydaları

9.1 Giriş

Kodlaşdırma qaydalarının müəyyənləşdirilməsi üçün tələblər 9.2-9.6-da müəyyən edilmişdir. Kodlaşdırma qaydası verilənləri giriş məlumatı strukturundan çıxış məlumat strukturuna çevirmək üçün çevirmə qaydalarını təsvir edir. Sxemlər həm giriş, həm də çıxış məlumat strukturu üçün təsvir edilməlidir. Daxil olan verilənlərin strukturu üçün sxem tətbiq sxemi adlanır.

Kodlaşdırma qaydası ümumiyyətlə aşağıdakıları müəyyən etməlidir:

a) ümumi kodlaşdırma tələbləri (9.2):

1) tətbiq sxemləri və sxem dili,

- 2) hər baytda bitlərin sırası və bir sözdə bayt (mümkün olduqda),
- 3) xarakter repertuarı və kodlaşdırılması,
- 4) zəruri mübadilə metadata,
- 5) verilənlər bazası və obyektin identifikasiyası konvensiyası;

b) giriş məlumat strukturu (9.3):

- 1) proqram sxemində uyğun olaraq məlumatları ötürmək üçün istifadə olunan məlumat strukturu (məlumat strukturları/Avə/B Şəkil 1-də) nümunə modeli adlanan kodlaşdırma xidmətinə,
- 2) nümunə modelinin tətbiq sxemi ilə necə əlaqəli olması;

c) mübadilə formatı adlanan çıxış məlumat strukturu (9.4);

d) nümunə modelindəki verilənləri mübadilə formatına çevirmək üçün xəritələşdirmə adlanan çevirmə qaydaları (9.5):

- 1) kodlaşdırma üçün çevirmə qaydaları,
- 2) zərurət yarandıqda, deşifrə üçün çevirmə qaydaları;

e) mücərrəd məlumatların kifayət qədər nümunələri, konversiya qaydalarının tətbiqi və kodlaşdırılmış məlumat (9.6).

9.2 Ümumi kodlaşdırma tələbləri

9.2.1 Tətbiq sxemi və sxem dili

Kodlaşdırma qaydası proqram sxemlərini müəyyən etmək üçün istifadə edilən sxem dilini təyin etməli və proqram sxeminin necə təşkil olunduğunu təsvir etməlidir.

9.2.2 Bit və bayt sıralaması

Əgər kodlaşdırma qaydası ikili kodlaşdırmanı müəyyən edirsə, o, hər bayt daxilində bitlərin sırasını və hər hansı çoxbaytlı strukturda (söz) sırasını və bayt sayını müəyyən etməlidir.

QEYD Bunlar mətn və ikili kodlaşdırma qaydalarına tətbiq olunan ümumi qaydalardır. Hətta mətn əsaslı kodlaşdırmalar (məsələn UTF-16, UCS-2, UCS-4) bayt sıralamasının spesifikasiyasını tələb edir (bax http://www.unicode.org/faq/utf_bom.html).

9.2.3 Xarakterlərin repertuarı və kodlaşdırılması

Xarakter repertuarı icazə verilən personajları müəyyənləşdirir. Xarakter repertuarı məlumatların dilini müəyyən etmir. Eyni xarakter repertuarı müxtəlif yollarla kodlaşdırıla bilər.

Birdən çox dil tələb olunarsa, proqram sxemi kodlaşdırılmış mətnin faktiki dilini müəyyən etmək üçün bir atribut modelləşdirməlidir.

QEYD ISO/TS 19103-də LocalizedCharacterString sinfinə baxın.

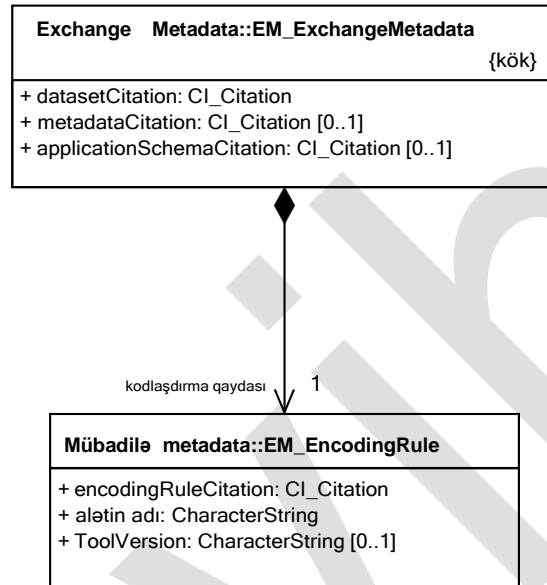
Kodlaşdırma qaydası istifadə olunan personajların repertuarını və simvolların kodlaşdırılmasını müəyyən etməlidir.

9.2.4 Metadata mübadiləsi

Mübadilə metadataları kodlaşdırılmış məlumat strukturu haqqında metadadır. Onlar verilənlər toplusunun yaradıcısını təsvir edər, verilənlər toplusu haqqında metadata məlumatına istinad edər, tətbiq sxemində istinad edər və tətbiq olunan kodlaşdırma qaydası haqqında məlumat verə bilər.

Kodlaşdırma qaydası mübadilə metadatasını və onun kodlaşdırılmış məlumat strukturunu necə müşayiət etdiyini müəyyən etməlidir.

Şəkil 6 ExchangeMetadata sinfinin və müvafiq EncodingRule sinfinin nümunəsini göstərir. CI_Citation sinfi ISO 19115-də müəyyən edilmişdir. Mübadilə faylı adətən ExchangeMetadata sinfinin bir nümunəsini ehtiva edir. "DatasetCitation" atributunda verilənlər toplusunun yaradıcısı haqqında məlumat var, "metadataCitation" atributunda verilənlər bazası haqqında metadatanın harada tapılması barədə məlumat ola bilər və "applicationSchemaCitation" atributunda tətbiq sxeminə istinad ola bilər. EncodingRule-un "encodingRuleCitation" atributu tətbiq edilən kodlaşdırma qaydasını və bu xüsusi halda qaydanın necə tətbiq olunduğunun təsvirini müəyyən edir. O, həmçinin verilənlər toplusunu kodlaşdırmaq üçün istifadə edilən kodlaşdırma xidməti aləti haqqında məlumatı ehtiva edə bilər, burada "toolName" və "toolVersion" kodlaşdırma xidməti alətinin adını və onun versiya nömrəsini göstərir.



Şəkil 6 — Mübadilə metaməlumatlarının nümunəsi

9.2.5 Transfer vahidi

9.2.5.1 Qranulyarlıq və quruluş

Ötürmə vahidinin qranulyarlığını və strukturunu müəyyən etmək vacibdir ki, o, effektiv şəkildə kodlaşdırıla və deşifrə edilsin. Burada obyekt məlumatın əsas vahidi hesab olunur və müxtəlif növ obyektlərin müəyyən edilməsi zəruridir. Köçürmə vahidindəki obyektlər ardıcıl və/yaxud iyerarxik şəkildə strukturlaşdırıla bilər. Obyekt daxili olaraq atributlar ardıcılığı kimi strukturlaşdırıla bilər, o, digər obyektlərə istinadları ehtiva edə bilər və iyerarxik şəkildə digər obyektlərdən də ibarət ola bilər.

Obyekt müxtəlif fraqmentlərə bölünə bilər və ya onun xassələri digər obyektlərlə birləşdirilə bilər. Kodlaşdırma qaydası strukturda hansı dəyişiklikləri göstərsə də, misalın çevrilməsi birmənalı şəkildə geri çevrilməlidir.

Kodlaşdırma qaydası aşağıdakıları müəyyən etməlidir:

- obyektin nə olduğunu və obyektin müxtəlif növlərini;
- obyektin quruluşu;
- verilənlər dəstinin strukturu.

9.2.5.2 Obyektin identifikasiyası

Obyektlərə müəyyən kontekstdə onların unikal identifikasiyasına imkan verən identifikatorlar təyin edilə bilər. İki fərqli kontekst nəzərə alınmalıdır.

- a) Birinci kontekst köçürmə vahididir. Burada obyekt identifikatorları məlumat ötürülməsinin xüsusi vahidi daxilində unikaldir. Bu identifikatorlar obyektlərə həmin bir ötürmə vahidi kontekstində digər obyektlərə istinad etməyə imkan verir. İdentifikatorlar ötürmə vahidinə daxil edildikdə və keçici xarakter daşdıqda obyektlərə təyin edilə bilər.
- b) İkinci kontekst tətbiq sahəsidir. Tətbiq domeni bir kainatı və unikal domen identifikatorları (DUIDs) adlanan identifikasiya konvensiyasını müəyyən edir. DUID obyekt yaradıldıqda ona təyin edilir və obyektin bütün ömrü boyu sabitdir. Silinmiş obyektin DUID-i yenidən istifadə edilməməlidir. DUID əsasən həyata keçirilməsi üçün istifadə olunur. DUID-lər uzunmüddətli paylanmış məlumatların idarə edilməsi və yeniləmə mexanizmlərinin həyata keçirilməsi üçün tələb olunur. Bu identifikatorlara davamlı identifikatorlar da deyilir. Davamlı identifikatorları həll etmək üçün xüsusi ad serverindən istifadə edilə bilər. İdentifikatorlar tətbiq sahəsi ilə müəyyən edilmiş yaxşı müəyyən edilmiş, məhdud kainat daxilində unikal olmalıdır.

Tətbiq sahəsi obyektlə təmsil olunan real dünya hadisələrini müəyyən etmək üçün əlavə xassələrdən istifadə edə bilər (məsələn, bağlama nömrəsi). Bu əlavə xüsusiyyətlər bu spesifikasiyaya çərçivəsində obyekt identifikatorları hesab edilmir.

Kodlaşdırma qaydası verilənlərin tərtibatını dəyişə bilər, məsələn, ümumi simvol sətirlərini nəzərə almaq. Bunu dəstəkləmək üçün tətbiq sxemində uyğun olaraq şəxsiyyəti olmayan növlər fraqment identifikatoru əldə edə bilər. Bu fraqment identifikatorları istənilən halda keçicidir (yalnız ötürmə vahidi daxilində etibarlıdır) və heç vaxt DUID deyil.

Kodlaşdırma qaydaları aşağıdakıları müəyyən etməlidir:

- istifadə olunan müxtəlif obyekt identifikasiyası mexanizmləri;
- onların daxili quruluşu.

9.3 Daxil edilən verilənlərin strukturu

Giriş verilənləri strukturu nümunə modeli adlanır və o, tətbiq məlumatları haqqında düşünmək, çevirmə qaydalarını müəyyən etmək və nümunələri ifadə etmək üçün bir vasitədir. Nümunə modeli tətbiq sxemindəki spesifikasiyaya uyğun olaraq məlumatları təmsil edə bilməlidir. Müəyyən bir tətbiq sxemində xas ola bilər və ya hər hansı bir sxemə uyğun olaraq məlumatları təmsil edə bilər. Bu, mücərrəd ola bilər, çünki kodlaşdırma xidmətini həyata keçirmək üçün onun həyata keçirilməsinə ehtiyac yoxdur.

Kodlaşdırma qaydası 8-ci bənddə müəyyən edildiyi kimi ümumi nümunə modelinə istinad etməli və ya nümunə modelini və onun tətbiq sxemi ilə əlaqəsini göstərməlidir.

Bu Beynəlxalq Standart nümunə modelinin xüsusi tətbiqini və ya verilənlərin kodlaşdırma xidmətindən/kodlaşdırma xidmətindən necə ötürülməsini tələb etmir.

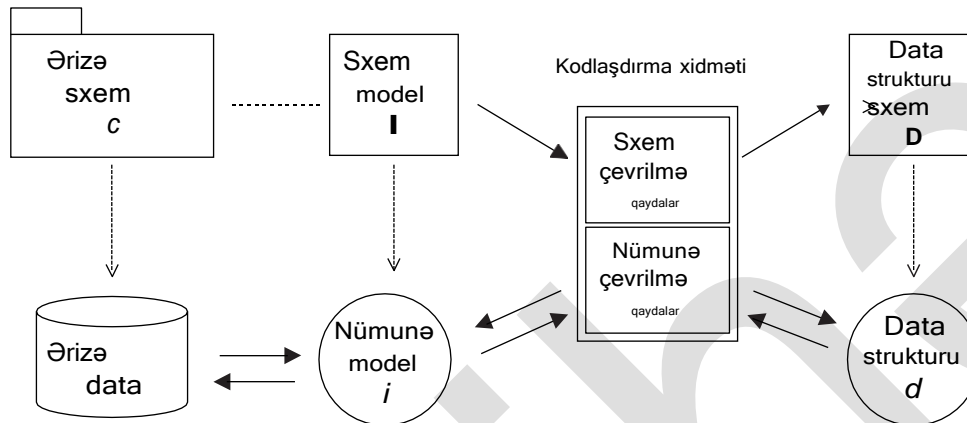
9.4 Çıxış məlumatlarının strukturu

Çıxış verilənləri strukturu verilənlərin necə strukturlaşdırıldığını və mübadilə faylında təmsil olunduğunu müəyyən edir. Çıxış məlumat strukturunu bir sxem müşayiət edə bilər.

Kodlaşdırma qaydası çıxış məlumat strukturunu və əgər varsa, çıxış məlumat strukturunu sxemini müəyyən etməlidir.

9.5 Dönüşüm qaydaları

Dönüşüm qaydası, giriş məlumatı strukturundakı məlumat nümunəsinin çıxış məlumat strukturunda verilənlər nümunəsinə necə çevrildiyini müəyyən edir. İki çevirmə qaydaları dəsti mövcud ola bilər. Birincisi, UML sxemindən çıxış məlumat strukturunun sxemində xəritəçəkməni təyin edən sxemlərin çevrilməsi qaydaları toplusudur. İkincisi, nümunə modelindəki nümunələrdən nəticədə əldə edilən məlumat modelindəki nümunələrə xəritələşdirməni müəyyən edən nümunə çevirmə qaydaları toplusudur. Şəkil 7 müxtəlif çevirmə qaydalarını göstərir.



Şəkil 7 — Dönüşüm qaydaları

Kodlaşdırma qaydası aşağıdakıları müəyyən etməlidir:

- sxemlərin çevrilməsi qaydaları;
- nümunə çevirmə qaydaları.

9.6 Nümunələr

Nümunələr çevirmə qaydalarını başa düşmək və kodlaşdırma xidmətlərini sınaqdan keçirmək üçün vacibdir.

Kodlaşdırma qaydası aşağıdakıları təmin etməlidir:

- çevirmə qaydalarını göstərən nümunələr;
- kodlaşdırma xidmətlərinin həyata keçirilməsində istifadə edilə bilən test məlumatları.

10 Kodlaşdırma xidməti

Kodlaşdırma xidməti kodlaşdırma qaydasını həyata keçirən və onun funksionallığı üçün interfeys təmin edən program komponentidir.

Kodlaşdırma xidməti bir və ya bir neçə interfeys spesifikasiyası vasitəsilə öz funksionallığına interfeys təqdim etməlidir.

Kodlaşdırma xidməti aşağıdakılardan birini və ya bir neçəsini təmin etməlidir:

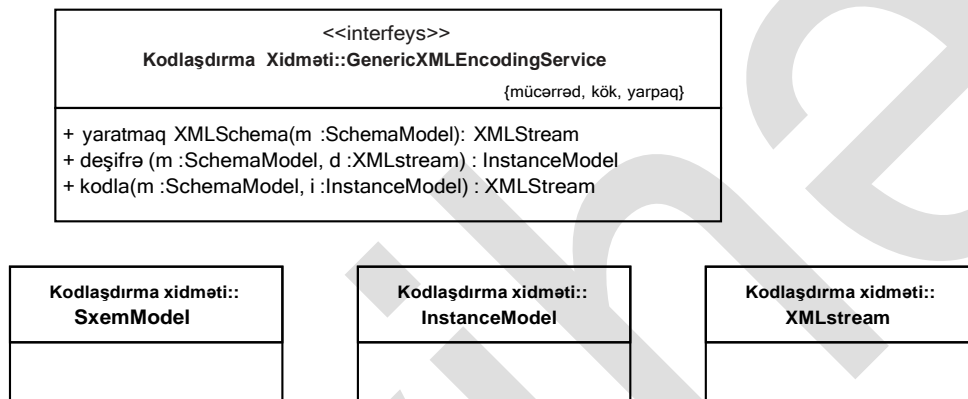
- nümunənin çevrilməsi qaydalarına uyğun olaraq məlumatları kodlaşdırmaq imkanı;
- nümunənin çevrilməsi qaydalarına uyğun olaraq məlumatların şifrəsini açmaq imkanı;
- sxemə çevirmə qaydalarına uyğun olaraq çıxış məlumat strukturunu sxemini yaratmaq imkanı.

Bu Beynəlxalq Standart kodlaşdırma xidmətinə xüsusi interfeys tələb etmir.

Kodlaşdırma xidməti yalnız kodlaşdırma və ya dekodlaşdırma imkanlarını təmin edə bilər.

Şəkil 8 XML əsaslı kodlaşdırma xidmətinin nümunəsidir.

QEYD Şəkil 8 bir interfeys spesifikasiyası ilə kodlaşdırma xidmətini göstərir. Üç əməliyyatı dəstəkləyir: yaratmaq, XMLSchema, kodlaşdırmaq və deşifrə etmək. XML Schema faylı yaratmaq üçün "generateXMLSchema" əməliyyatından istifadə edilə bilər. Bu əməliyyat giriş parametri kimi sxem modelini götürür və nəticədə XMLStream obyektini yaradır. XML sənədini yaratmaq üçün "kodlaşdırma" əməliyyatından istifadə etmək olar. O, giriş parametrləri kimi sxem modelini və nümunə modelini götürür və XMLStream obyektini qaytarır. "Deşifrə" əməliyyatı XML sənədini şərh etmək üçün istifadə edilə bilər. O, giriş kimi sxem modelini və XMLStream obyektini götürür və nümunə modelini qaytarır.



Şəkil 8 — Nümunə kodlaşdırma xidməti interfeysi

Əlavə A (normativ)

XML əsaslı kodlaşdırma qaydası

A.1 Bu Beynəlxalq Standartda XML kodlaşdırması Genişləndirilə bilən İşarələmə Dili (XML) 1.0-a uyğun olmalıdır. Bütün sxemlər üçün vahid XML əsaslı kodlaşdırma qaydasını təmin edən sabit kodlaşdırma tələbləri dəsti yoxdur. Nəticə olaraq, bu Beynəlxalq Standart sxemlərin çevrilməsi qaydasını müəyyən etmir. Hər bir XML əsaslı kodlaşdırma qaydası olmalıdır

- daxil məlumat strukturunu, adətən kodlaşdırma qaydası tərəfindən istifadə edilən UML profilini təyin edin və ya müəyyənəşdirin,
- XML çıxış məlumat strukturunu təyin edin,
- giriş və çıxış məlumat strukturları arasında sxem çevirmə qaydalarını müəyyən etmək və
- bu əlavədə göstərilən bütün tələblərə uyğun olmalıdır.

A.2 Fərqli istifadə halları fərqli kodlaşdırma qaydaları ilə nəticələnir. Hal-hazırda, "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar dəsti iki XML əsaslı kodlaşdırma qaydasını müəyyən edir:

- ISO 19136:2007, Əlavə E, XML sxemində çevrilməyə imkan verən məhdud UML profilindən istifadə etməklə təqdim edilə bilən ISO 19109-a uyğun tətbiq sxemləri üçün XML əsaslı kodlaşdırma qaydasını müəyyən edir. Kodlaşdırma qaydası əsasən xüsusiyyət növlərini və onların xassələrini təyin edən tətbiq sxemləri məqsədi ilə hazırlanmışdır. Kodlaşdırma qaydası çıxış məlumatı strukturu sxemi üçün XML sxemindən istifadə edir.
- ISO/TS 19139 coğrafi resursları, məsələn, ISO 19115-ə uyğun metadata və ISO 19110-a uyğun xüsusiyyət kataloqlarını təsvir edən konseptual sxemlər üçün XML əsaslı kodlaşdırma qaydasını müəyyən edir. Kodlaşdırma qaydası ümumi istifadə olunan UML modellərində istifadə olunan UML profilini dəstəkləyir. Texniki Komitə ISO/TC 211 tərəfindən hazırlanmış standartlarda. Kodlaşdırma qaydası çıxış məlumatı strukturu sxemi üçün XML Sxemindən istifadə edir.

A.3 Müxtəlif XML əsaslı kodlaşdırma qaydaları informasiya cəmiyyəti tərəfindən tələb oluna və müəyyən edilə bilər. Bu cür tələblərə misal olaraq aşağıdakıları əhatə edir, lakin bunlarla məhdudlaşmır:

- Əlavə C-də göstərilən XML əsaslı kodlaşdırma qaydası üçün dəstək (bu kodlaşdırma qaydası icmalar tərəfindən istifadə olunur);
- mövcud XML əsaslı kodlaşdırma qaydası ilə əhatə olunmayan fərqli UML profilinə dəstək;
- XML Sxemindən başqa çıxış məlumatı strukturu sxemi üçün dəstək (məsələn, Relax NG);
- yeni XML texnologiyalarına və ya mövcud XML texnologiyalarının yeni versiyalarına dəstək;
- XML imkanlarından istifadəni optimallaşdırmaq üçün xüsusi çevrilmələrə dəstək;
- icmada yaradılmış digər XML ilə bağlı tələblərə dəstək.

Əlavə B (normativ)

Abstrakt test kostyumu

B.1 Kodlaşdırma qaydası üçün sınaq halları

B.1.1 Ümumi

Bütün kodlaşdırma qaydaları, 2.2-ci bənddə müəyyən edilmiş uyğunluq siniflərinə uyğun olaraq, B.1-dən B.3-ə qədər olan mücərrəd test paketinin bütün sınaq işlərindən keçməlidir.

B.1.2 Konversiya qaydalarının sənədləşdirilməsi

- Test məqsədi: Kodlaşdırma qaydasının nümunə və ya şemaya çevrilmə qaydalarını müəyyən etdiyini yoxlayın.
- Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- İstinad: 9.5.
- Test növü: Əsas.

B.1.3 Ardıcıl instansiya və sxemlərin çevrilməsi qaydaları

- Test məqsədi: Əgər kodlaşdırma qaydası sxem və misal çevirmə qaydalarını müəyyən edirsə, onların ardıcıl olduğunu yoxlayın.
- Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- İstinad: 9.5.
- Test növü: Bacarıq.

B.1.4 Bit və bayt sıralaması

- Test məqsədi: Kodlaşdırma qaydasının bit və bayt sıralamasını müəyyən etdiyini və ya onu icra zamanı təyin etmək üçün metadata təyin etdiyini yoxlayın.
- Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- İstinad: 9.2.2, 9.2.4.
- Test növü: Bacarıq.

B.1.5 Simvol dəsti və kodlaşdırma

- Test məqsədi: Kodlaşdırma qaydasının xarakterlər dəstinə və kodlaşdırmanı müəyyən etdiyini və ya onu icra zamanı təyin etmək üçün metadata təyin etdiyini yoxlayın.
- Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- İstinad: 9.2.3, 9.2.4.
- Test növü: Bacarıq.

B.1.6 Eyniləşdirmə konvensiyası

- a) Test məqsədi: Kodlaşdırma qaydasının identifikasiya konvensiyasını təyin etdiyini və ya onu icra müddətində təyin etmək üçün metadada müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: 9.2.4, 9.2.5.2.
- d) Test növü: Bacarıq.

B.1.7 Metadatanın kodlaşdırılması

- a) Test məqsədi: Kodlaşdırma qaydasının kodlaşdırma qaydasında sabitləşdirilməmiş icra zamanı kodlaşdırma aspektlərini müəyyən etmək üçün metadada müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: 9.2.2, 9.2.3, 9.2.4, 9.2.5.2, 9.5.
- d) Test növü: Bacarıq.

B.2 Nümunələrin çevrilmə qaydaları üçün sınaq halları

B.2.1 Nümunələrin çevrilməsinin sənədləşdirilməsi

- a) Test məqsədi: Kodlaşdırma qaydasının ümumi nümunə modelinin nümunələrinin transfer formatına necə uyğunlaşdırıldığını müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: 9.5.
- d) Test növü: Bacarıq.

B.2.2 Nümunələrin çevrilməsinin tamlığı

- a) Test məqsədi: Kodlaşdırma qaydasının tam ümumi nümunə modeli üçün nümunə çevirmə qaydalarını müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: Maddə 8 və 9.5.
- d) Test növü: Bacarıq.

B.2.3 Birmənalı instansiya çevrilməsi

- a) Test məqsədi: Kodlaşdırma qaydasının məlumat itkisi olmadan ümumi nümunə modelindən köçürmə formatına və yenidən geri qaytaran nümunənin konvertasiya qaydalarını müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: 9.2.5.1.
- d) Test növü: Bacarıq.

B.3 Sxemlərin çevrilməsi qaydaları üçün sınaq nümunələri

B.3.1 Sxemanın çevrilməsinin sənədləşdirilməsi

- a) Test məqsədi: Kodlaşdırma qaydasının proqram sxeminin köçürmə formatı sxeminə necə uyğunlaşdırıldığını müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: 9.5.
- d) Test növü: Bacarıq.

B.3.2 Sxemanın çevrilməsinin tamlığı

- a) Test məqsədi: Kodlaşdırma qaydasının ISO/TS 19103 tərəfindən müəyyən edilmiş tam UML profili və ya onun profili üçün sxem çevirmə qaydalarını müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: 9.5.
- d) Test növü: Bacarıq.

B.3.3 Birmənalı instansiya çevrilməsi

- a) Test məqsədi: Kodlaşdırma qaydasının məlumat itkisi olmadan ümumi nümunə modelindən köçürmə formatına və yenidən geriye nümunənin xəritələşdirilməsi ilə nəticələnən sxem çevirmə qaydalarını müəyyən etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma qaydalarının sənədlərini yoxlayın.
- c) İstinad: 9.2.5.1.
- d) Test növü: Bacarıq.

B.4 Kodlaşdırma xidməti üçün sınaq halları

B.4.1 Xidmət interfeysinin sənədləşdirilməsi

- a) Test məqsədi: Kodlaşdırma xidmətinin sənədləşdirilmiş interfeysi təmin etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma xidməti sənədlərini yoxlayın.
- c) İstinad: Maddə 10.
- d) Test növü: Əsas.

B.4.2 Kodlaşdırma qaydasına istinad

- a) Test məqsədi: Kodlaşdırma xidməti sənədlərinin həyata keçirdiyi kodlaşdırma qaydasına istinad etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma xidməti sənədlərini yoxlayın.
- c) İstinad: Maddə 10.
- d) Test növü: Əsas.

B.4.3 Müəyyən edilmiş kodlaşdırma qaydasının həyata keçirilməsi

- a) Test məqsədi: Kodlaşdırma xidmətinin istinad edilən kodlaşdırma qaydasını tətbiq etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma xidmətinin həyata keçirilməsini yoxlayın.
- c) İstinad: Maddə 10.
- d) Test növü: Bacarıq.

B.5 İstənilən proqram sxeminin dəstəklənməsi

- a) Test məqsədi: Kodlaşdırma xidmətinin kodlaşdırma qaydası tərəfindən istifadə edilən UML profili ilə müəyyən edilmiş istənilən proqram sxemini dəstəklədiyini yoxlayın.
- b) Test üsulu: Ümumi nümunə modelini dəstəklədiyini görmək üçün kodlaşdırma xidməti interfeysini yoxlayın.
- c) İstinad: 8-ci bənd.
- d) Test növü: Bacarıq.

B.6 Məlumatların kodlaşdırılması

- a) Test məqsədi: Kodlaşdırma xidmətinin məlumat yazmaq üçün funksionallığı təmin etdiyini yoxlayın.
- b) Test üsulu: Məlumatların yazılması üçün funksionallığı təmin edib-etmədiyini görmək üçün kodlaşdırma xidməti interfeysini yoxlayın.
- c) İstinad: Maddə 10.
- d) Test növü: Bacarıq.

B.7 Məlumatların dekodlanması

- a) Test məqsədi: Kodlaşdırma xidmətinin məlumatları oxumaq üçün funksionallıq təmin etdiyini yoxlayın.
- b) Test üsulu: Məlumatların oxunması üçün funksionallıq təmin edib-etmədiyini görmək üçün kodlaşdırma xidmətinin interfeysini yoxlayın.
- c) İstinad: Maddə 10.
- d) Test növü: Bacarıq.

B.8 Sxemanın yaradılması

- a) Test məqsədi: Kodlaşdırma xidmətinin format sxemini yaratmaq üçün funksionallıq təmin etdiyini yoxlayın.
- b) Test üsulu: Kodlaşdırma xidməti interfeysini yoxlayın ki, o, format sxeminin yaradılması funksiyasını təmin edir.
- c) İstinad: Maddə 10.
- d) Test növü: Bacarıq.

Əlavə C

(məlumatlandırıcı)

XML əsaslı kodlaşdırma qaydası icmalar tərəfindən istifadə olunur

C.1 Giriş

Bu əlavə bəzi icmalarda neytral məlumat mübadiləsi üçün XML əsaslı kodlaşdırma qaydasının nümunəsini təqdim edir. Kodlaşdırma qaydası Vahid Modelləşdirmə Dili (UML) ilə uyğun gəlir və Genişləndirilə bilən İşarələmə Dilinə (XML) əsaslanan kodlaşdırma qaydasını müəyyən edir.

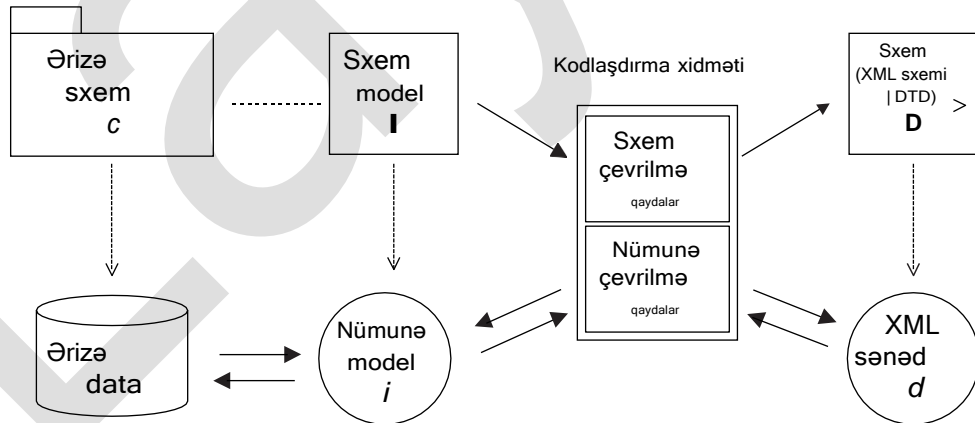
QEYD ISO 19118:2005, Əlavə A, XML əsaslı kodlaşdırma qaydasını müəyyən etdi. Qayda bəzilərində istifadə edilmişdir icmalar. Bu Əlavə C bu Beynəlxalq Standarta uyğun olaraq icmalar üçün qaydanın yenidən nəzərdən keçirilməsidir.

Bu əlavə 9-cu bəndin tələblərinə uyğundur və aşağıdakıları müəyyən edir:

- C.2-də ümumi kodlaşdırma tələbləri;
- C.3-də daxil olan verilənlərin strukturu;
- C.4-də çıxış məlumat strukturu;
- C.5 və C.6-dakı çevirmə qaydaları.

Nümunələr C.5.8 və C.6.4-də verilmişdir.

Dönüşüm qaydaları tətbiq sxemindəki sinif təriflərinin XML Sxemindəki tip bəyannamələri ilə əlaqələndirilməsi və nümunə modelindəki obyektlərin XML sənədindəki müvafiq element strukturları ilə əlaqələndirilməsi ideyasına əsaslanır. Şəkil C.1 iki növ çevirmə qaydasını təsvir edir.



Şəkil C.1 — XML əsaslı çevirmə qaydaları

C.2 Ümumi kodlaşdırma tələbləri

C.2.1 Tətbiq sxemi

C.2.1.1 Giriş

Tətbiq sxemi ISO/TS 19103 və ISO 19109-da müəyyən edilmiş qaydalara uyğun olaraq UML sxem dilində ifadə edilməlidir. Tətbiq sxemi siniflər və assosiasiyalar kimi ifadə edilən tətbiq tərəfindən müəyyən edilmiş anlayışlardan ibarətdir. Bu siniflərdən bəziləri digər standartlarda standartlaşdırılmış sxemlərdən idxal edilə bilər.

C.2.1.2 Sinif

UML-də əsas modelləşdirmə konsepsiyası sinifdir. Bu adla müəyyən edilir və stereotipi ola bilər. Sinfin atributları, əməliyyatları və məhdudiyyətləri ola bilər və assosiasiyalarda iştirak edə bilər. Sinif digər sinifləri müəyyən etmək üçün tikinti bloku kimi istifadə edilə bilən bir növü müəyyən edir. Növlər verilənlərin bütün formalarının, məsələn, ədədlər, koordinatlar, mətn sətirləri, tarixlər və obyektlər kimi tərtib oluna və kodlaşdırıla bilən əsas tikinti bloklardır. Növ hüquqi dəyər sahəsini və bu domenin dəyərləri üzərində əməliyyatları müəyyən edir.

Tiplərin iki əsas kateqoriyası sadə məlumat növləri və mürəkkəb məlumat növləridir. Tip üçün müəyyən edilmiş kanonik kodlaşdırma varsa, bir növ sadə məlumat növüdür. Bu kanonik kodlaşdırma növün dəyərlərini yaddaşdakı bitlər və ya mətn kodlaşdırmasındakı simvollar kimi necə təqdim edəcəyini müəyyən edə bilər. Sadə növlərə tam, float və sətir misal ola bilər. Tip üçün müəyyən edilmiş kanonik kodlaşdırma olmadıqda, bir növ mürəkkəb məlumat növüdür. Nümunələr obyekt növləri, strukturlaşdırılmış tiplər, qeydlər və kolleksiyalardır. Müəkkəb tip əsas növlərin və xüsusi strukturlaşdırma primitivlərinin birləşməsindən istifadə etməklə kodlaşdırıla bilən əsas və mürəkkəb atributların strukturlaşdırılmış toplusundan ibarətdir. Üçüncü kateqoriya "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar dəstindən kənarında müəyyən edilmiş xarici məlumat növləridir.

a) Sadə məlumat növləri - Dəyərləri təmsil etmək üçün əsas tiplər:

- 1) əsas məlumat növləri: CharacterString, Integer, Binary, Boolean, Date, Time və s.;
- 2) sadalanan məlumat növləri və kod siyahıları: Hüquqi dəyərlərin siyahısı, burada hər bir dəyər əlaqəli semantika ilə bir söz və ya koddur.

b) Müəkkəb məlumat növləri - Dəyərlərin daha mürəkkəb kolleksiyalarını təmsil etmək üçün növlər:

- 1) toplanması məlumat növləri: Digər növlərin çoxsaylı hadisələrini təmsil etmək üçün şablon növləri - Set, Çanta, Ardıcılıq, Lüğət və s.;
- 2) strukturlaşdırılmış məlumat növləri: Atribut qruplarını müəyyən edən tiplər;
- 3) obyekt növləri: Nümunələri obyekt olan tiplər; tez-tez tətbiq sxemlərində və ya standartlaşdırılmış sxemlərdə müəyyən edilir - GM_Point, Bina və s.;
- 4) interfeyslər: Nümunələri xidmət komponentləri olan növlər.

c) Xarici məlumat növləri - "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar toplusunda müəyyən edilməyən, dəqiq müəyyən edilmiş kodlaşdırma malik əsas və ya mürəkkəb növlər; nümunələr NASA MODIS, TIFF və s. kimi şəkil formatlarıdır. Xarici məlumat növlərinə adətən ayrı-ayrı fayllarda saxlanılan istinadlara imkan verən xüsusi istinad mexanizmləri müəyyən edilməlidir.

ISO/TS 19103 tətbiq olunan modelləşdirmə konsepsiyalarını, o cümlədən bir sıra sadə və mürəkkəb məlumat növlərini semantikası ilə müəyyən edir. Proqram tərtibatçıları UML-in stereotip genişləndirmə mexanizmindən istifadə edərək istifadəçi tərəfindən müəyyən edilmiş məlumat növlərini və obyekt növlərini müəyyən etməkdə sərbəstdirlər. "ISO 19100 seriyası" kimi tanınan Beynəlxalq Standartlar dəsti daxilindəki digər Beynəlxalq Standartlar daha çox ixtisaslaşmış məlumat növlərini müəyyən edir. Tətbiq sxemi tərtibatçıları üçün bəzən xarici məlumat növlərindən istifadə etmək lazım ola bilər. Xarici məlumat növləri bu Beynəlxalq Standartlar seriyasından kənarında müəyyən edilmiş əsas və ya strukturlaşdırılmış məlumat növləridir. bu

onların dəqiq müəyyən edilmiş kodlaşdırmaya malik olması zəruridir; nümunələr NASA MODIS və TIFF kimi şəkil formatlarıdır. Xarici məlumat növlərinin obyektləri ayrı-ayrı fayllarda saxlanıla bilər və xarici məlumat növlərinə istinad etməyə imkan verən xüsusi istinad mexanizmləri müəyyən edilməlidir.

Dərslərdə icazə verilən stereotiplər Cədvəl C.1-də təsvir edilmişdir.

Cədvəl C.1 — Siniflər üzrə stereotiplər

Stereotip	Təsvir
<<BasicType>>	Kanonik kodlaşdırmanı təyin edən əsas məlumat növünü müəyyən edir
<<Məlumat Tipi>>	Strukturlaşdırılmış məlumat növünü müəyyən edir. Nümunələr obyekt hesab edilmir və buna görə də şəxsiyyətə malik olmamalıdır
<<Birlik>>	Birlik məlumat növünü müəyyən edir
<<Sadalama>>	Sadalanan məlumat növünü müəyyən edir
<<Kod Siyahısı>>	Kod və dəyər cütlərindən ibarət uzadılan sadalanan məlumat tipini müəyyənləşdirir
<<İnterfeys>>	Xidmət interfeysini müəyyən edir və kodlaşdırılmamalıdır
<<Növ>>	Obyekt tipini müəyyənləşdirir. Nümunələrin şəxsiyyəti olmalıdır
<<Feature Type>>	Xüsusiyyət növünü müəyyən edir. Nümunələrin şəxsiyyəti olmalıdır
YOX	Obyekt tipini müəyyənləşdirir. Nümunələrin şəxsiyyəti olmalıdır

Obyekt əsas mübadilə vahidi hesab olunur. Məlumat mübadiləsi məqsədləri üçün yalnız obyektin vəziyyətini tutmaq üçün vacib olan aspektlər nəzərə alınmalıdır. Atributlar və assosiasiyalar kodlaşdırılmalıdır. Əməliyyatlar və məhdudiyətlər əlavə nəzərə alınmayacaq. Kodlaşdırma qaydası atributların və assosiasiyaların necə strukturlaşdırıldığını və təmsil olunduğu daxil olmaqla, sinif nümunələrinin necə təmsil olunduğunu müəyyən etməlidir.

C.2.1.3 Atributlar

Bir atribut adla müəyyən edilir. Onun çoxluq ifadəsi ola bilər və həmişə bir növü olmalıdır. Çoxluq ifadəsi konkret atributun hüquqi dəyərinin baş vermə sayını göstərməlidir.

Çoxluğun birləşməsi və kolleksiya məlumat növlərinin istifadəsi dəyərlərin yuvalanmasına imkan verir.

Null dəyərləri idarə etmək üçün mexanizmlər olmalıdır. Növ null dəyərin necə təmsil olunduğunu müəyyən etməlidir. Əsas və toplama məlumat növlərinin tərfi üçün ISO/TS 19103-ə baxın.

C.2.1.4 Assosiasiyalar

Assosiasiyalar, onların nümunələri arasında əlaqələri əhatə edən siniflər arasındakı əlaqələri müəyyənləşdirir. Birliyin instansiyasına keçid deyildir. Link obyektlərə istinadların ardıcıl siyahısını ehtiva edir. UML assosiasiya, aqreqasiya və kompozisiya adlanan üç müxtəlif assosiasiya növünü müəyyən edir.

- Assosiasiya siniflər arasında ümumi əlaqələri müəyyən edir.
- Aqreqasiya siniflər arasında zəif hissə-bütöv münasibətləri müəyyən edir.
- Kompozisiya siniflər arasında güclü bir hissə-bütöv əlaqələri müəyyənləşdirir.

Üç assosiasiya növü fərqli semantikaya malikdir. ISO/TS 19103 daha ətraflı məlumat verir.

Assosiasianın sonu onun rol adı və hədəf sinfi ilə müəyyən edilir və daha sonra çoxluq ifadəsi ilə təsvir olunur. Rol adı keçidi təmsil etmək üçün istifadə edilməlidir. Rol adı yoxdursa, deməli heç bir əlaqə yoxdur.

C.2.2 Mübadilə strukturu və metadata mübadiləsi

Mübadilə strukturu üç hissəyə bölünməlidir. Birinci hissə mübadilə metaməlumatıdır və 9.2.4-ə uyğun olaraq təsvir edilməlidir. İkinci hissə müstəqil obyektlərə uyğun gələn XML elementlərini ehtiva edən verilənlər toplusudur. Üçüncü hissə isteğe bağlıdır və C.2.5-də təsvir olunduğu kimi yeniləmə primitivlərinin yeniləmə bölməsini ehtiva etməlidir.

C.2.3 Xarakter repertuarı və dili

Əsas mübadilə strukturu dil və xarakter repertuarını dəstəkləməlidir:

- dil (dilin etiketlenməsi üçün C.5.6-ya baxın);
- xarakter repertuarı (bax: ISO/IEC 10646).

C.2.4 Məlumat toplusu və obyektin identifikasiyası

9.2.5.2-də təsvir olunduğu kimi obyektin eyniləşdirilməsinin iki yolu aşağıdakı kimi olmalıdır:

- XML-in ID mexanizminə uyğun olaraq unikal verilənlər bazası identifikatorları;
- proqram domeninin spesifikasiyasına uyğun universal unikal identifikatorlar.

QEYD "Universal unikal identifikator" termini "domen unikal identifikatoru" ilə eyni məna daşıyır. Bu əlavə ondan istifadə edir və "UUID" istifadə olunan icmalar üçün uyğunluğu saxlamaq üçün qısaltması kimi. Bu Beynəlxalq Standartda UUID-lərin tətbiqi ISO/IEC 11578 və ya ISO/IEC 9834-8 kimi xüsusi riyazi əsaslarla məhdudlaşmır. "Universal unikal identifikator" tətbiq domeni tərəfindən təsdiq edilən "domen unikal identifikatorundan" fərqli olaraq, əslində proqram domenləri tərəfindən uyğunlaşdırılırlar.

NÜMUNƏ Tətbiq domeni iki komponentli identifikatordan istifadə etmək istəyə bilər. Birinci komponent domendir ad və növbəti komponent tam ədəd nümunə nömrəsidir. Komponentlər iki nöqtə ":" ilə ayrılır. Nümunələrin nömrələri hex ilə kodlanı bilər. Nümunə nömrəsinin ölçüsündə heç bir məhdudiyət yoxdur. "Nümunə" adlı domendə iki UUID nümunələri bunlardır: "nümunə:F23C30" və "nümunə:FFFFFF12345A".

C.2.5 Yeniləmə mexanizmi

Yeniləmə mexanizmi əvvəllər mübadilə edilmiş məlumatların tam yeni verilənlər toplusunun yenidən buraxılması tələbi olmadan yenilənməsinə imkan verir. Yeniləmə üçün siyasət və prosedurlar xüsusi tətbiq tərəfindən müəyyən edilir. Üç əsas yeniləmə primitivləri adətən müəyyən edilir: *əlavə edin*, *dəyişdirmə* və *silin*. Bu primitivlər obyekt səviyyəsində işləyir, lakin atribut və ya assosiasiya səviyyəsində işləmək üçün də müəyyən edilə bilər. Əvvəllər UUID ilə ötürülən hər hansı obyekt dəyişdirilə və ya silinə bilər. Yeniləmə verilənlər toplusu yeniləmə primitivlərinin ardıcıl ardıcılığını ehtiva edir. Əsas primitivlər aşağıdakı kimi təsvir edilmişdir.

- əlavə edin*: Mənbə verilənlər bazasına yeni obyekt əlavə edildi və hədəf verilənlər dəstinə əlavə edilməlidir. Əlavə primitivi əlavə ediləcək yeni obyekt haqqında məlumatı ehtiva etməlidir və onun hədəf verilənlər bazasına daxil edildiyi yer haqqında məlumatı ehtiva edə bilər.
- dəyişdirmə*: Mövcud obyekt mənbə verilənlər bazasında dəyişdirilib və hədəf verilənlər dəstində dəyişdirilməlidir. Dəyişdirmə primitivində hədəf obyektini və faktiki dəyişiklikləri müəyyən edən məlumat olmalıdır. Modifikasiya məlumatlarının nümunələri tam obyektədən yenilənmiş atributlara qədər dəyişə bilər.
- silin*: Mövcud obyekt mənbə verilənlər bazasında silinib və buna görə də hədəf verilənlər bazasında silinməlidir. Silinmə primitivində silinəcək hədəf obyektini müəyyən edən məlumat olmalıdır.

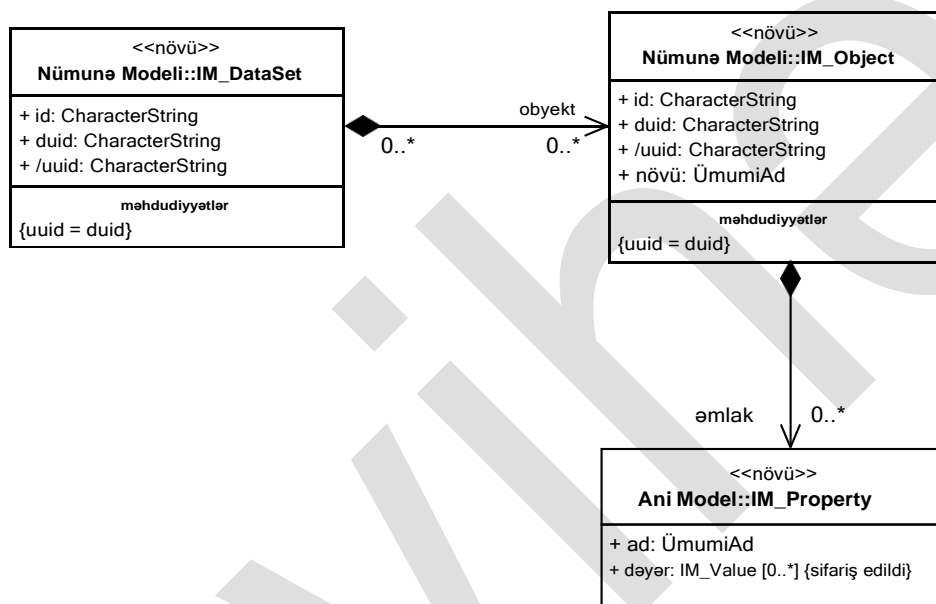
İstifadəçilər primitivlərin siyahısını genişləndirə bilərlər.

C.3 Daxil edilən verilənlərin strukturu

C.3.1 Nümunə modeli

Nümunə modeli UML-də ifadə olunan tətbiq sxemləri ilə təsvir edilən məlumatları təmsil etməyə qadirdir. Giriş verilənləri strukturu olaraq, bu kodlaşdırma qaydası 8-ci bənddə müəyyən edilmiş ümumi nümunə modelinə əsaslanan nümunə modelindən istifadə edir. Nümunə modeli Şəkil C.2 və C.3-də müəyyən edilmişdir.

Şəkil C.2-də IM_Dataset, IM_Object və IM_Property sinifləri göstərilir ki, onlar əsasən 8.1-də və Şəkil 4-də müəyyən edilənlərlə eynidir.



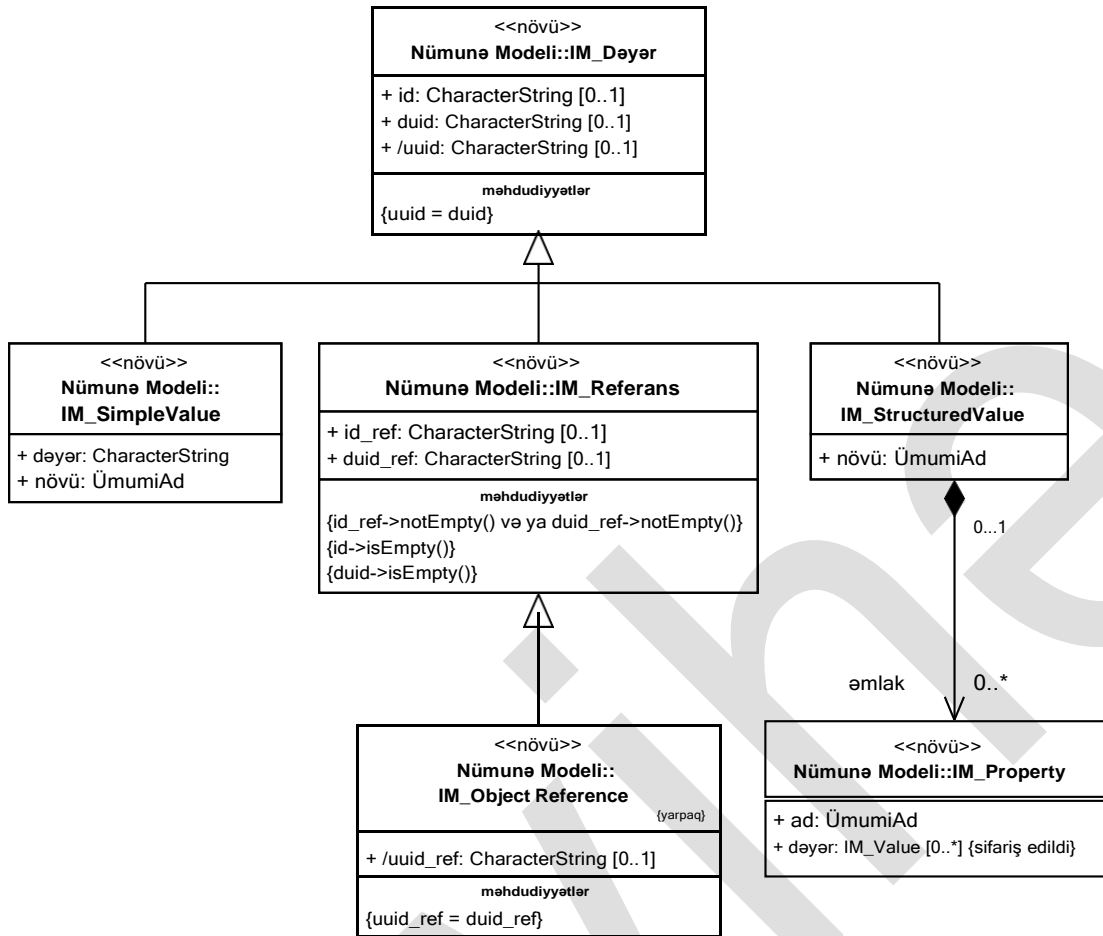
Şəkil C.2 — Nümunə modeli—Dataset, obyekt və mülkiyyət

Nümunə modeli IM_Dataset ilə təmsil olunan verilənlər toplusundan ibarətdir. Dataset IM_Object ilə təmsil olunan obyektlərin ardıcılığını ehtiva edir. Obyekt IM_Property ilə təmsil olunan xassələr ardıcılığından ibarətdir.

IM_Dataset və IM_Object-də uuid atributunun əlavə tərfi var. Alınmış atribut uuid atributunun alternativ adı kimi müəyyən edilir.

QEYD 1 Uuid atributunun tərfi istifadə olunan icmalar üçün uyğunluğu saxlamaqdır.

Nümunə modelindəki dəyər növləri Şəkil 5-ə əsaslanan Şəkil C.3-də müəyyən edilmişdir.



Şəkil C.3 — Nümunə modeli—Dəyər növləri

IM_Value aşağıdakı kimi müəyyən edilmiş dörd dəyər növünün supersinifidir.

- IM_SimpleValue əsas növün dəyərini təmsil edir.

NÜMUNƏ Tam ədəd və ya simvol sətri.

- IM_Reference hədəf obyektinə keçid və ya istinadı təmsil edir. Hədəf obyekt eyni və ya digər verilənlər bazasında yerləşə bilər.
- IM_ObjectReference IM_Reference-in alt növüdür. Unikal identifikator (id_ref) eyni verilənlər bazasında yerləşən obyektə hədəfləyir. Universal unikal identifikator (uuid_ref və duid_ref) proqram domeninin kontekstində yerləşən obyektə hədəfləyir.
- IM_StructuredValue mürəkkəb məzmunlu məlumat növü dəyərini təmsil edir [xassələr ardıcılığı (IM_Property)].

Bu nümunə modeli ilə 8-ci bənddə müəyyən edilmiş ümumi nümunə modeli arasındakı fərq aşağıdakı kimidir.

- IM_Value atributunun alternativ adı kimi uuid əldə edilmiş atributunu ehtiva edir.
- IM_ObjectReference müəyyən edilib və o, duid_ref atributunun alternativ adını təyin edən əlavə törəmə atributuna malikdir.

Şəkil C.3-də müəyyən edilmiş digər siniflər 8.1 və Şəkil 5-də müəyyən edilənlərlə eynidir.

QEYD 2 IM_ObjectReference sinifinin və IM_Value və IM_ObjectReference-da uuid atributunun tərfi istifadədə olan icmalar üçün uyğunluğu saxlamaqdır.

C.3.2 UML və nümunə modeli arasında əlaqə

C.2 və C.3 Cədvəlləri UML ilə nümunə modeli arasındakı əlaqənin xülasəsini verir. Beləliklə, mücərrəd sinif yaradılmamalıdır.

Üst tipləri olan siniflərə əsaslanan obyektlər öz sinifinin və supertiplərinin bütün xüsusiyyətlərini, birləşmələrini və kompozisiyalarını ehtiva etməlidir. Beləliklə, bütün atributlar və assosiasiyalar supertiplərdən kopyalanmalı və obyektin bir hissəsi hesab edilməlidir. Atribut və assosiasiya adları atributların dəyərlərinə çıxış yolu olmalıdır və buna görə də onlar sinif daxilində unikal olmalıdır.

Əməliyyatlar və məhdudiyətlər nümunə modelinə uyğunlaşdırılmamalıdır.

Cədvəl C.2 — UML və nümunə modeli arasındakı əlaqənin xülasəsi

UML konsepsiyası	Nümunə modeli
Paket	Yoxdur
Sinif Stereotip <<Interfeys>> <<BasicType>> <<Məlumat Tipi>> <<Birlik>> <<Sadalama>> <<Kod Siyahısı>> <<Növ> və ya NONE Abstrakt sinif Alt sinif	Yoxdur IM_SimpleValue IM_StructuredValue IM_StructuredValue IM_SimpleValue IM_SimpleValue IM_Obyekt Yoxdur atributlar və assosiasiyalar super siniflərdən kopyalanmalıdır
Atribut	Məlumat növünə görə IM_Value ilə IM_Property (ya IM_SimpleValue, ya da IM_StructuredValue)
Assosiasiya	IM_ObjectReference IM_Value ilə IM_Property
Toplama	IM_StructuredValue və ya IM_ObjectReference uyğun olaraq IM_Value ilə IM_Property
Tərkibi	IM_StructuredValue IM_Value ilə IM_Property
Əməliyyat	Yoxdur
Məhdudiyət	Yoxdur
^a	N/A tətbiq edilə bilməz deməkdir.

Cədvəl C.3 — Çoxluq və kolleksiya növü ilə atributların xəritələşdirilməsi

Atribut	Nümunə modeli
a1 [0..*] : Tam ədəd ilə eynidir a1 : Ardıcılıq<tam>a	Çoxsaylı dəyər hadisələri ilə IM_Property
a2 : Ardıcılıq<T>	IM_T dəyər növünün çoxsaylı təkrarlanması ilə xassə
a3 : Lüğət<T1, T2>	T1 və T2 dəyər tipli iki elementi olan IM_StructuredValue-nin çoxsaylı təkrarlanması ilə IM_Property
a4 [0..*] : Ardıcılıq<Tam ədəd>	Tam ədəd elementləri ilə IM_StructuredValue çoxlu təkrarlanan IM_Property
a Bu, hər hansı digər əsas növ üçün də keçərlidir.	

Atribut və assosiasiya adı toqquşmaları mirasdan istifadə edərkən problemlər yarada bilər. Bunun qarşısını almağın sadə yolu bütün atributların və assosiasiyaların müvafiq sinif adı ilə prefiks olmasını təmin etməkdir; Alternativ olaraq, ad toqquşmalarının qarşısını almaq üsulu istifadəçinin ixtiyarına verilir.

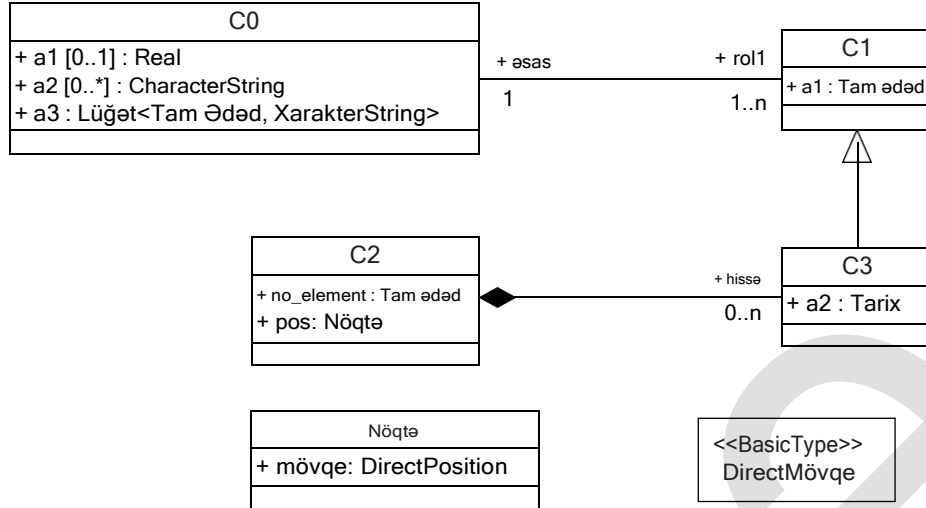
Atribut və assosiasiyaların yenidən bəyan edilməsi də mirasdan istifadə edərkən problemlər yarada bilər. Yenidən bəyan etmə supertipdə elan edilmiş atribut və ya assosiasiya yeni və ya məhdudlaşdırılmış tipli alt tipdə yenidən elan edildikdə baş verir. Bir çox obyekt yönümlü proqramlaşdırma dilləri yenidən bəyannamələri idarə edə bilmir və yenidən bəyannamələrin ləğv edilib-edilməməsi diqqətlə nəzərdən keçirilməlidir.

C.3.3 Tətbiq sxemi və nümunə modeli—Misal

C.3.3.1 Tətbiq sxemi

Bu nümunə proqram sxemini müəyyən edir, bəzi məlumatlar verir və verilənlərdən nümunə modelinə xəritələşdirməni göstərir.

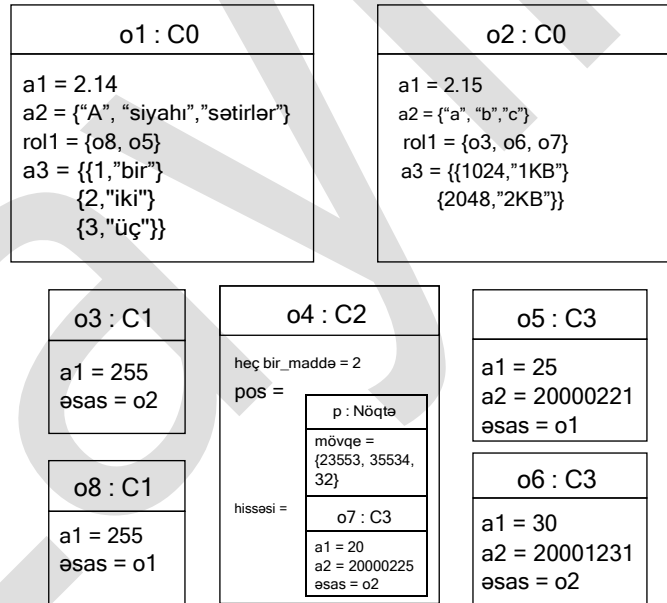
Şəkil C.4 dörd sinfi və onların əlaqələrini müəyyən edən nümunə tətbiq sxemini müəyyən edir. Burada C0 sinfinin üç atributu var. “a1” atributunun çoxluğu sıfır və ya birdir, yəni istəyə bağlıdır və əsas məlumat növü “Real”dır. “a2” atributunun çoxluğu sıfır və ya çoxdur. “a3” atributu strukturlaşdırılmış məlumat növüdür. “C0” və “C1” arasındakı əlaqənin iki rol adı var. “rol1” C0-a aiddir və C0 kontekstində assosiasiyayı C1 adlandırır. C1 sinfinin “əsas” adlanan C0 ilə əlaqəsi var. Diqqət yetirin ki, C2 sinfində Point sinif məlumat növü olan “pos” adlı atribut var.



Şəkil C.4 — Tətbiq sxemi—Misal

C.3.3.2 Məlumat—Misal

Nümunə verilənlər Şəkil C.5-də obyekt diaqramında göstərilmişdir.



Şəkil C.5 — Məlumat—Misal

C.4 Çıxış verilənlərinin strukturu

C.4.1 XML sənədi

Bu kodlaşdırma qaydası XML Tövsiyə 1.0-a əsaslanır. XML mətn formatıdır və bütün məlumat növlərinin dəyərlərinin simvollarla kodlaşdırılması zəruridir. Məlumatlar XML elementləri və XML tövsiyəsində verilmiş qaydalar əsasında kodlaşdırılmalıdır. XML-də kodlaşdırmanın əsas vahidləri XML elementləridir. Elementin atributları və məzmunu ola bilər. Bu, iyerarxik quruluşa imkan verir və XML-in əlaqələndirici vasitələri ilə birləşir və

şəbəkə strukturu yaradıla bilər. Mübadilə strukturu nümunə modelindəki obyektlərə uyğun gələn elementlər ardıcılığından ibarət olmalıdır.

Mübadilə formatının strukturunu tənzimləyən çıxış məlumat strukturu üçün sxem etibarlı XML Schema olmalıdır.

C.4.2 XML sxemi

XML Sxem Sənədi (XSD) bir sıra müəkkəb tipləri, sadə tipləri və XML sənədinin icazə verilən strukturunu və məlumat nümunələrini müəyyən edən element bəyannamələrini müəyyən edir. XML Şemasının çevrilmə qaydaları C.5-də müəyyən edilmişdir.

C.5 Şemanın çevrilməsi qaydaları

C.5.1 XML sxemi

Sxemaya çevrilmə qaydaları UML-də ifadə olunan tətbiq sxeminə uyğun olaraq XML Sxem Sənədinin (XSD) necə hazırlanacağını müəyyənləşdirir. XSD-nin əsas məqsədi məlumatların çevrilməsi qaydalarından istifadə etməklə hazırlanmış XML sənədlərinin etibarlı olmasını təmin etməkdir.

XSD proqram sxemində müəyyən edilmiş siniflərə uyğun gələn tip tərifləri və atribut və element bəyannamələrini ehtiva etməlidir. Elementlər birja strukturunda təşkil edilməlidir. XSD XML sxeminə 1-ci hissə: Strukturlar və 2-ci hissə: Məlumat növlərinə uyğun olmalıdır.

XSD fiziki olaraq bir sxem sənədində təmsil oluna və ya bir neçə ayrı (alt) sxem sənədlərinə bölünə bilər. Məntiqi olaraq, o, idxaldan istifadə edən və ya XML Schema mexanizmlərini daxil edən vahid sxem kimi istinad edilməlidir. XSD-də ad boşluqlarının istifadəsinə dair heç bir məhdudiyyət yoxdur.

Bir sıra ümumi qaydalar C.5.2-C.5.8-də müəyyən edilmişdir. Sınıf ümumiyyətlə C.5.2-yə uyğun tip tərifinə çevrilməlidir, C.5.4-ə uyğun olaraq element bəyannaməsinə çevrilə bilər və birja strukturunun üzvü ola bilər.

Ümumi qaydalardan istisnalara sənədləşdirildiyi müddətcə icazə verilir.

QEYD Aşağıda "xs:" ad sahəsi XML Şemasının ad sahəsinə istinad etmək üçün istifadə olunur. "http://www.w3.org/2001/XMLSchema".

C.5.2 Növlər

C.5.2.1 <<BasicType>>

C.5.2.1.1 Ümumi qayda

<<BasicType>> stereotipi sınıf XML Sxemasında simpleType bəyannaməsinə çevrilməlidir. XML Şemasında müəyyən edilmiş hər hansı məlumat növlərindən istifadəçi tərəfindən müəyyən edilmiş əsas növləri müəyyən etmək üçün tikinti blokları kimi istifadə edilə bilər. Əsas növlərin kodlaşdırılması XML Schema Hissə 2-də müəyyən edilmiş kanonik təsvirə uyğun olmalıdır: Məlumat tipləri.

Əsas növlər üçün modelləşdirmə qaydaları ISO/TS 19103-də müəyyən edilmişdir. ISO/TS 19103-də müəyyən edilmiş əsas növlər C.5.2.1.2-də C.5.2.1.15-ə çevrilmişdir.

İstifadəçilərə XML sxemində müəyyən edilmiş məhdudiyyət mexanizmlərindən istifadə edərək əsas növləri daha da məhdudlaşdırmağa icazə verilir.

QEYD Müxtəlif növlər ISO/TS 19103-də dəqiq müəyyən edilməmişdir və nə də <<BasicType>> stereotipindən istifadə edilmir. Buna görə də, aşağıdakı bəyannamələr XML Schema Hissə 2-də verilənlər növü təriflərinin alt çoxluğuna uyğundur: Məlumat növləri.

C.5.2.1.2 Nömrə

Rəqəm ISO/TS 19103-də mücərrəd tip və Tam, Həqiqi və Ondalığın super növü kimi müəyyən edilmişdir. Bununla belə, müxtəlif standartlaşdırılmış sxemlərdə çox istifadə olunur. XML Şeması mücərrəd ədəd tipini təyin etmir, lakin tam ədədin supertipi olmaq üçün onluq təyin edir.

QEYD Nömrə məlumat tipi, onluq verilənlər növünə əsaslanan sadə məlumat növü kimi elan edilir.

```
<xs:simpleType name="Nömrə">
  <xs:restriction base="xs:decimal"/> </
  xs:simpleType>
```

C.5.2.1.3 Tam ədəd

Tam ədəd XML Sxema tam məlumat tipinə əsaslanmalıdır. Dəyər sahəsi məhdudlaşdırıla bilər.

```
<xs:simpleType name="Integer">
  <xs:restriction base="xs:integer"/>
</xs:simpleType>
```

C.5.2.1.4 Onluq və Real

Ondalığ və Həqiqi növlər hər ikisi XML Şeması onluq tipinə əsaslanır.

```
<xs:simpleType name="Ondalığ">
  <xs:restriction base="xs:decimal"/> </
  xs:simpleType>
```

```
<xs:simpleType name="Real">
  <xs:restriction base="xs:decimal"/> </
  xs:simpleType>
```

QEYD ISO/TS 19103 Ondalığ və Real arasındakı konseptual fərqi müəyyən etmir.

C.5.2.1.5 Vektor

Vektor nömrələr ardıcılığı kimi müəyyən edilir. Siyahı konstruksiyası onluq qiymətlərin siyahısını müəyyən edir.

```
<xs:simpleType name="Vektor">
  <xs:list itemType="xs:decimal"/> </
  xs:simpleType>
```

C.5.2.1.6 Xarakter

Xarakter yalnız bir simvol ehtiva edən XML Sxema sətri kimi təqdim olunur.

```
<xs:simpleType name="Xarakter">
  <xs:restriction base="xs:string">
    <xs:length value="1" fixed="true"/>
  </xs:restriction>
</xs:simpleType>
```

C.5.2.1.7 CharacterString

CharacterString istənilən ISO/IEC 10646 sətirini təmsil edə bilən XML Sxem sətirinə əsaslanır.

```
<xs:simpleType name="CharacterString">  
  <xs:restriction base="xs:string"/>  
</xs:simpleType>
```

QEYD Dilin identifikasiyası üçün C.5.6-ya baxın.

C.5.2.1.8 Tarix

Tarix, ISO 8601-ə uyğun olaraq kanonik kodlaşdırmaya malik XML Sxema tarixinə əsaslanır.

```
<xs:simpleType name="Tarix">  
  <xs:restriction base="xs:date"/>  
</xs:simpleType>
```

C.5.2.1.9 Vaxt

Vaxt ISO 8601-ə uyğun olaraq kanonik kodlaşdırmaya malik olan XML Sxema vaxtına əsaslanır.

```
<xs:simpleType name="Time">  
  <xs:restriction base="xs:time"/>  
</xs:simpleType>
```

C.5.2.1.10 Tarix Saat

DateTime, ISO 8601-ə uyğun olaraq kanonik kodlaşdırmaya malik olan XML Sxema dateTime-ə əsaslanır.

```
<xs:simpleType name="DateTime">  
  <xs:restriction base="xs:dateTime"/> </  
  xs:simpleType>
```

C.5.2.1.11 Boolean

Boolean XML Sxema boolean-a əsaslanır. Dəyərlər məntiqi yalanı təmsil edən "0" və ya "yanlış", məntiqi doğru olan isə "1" və "doğrudur".

```
<xs:simpleType name="Boolean">  
  <xs:restriction base="xs:boolean"/> </  
  xs:simpleType>
```

C.5.2.1.12 Məntiqi

Məntiq üç dəyəri müəyyən edir: doğru, bəlkə və yalan. O, XML Sxema boolean və bəlkə dəyərini təmsil edən iki sadalanan dəyər, "0.5" və "bəlkə" arasında birləşmə kimi təmsil olunur.

```
<xs:simpleType name="Məntiqi">  
  <xs:union>  
    <xs:simpleType>  
      <xs:restriction base="xs:boolean"/> </  
    xs:simpleType>
```

```

<xs:simpleType>
  <xs:məhdudiyət bazası="xs:NMTOKEN">
    <xs:enumeration value="bəlkə"/>
    <xs:enumeration value="0.5"/> </
  xs:restriction>
</xs:simpleType>
</xs:union>
</xs:simpleType>

```

C.5.2.1.13 Ehtimal

Ehtimal 0 ilə 1.0 arasındakı onluq ədəddir.

```

<xs:simpleType name="Ehtimal">
  <xs:restriction base="xs:decimal">
    <xs:minInclusive value="0.0"/>
    <xs:maxInclusive value="1.0"/> </
  xs:restriction>
</xs:simpleType>

```

C.5.2.1.14 İkili

XML Şeması iki ikili məlumat növlərini müəyyən edir base64Binary və hexBinary. UML-də binar məlumat növləri kimi istifadə ediləcək iki növ müəyyən edilmişdir: BinaryBase64 və BinaryHex. Binary adlı xüsusi seçim növü müəyyən edilmişdir ki, o, ya BinaryBase64, ya da BinaryHex-i ehtiva edir.

```

<xs:simpleType name="BinaryBase64">
  <xs:restriction base="xs:base64Binary"/> </
  xs:simpleType>

  xs:simpleType name="BinaryHex">
    <xs:restriction base="xs:hexBinary"/> </
    xs:simpleType>

  <xs:complexType name="İkili">
    <xs:seçim>
      <xs:element name="BinaryBase64" type="BinaryBase64"/>
      <xs:element name="BinaryHex" type="BinaryHex"/> </
    xs:choice>
  </xs:complexType>

```

QEYD Binar məlumat növü ISO/TS 19103-də müəyyən edilməyib.

C.5.2.1.15 UnlimitedInteger

UnlimitedInteger 0-dan sonsuzluğa qədər dəyər sahəsi olan əsas növdür. "*" simvolu sonsuz dəyəri təmsil etmək üçün müəyyən edilmişdir.

```

<xs:simpleType name="UnlimitedInteger">
  <xs:union>
    <xs:simpleType>
      <xs:restriction base="xs:nonNegativeInteger"/> </
    xs:simpleType>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="*" />

```

```
</xs:məhdudiyət>
</xs:simpleType>
</xs:union>
</xs:simpleType>
```

C.5.2.2 <<Məlumat Tipi>>

C.5.2.2.1 Ümumi qayda

<<DataType>> stereotipi olan sinif XML Sxemində kompleks Tip tərifinə çevrilməlidir. Atributlar və mümkün assosiasiyalar C.5.3-ə uyğun olaraq ardıcıl konstruksiyada XML atributları və ya yerli XML elementləri kimi elan edilməlidir. Buna görə də, əmlak elementlərinin sırası kompleks tip tərifində verilmişdir.

ISO/TS 19103-də müəyyən edilmiş məlumat növləri C.5.2.2.2 və C.5.2.2.3-də çevrilir.

QEYD Müxtəlif növlər ISO/TS 19103-də aydın şəkildə müəyyən edilməmişdir və strukturlaşdırılmış verilənlər olan bütün siniflər də deyil. <<DataType>> stereotipindən istifadə etməklə müəyyən edilən növlər. Buna görə də, C.5.2.2.2 və C.5.2.2.3-dəki bəyannamələr ISO/TS 19103-də verilənlər tipi təriflərini şərh edir.

C.5.2.2.2 Çoxluq

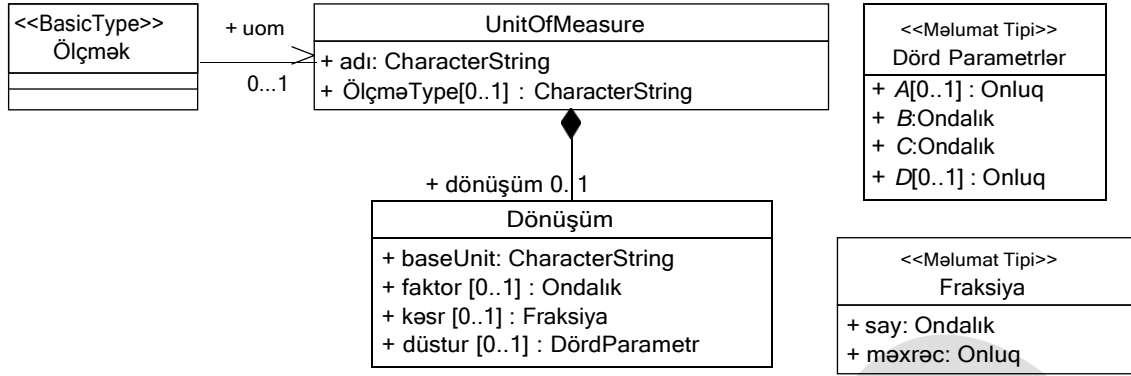
Çoxluq sinfi ISO/TS 19103-də aşağıdan yuxarıya çoxluq diapazonu kimi müəyyən edilir. Burada o, <<DataType>> kimi şərh edilir və aşağıdakı kimi müəyyən edilir.

```
<xs:complexType name="Çoxluq">
  <xs:ardıcılıq>
    <xs:element name="range" type="MultiplicityRange" minOccurs="1" maxOccurs="unbounded"/> </
  xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="MultiplicityRange">
  <xs:ardıcılıq>
    <xs:element name="lower" type="xs:nonNegativeInteger"/>
    <xs:element name="yuxarı" type="UnlimitedInteger"/> </
  xs:sequence>
</xs:complexType>
```

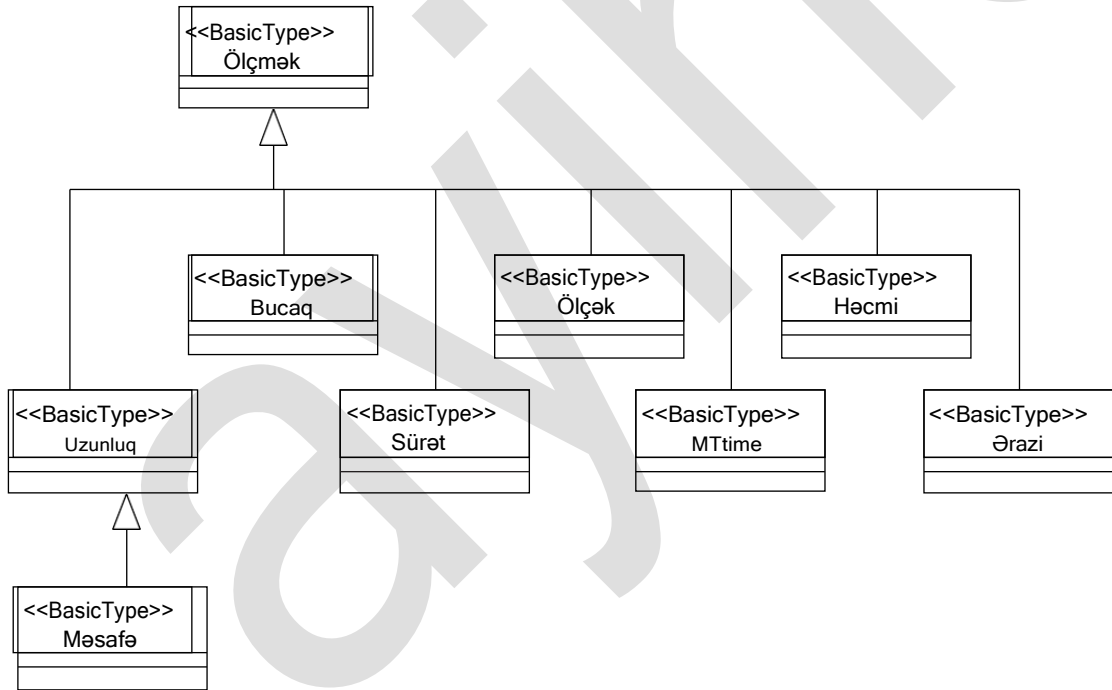
C.5.2.2.3 Ölçü vahidləri

ISO/TS 19103-də müəyyən edilmiş ölçü növlərinin vahidləri yerli və beynəlxalq ölçmə sistemlərinin tərifləri kimi istifadə üçün nəzərdə tutulub. Sonra kəmiyyətlər, ehtimal ki, ölçü vahidləri lüğətində dərc edilmiş vahid sistemine uyğun olaraq ölçülür. Təəssüf ki, modelləri başa düşmək çətindir, heç bir nümunə verilmir və təsvir qeyri-müəyyəndir. POSC (Petroleum Open Standards Consortium) Ölçü Vahidləri Tövsiyələri ISO/TS 19103-ə alternativdir. Spesifikasiyada bəzi əsas ideyaları təsvir edən UML diaqramı Şəkil C.6-da göstərilmişdir. UnitOfMeasure nümunəsi ölçü sistemini müəyyənləşdirir, ona ad verir və lazım gələrsə, ümumi düsturdan istifadə edərək kəmiyyətləri əsas vahidə necə çevirmək barədə məlumat verir. $Y = (A-BX)/(C-DX)$, harada X çevriləcək vahidin qiymətidir və Y baza blokundadır. Əgər $A-D=0$ və $C=1$, onda B çevrilmə faktoruna çevrilir. Əgər $A-D=0$, çevrilmə əmsalı kəsrlə təsvir edilir. Əks təqdirdə, dörd parametrlə təsvir olunur. Ölçmə nümunəsi ölçü vahidinə istinadla onluq dəyərdir.



Şəkil C.6 — Ölçü vahidləri

POSC Tövsiyə Vahidlərində müəyyən edilmiş XML Sxem təriflərindən istifadə etmək tövsiyə olunur.



Şəkil C.7 — Ölçü növləri

ISO/TS 19103 (Uzunluq, Bucaq, Sürət, Ölçək, MTtime, Sahə və Həcm) ilə müəyyən edilmiş ölçü növləri aşağıdakı kimi müəyyən edilməlidir.

```
<xs:complexType name="Measure">
  <xs:simpleContent>
    <xs:extension base="Ondalıq">
      <xs:atribut adı="uom" type="URI"/> </
    <xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

```

<xs:complexType name="Uzunluq">
  <xs:simpleContent>
    <xs:extension base="Measure"/> </
    xs:simpleContent>
  </xs:complexType>

<xs:complexType name="Bucaq">
  <xs:simpleContent>
    <xs:extension base="Ondaliq"/> </
    xs:simpleContent>
  </xs:complexType>

xs:complexType name="Ölçək">
  <xs:simpleContent>
    <xs:extension base="Measure"/> </
    xs:simpleContent>
  </xs:complexType>

<xs:complexType name="Sahə">
  <xs:simpleContent>
    <xs:extension base="Measure"/> </
    xs:simpleContent>
  </xs:complexType>
<xs:complexType name="Velocity">
  <xs:simpleContent>
    <xs:extension base="Measure"/> </
    xs:simpleContent>
  </xs:complexType>

<xs:complexType name="MTime">
  <xs:simpleContent>
    <xs:extension base="Measure"/> </
    xs:simpleContent>
  </xs:complexType>

<xs:complexType name="Məsafə">
  <xs:simpleContent>
    <xs:extension base="Measure"/> </
    xs:simpleContent>
  </xs:complexType>

<xs:complexType name="Həcmi">
  <xs:simpleContent>
    <xs:extension base="Measure"/> </
    xs:simpleContent>
  </xs:complexType>

```

C.5.2.3 <<Saylama>>

<<enumeration>> kimi stereotipləşdirilmiş sinif mətn sətirini bir sıra sadalanan dəyərlərlə məhdudlaşdıran sadə tipə çevrilməlidir.

NÜMUNƏ Nümunə, sətirə əsaslanan və dəyərini "-" ilə məhdudlaşdıran işarə nömrələməsidir. "müsbət", "-." və ya "mənfi".

<<Sadalama>> İmza
müsbət + mənfi -

Şəkil C.8 — <<Sadalama>> nümunəsi

```
<xs:simpleType name="İmza">
  <xs:restriction base="xs:string">
    <xs:enumeration value="+"/>
    <xs:enumeration value="-"/>
    <xs:enumeration value="pozitiv"/>
    <xs:enumeration value="negative"/>
  </xs:restriction>
</xs:simpleType>
```

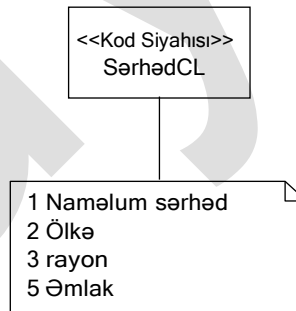
C.5.2.4 <<Kod Siyahısı>>

<<codelist>> kimi stereotipləşdirilmiş sinif çıxış sxemində çevrilməməlidir, əksinə kod siyahısında müəyyən edilmiş kodu və dəyər cütlərini saxlayan lüğətə uyğunlaşdırıla bilər. Lüğət ictimaiyyətə açıqlanmalı və onun veb ünvanı URI kimi verilməlidir.

Kod siyahısı tipli atribut sətir dəyəri kimi kodlaşdırılmalıdır.

Həmçinin CodeListExtraction sinfi üçün Şəkil C.22-yə və kod siyahılarının təsviri üçün Şəkil C.27-yə baxın.

NÜMUNƏ Şəkil C.9 XML-də lüğətə uyğunlaşdırılmış BorderCL adlı kod siyahısını göstərir. Bunun əvəzinə qeyd edin BorderCL-in atributları kimi kod-dəyər cütlərini sıralayan şərh qutusu istifadə olunur.



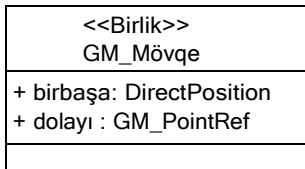
Şəkil C.9 — <<Kod Siyahısı>> nümunəsi

```
<codelist name="BorderCL">
  <codevalue code="1" value="Naməlum sərhəd"/>
  <codevalue code="2" value="Ölkə"/> <codevalue
  code="3" value="Ölkə"/> <codevalue code="5"
  value="Əmlak"/> </codelist>
```

C.5.2.5 <<Birlik>>

<<Birlik>> stereotipi olan sinif bir sıra atributları sadalayır və semantika ondan ibarətdir ki, atributlardan yalnız biri istənilən vaxt mövcud ola bilər. O, seçim konstruksiyasında elementlər kimi atributlarla mürəkkəb tip tərifinə çevrilməlidir.

NÜMUNƏ GM_Position birliyi ISO 19107-dən götürülmüşdür. seçim tikintisində yerli element bəyannamələri.



Şəkil C.10 — <<Birlik>> nümunəsi

```
<xs:complexType name="GM_Position">
  <xs:seçim>
    <xs:element name="birbaşa" type="DirectPosition"/>
    <xs:element name="dolay" type="GM_PointRef"/> </
  <xs:choice>
</xs:complexType>
```

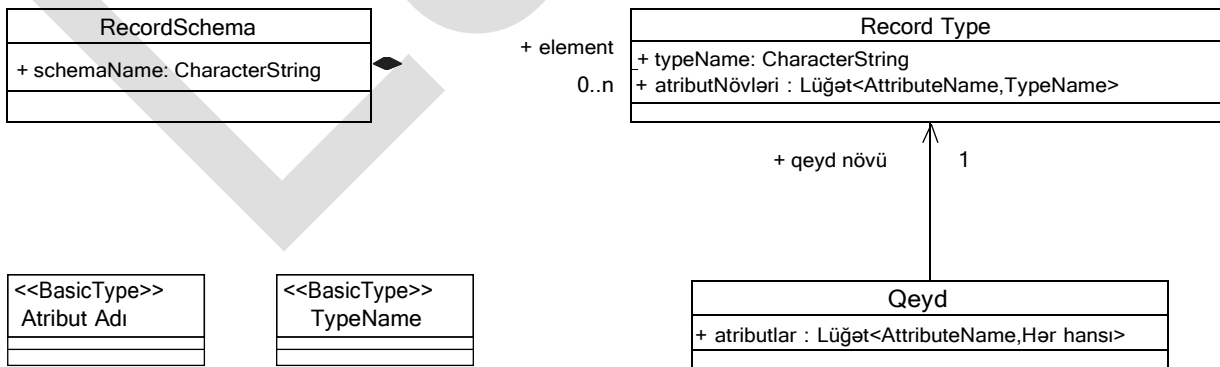
C.5.2.6 Obyekt növləri

C.5.2.6.1 Ümumi qayda

Stereotip və ya stereotipi olmayan sinif <<Tip>> siniflə eyni adlı mürəkkəb tip tərifinə uyğunlaşdırılmalıdır. Mürəkkəb tip tərifinə ya IM_Object-dən miras qalmış, ya da IM_ObjectIdentificaton atribut qrupuna istinad vasitəsilə miras qalan identifikasiya atributları daxil edilməlidir.

C.5.2.6.2 Qeyd növləri

Bir neçə qeyd növü ISO/TS 19103-də müəyyən edilmişdir. Onlar Şəkil C.11-də yenidən qurulmuş və şərh edilmişdir. AttributeName və TypeName CharacterString əsasında əsas növlər kimi modeləşdirilmişdir.



Şəkil C.11 — Qeyd növləri

```

<xs:complexType name="RecordSchema">
  <xs:ardıcılıq>
    <xs:element name="schemaName" type="CharacterString"/>
    <xs:element name="element" type="RecordType" minOccurs="0" maxOccurs="unbounded"/> </
    xs:sequence>
  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

<xs:complexType name="RecordType">
  <xs:ardıcılıq>
    <xs:element name="typeName" type="CharacterString"/>
    <xs:element name="attributeTypes" type="Dictionary_AttributeName_TypeName_"/> </
    xs:sequence>
  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

<xs:complexType name="Qeyd">
  <xs:ardıcılıq>
    <xs:element name="atributlar" type="Dictionary_AttributeName_Any_"/>
    <xs:element name="recordType" type="ref_RecordType"/> </xs:sequence>

  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

```

C.5.2.7 Bağlanmış şablon növü

Bağlanmış şablon növü, parametrlərin faktiki arqument dəyərlərinə bağlandığı bir növdür. ISO/TS 19103 beş müxtəlif şablon növünü müəyyən edir: Set<T>, Bag<T>, Sequence<T>, CircularSequence<T> və Dictionary<K,V>. İlk dörd bir parametr, Lüğət isə iki parametr alır. Bu növlər adətən atribut bəyannamələrində bağlanır; Şəkil C.11-də müəyyən edilmiş RecordType-in "attributeTypes" atributuna baxın.

- Bağlanmış şablon növü şablon növünə uyğun gələn mürəkkəb tip tərifinə çevrilməlidir. Növ bəyannaməsi adlandırılmalıdır. Ad, şablon adını alt xətt “_” simvolları ilə ayrılmış arqument adları ilə birləşdirməklə tikilə bilər. “<-dən kiçik, vergül “,” və “>-dən böyük simvollar adda istifadə edilə bilməz.
- Bağlanmış Dəst, Çanta, Ardıcılıq və ya Dairəvi Ardıcılıq şablon növü vahid parametr növünə uyğun olaraq adlandırılan və tiplənmiş bir elementi ehtiva edən qeyri-məhdud çoxluqlu ardıcılıq konstruksiyasından ibarət mürəkkəb tip tərifinə uyğunlaşdırılmalıdır. Parametr növü əsas tipdirsə, XML sxeminin siyahı konstruksiyasından istifadə edərək, əvəzinə sadə tip tərifindən istifadə edilə bilər.
- Lüğətin bağlanmış şablon növü, iki parametrə uyğun olaraq adlandırılan və yığılan iki elementdən ibarət qeyri-məhdud çoxluğun ardıcıl konstruksiyasından ibarət mürəkkəb tip tərifinə uyğunlaşdırılmalıdır.

NÜMUNƏ 1 ISO 19107-də müəyyən edilmiş birbaşa mövqe məlumat növü Şəkil C.12-də göstərilmişdir.

<<Məlumat Tipi>> DirectMövqe
+ koordinat : Ardıcılıq<Nömrə>

Şəkil C.12 — Məhdudlaşdırılmış şablon tipinin nümunəsi

Sequence<Number> bağlı şablon növünü müəyyən edir və ümumi qaydaya uyğun olaraq aşağıdakılarla əlaqələndirilir.

```
<xs:complexType name="Sequence_Number_">
  <xs:sequence maxOccurs="bounded">
    <xs:element name="Number" type="Number"/> </
  </xs:sequence>
</xs:complexType>
```

Argument növü əsas tip olduğundan, o, həmçinin aşağıdakılarla əlaqələndirilə bilər:

```
<xs:simpleType name="Sequence_Number_">
  <xs:list itemType="Number"/>
</xs:simpleType>
```

NÜMUNƏ 2 Şəkil C.11-də müəyyən edilmiş bağlanmış şablon növləri aşağıdakı kimi təsvir edilmişdir.

```
<xs:complexType name="Dictionary_AttributeName_Hər_">
  <xs:sequence maxOccurs="bounded">
    <xs:element name="AttributeName" type="CharacterString"/>
    <xs:element name="Hər hansı" type="xs:anyType"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="Dictionary_AttributeName_TypeName_">
  <xs:sequence maxOccurs="bounded">
    <xs:element name="AttributeName" type="CharacterString"/>
    <xs:element name="TypeName" type="CharacterString"/> </
  </xs:sequence>
</xs:complexType>
```

C.5.2.8 Vərəsəlik

C.5.2.8.1 General

UML-də varislik mexanizmi alt tipə öz supertiplərinin atributlarını və assosiasiyalarını miras almağa imkan verir. Tək irsiyyətdə bir növ yalnız bir supertiptən miras qala bilər, halbuki çoxlu mirasda bir növ birdən çox növdən miras qala bilər. UML həm tək, həm də çoxlu varisliyə imkan verir. XML Sxema yalnız tək irsiyyəti dəstəkləyir. Buna görə çoxlu mirasın simulyasiyası lazımdır.

Vərəsəlik də həyata keçirilməlidir

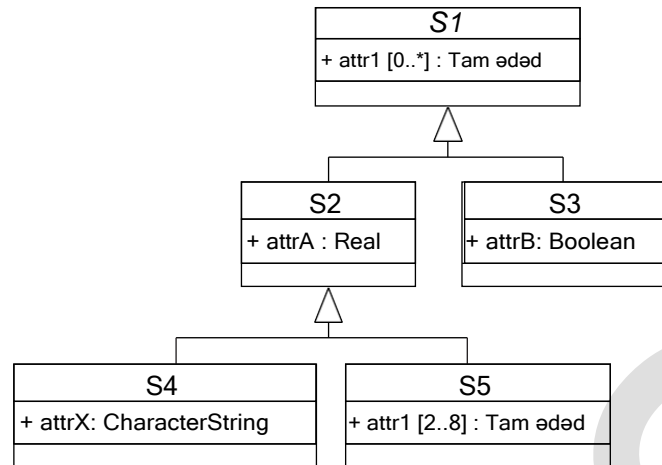
- adlanan XML Şeması genişləndirilməsi və ya məhdudlaşdırma mexanizmi ilə *tək miras*(C.5.2.8.2) və ya
- atributları və assosiasiyaları supertiplərdən çağırılan hədəf tipə köçürməklə *çoxlu miras* (C.5.2.8.3).

Çoxsaylı vərəsəlik zamanı atributlar və assosiasiyalar hədəf tipə kopyalanmalıdır.

C.5.2.8.2 Tək varislik

Ümumi qayda kompleks növlər üçün XML Şemasının genişləndirilməsi mexanizmindən istifadə etməkdir. Lakin atribut və ya assosiasiya yenidən müəyyən edilərsə, məhdudlaşdırma mexanizmindən istifadə edilməlidir.

NÜMUNƏ Şəkil C.13 S2 və S3 alt tipləri olan S1 supertipini göstərir. S1 abstrakt sinifdir. S4 və S5-dir S2-nin alt növləri. Qeyd edək ki, S5 "attr1"i yenidən müəyyənləşdirir və buna görə də məhdudlaşdırma mexanizmindən istifadə etmək lazımdır.



Şəkil C.13 — Tək miras nümunəsi

```

<xs:complexType name="S1" abstract="true">
  <xs:complexContent>
    <xs:extension base="IM_Object">
      <xs:ardıcılıq>
        <xs:element name="attr1" type="Integer" minOccurs="0" maxOccurs="unbounded"/> </
        xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

<xs:complexType name="S2">
  <xs:complexContent>
    <xs:extension base="S1">
      <xs:ardıcılıq>
        <xs:element name="attrA" type="Real"/> </
        xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

<xs:complexType name="S3">
  <xs:complexContent>
    <xs:extension base="S1">
      <xs:ardıcılıq>
        <xs:element name="attrB" type="Boolean"/> </
        xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

<xs:complexType name="S4">
  <xs:complexContent>
    <xs:extension base="S2">
      <xs:ardıcılıq>
        <xs:element name="attrX" type="CharacterString"/> </
        xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  
```

```

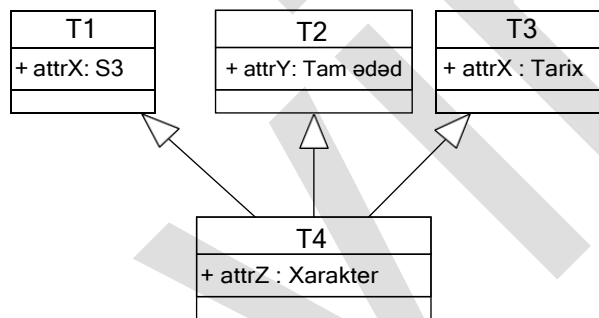
<xs:complexType name="S5">
  <xs:complexContent>
    <xs:məhdudiyət bazası="S2">
      <xs:ardıcılıq>
        <xs:element name="attr1" type="Tam" minOccurs="2" maxOccurs="8"/>
        <xs:element name="attrA" type="Real"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

```

C.5.2.8.3 Çoxsaylı vərəsəlik

Atributların/assosiasiyaların surətinin çıxarılması proseduru sol supertipdən başlamaq və onun atributlarını və assosiasiyalarını kopyalamaq, sonra ən sağdakı supertipə çatana qədər növbəti supertiplə sağa davam etməkdir. Alt növün atributları sonuncu əlavə edilir. Əgər supertip və ya alt tip əvvəllər kopyalanmış eyni adlı atribut və ya assosiasiyaları müəyyən edərsə, münaqişə baş verir. Ad ziddiyyəti olduqda, sonuncu atribut və ya assosiasiya üstünlük təşkil etməli və əvvəllər kopyalanana əvəz etməlidir.

NÜMUNƏ Şəkil C.14 dörd növü müəyyən edir: T1, T2, T3 və T4. T4 T1, T2 və T3-ün alt növüdür.



Şəkil C.14 — Çoxlu miras nümunəsi

```

<xs:complexType name="T1">
  <xs:ardıcılıq>
    <xs:element name="attrX" type="S3"/> </
  </xs:sequence>
  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

<xs:complexType name="T2">
  <xs:ardıcılıq>
    <xs:element name="attrY" type="Real"/> </
  </xs:sequence>
  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

<xs:complexType name="T3">
  <xs:ardıcılıq>
    <xs:element name="attrX" type="Tarix"/> </
  </xs:sequence>
  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

```

```

<xs:complexType name="T4">
  <xs:ardıcılıq>
    <xs:element name="attrX" type="Tarix"/>
    <xs:element name="attrY" type="Integer"/>
    <xs:element name="attrZ" type="Xarakter"/> </
  <xs:sequence>
    <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

```

C.5.2.9 Əvəzetmə növləri

Bir atribut növü kimi supertipdən istifadə o deməkdir ki, atribut nümunəsi supertipin irsiyyət iyerarxiyası ilə müəyyən edilmiş konkret alt tiplərdən birinə aid ola bilər. XML sxemi bu dinamik tip mexanizmini birbaşa dəstəkləmir.

Üç alternativ yanaşma istifadə edilə bilər.

- Supertipə uyğun tipli standart element bəyannaməsini elan edin. Mübadilə faylında tələb olunan növü göstərmək üçün `xsi:type` atributundan istifadə edilməlidir. Misal 1-ə baxın.
- Qlobal elementləri supertipin irsiyyət iyerarxiyasına uyğun gələn əvəzetmə qrupu ilə müəyyənləşdirin. Qlobal elementə element bəyannaməsində istinad edilməlidir. Misal 2-ə baxın.
- Supertipin varislik iyerarxiyasındakı konkret tiplərin hər biri üçün element bəyannamələrinin seçimini ehtiva edən hər bir supertip üçün seçim qruplarını müəyyənləşdirin. Seçim qrupuna element bəyannaməsi daxilində istinad edilməlidir. Misal 3-ə baxın.

Əgər C.5.2.8.3-də təsvir olunan surətin çıxarılması mexanizmi istifadə olunursa, yalnız c) yanaşmasından istifadə edilməlidir.

NÜMUNƏ 1 a yanaşması): Ex2 sinfi S1 tipli atributu müəyyən edir, Şəkil C.15-ə baxın. S1 abstrakt supertipdir Şəkil C.13-də müəyyən edilmiş irsiyyət iyerarxiyası ilə.

Məsələn 2
+ istifadə 1: S1

Şəkil C.15 — Supertipin atributunun nümunəsi

```

<xs:complexType name="Ex2">
  <xs:ardıcılıq>
    <xs:element name="use1" type="S1"/> </
  <xs:sequence>
</xs:complexType>

```

Mübadilə faylında `xsi:type` istifadə edin:

```

<Ex1>
  <istifadə1 xsi:type="S3">
    <attr1>42</attr1><attr1>43</attr1><attr1>44</attr1>
    <attrB>doğru</attrB>
  </use1>
</Ex1>

```

NÜMUNƏ 2 Yanaşma b): Qlobal əvəzetmə elementlərinin istifadəsi.

```
<xs:element name="S1" type="S1" abstract="true"/> <xs:element
name="S2" type="S2" substitutionGroup="S1"/> <xs:element
name="S3" type="S3" substitutionGroup="S1"/> <xs:element
name="S4" type="S4" substitutionGroup="S2"/> <xs:element
name="S5" type="S5" substitutionGroup="S2"/>
```

```
<xs:complexType name="Ex2">
  <xs:ardıcılıq>
    <xs:element name="istifadə1">
      <xs:complexType>
        <xs:ardıcılıq>
          <xs:element ref="S1"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

```
<Ex1>
  <istifadə1>
    <S3>
      <attr1>42</attr1><attr1>43</attr1><attr1>44</attr1>
      <attrB>doğru</attrB>
    </S3>
  </use1>
</Ex1>
```

NÜMUNƏ 3 Yanaşma c): Seçim qruplarının istifadəsi.

```
<xs:grup adı="S1">
  <xs:seçim>
    <xs:element name="S2" type="S2"/>
    <xs:element name="S3" type="S3"/>
    <xs:element name="S4" type="S4"/>
    <xs:element adı="S5" type="S5"/> </
  xs:choice>
</xs:grup>
```

```
<xs:complexType name="Ex2">
  <xs:ardıcılıq>
    <xs:element name="istifadə1">
      <xs:complexType>
        <xs:grup ref="S1"/> </
      xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

Nümunə 2-ci Nümunədəki nümunə ilə tam olaraq eyni olur. Bu yanaşmanın üstünlüyü ondan ibarətdir ki, qlobal elementlər yoxdur.

C.5.2.10 <<Abstrakt>> və ya mücərrəd sinif

Mücərrəd sinif, sürəti azaltma mexanizmindən istifadə edilərsə, kompleks Tip bəyannaməsi kimi təqdim edilə bilməz. Tək irsiyyətdən istifadə edilərsə, mücərrəd sinif C.5.2.6-ya uyğun olaraq "mücərrəd" atributunun doğru olaraq təyin edilmiş kompleks Tip tərifinə çevrilməlidir.

C.5.2.11 <<ExternalType>>

Stereotiplənmiş bütün siniflər <<ExternalType>> ya XML Sxema tipli anyURI-ni məhdudlaşdıran sadə tipə, ya da NOTATION tipinə uyğunlaşdırılmalıdır.

NÜMUNƏ

```
<xs:simpleType name="Modis">
  <xs:restriction base="xs:anyURI"/> </
  xs:simpleType>
```

C.5.3 Mülkiyyət elementi bəyannamələri

C.5.3.1 Atribut

Atribut bir sinfin xarakteristikasını təyin edir. Atribut a çevrilməlidirelement bəyannaməsivə ya bir atribut bəyannaməsinin mürəkkəb tip bəyannaməsində. Anelement bəyannaməstandart qaydadır. Əgər atribut sıfır və ya bir çoxluğu olan əsas tiplidirsə, o, a çevrilə bilərətribut bəyannaməsi. Bütün digər atributlar a çevrilməlidirelement bəyannaməsi.

Standart qayda bütün atributları və törəmə atributları çevirməkdir.

Anatribut bəyannaməsiadı və əsas məlumat növü ilə atribut müəyyən edir. Atributun çoxluğu Cədvəl C.4-ə uyğun olmalıdır. XML Şemasında atribut bəyannaməsinin standart çoxluğu istəyə bağlıdır.

Cədvəl C.4 — Atributlar üçün çoxluğun xəritələşdirilməsi

UML	Könüllü	Lazimi atribut bəyannaməsi
1 (defolt)	yalan	isteğe bağlı = "yanlış"
0..1	doğru (defolt)	—

Anelement bəyannaməsiadı və növü olan elementi müəyyən edir. Çoxluq Cədvəl C.5-ə uyğun olmalıdır. Standart dəyərlər cədvəldə göstərilmişdir və onların elan edilməsinə ehtiyac yoxdur. Alternativ olaraq həm minimum, həm də maksimum dəyərlər verilə bilər.

Cədvəl C.5 — Məzmun elementləri üçün çoxluq xəritəsi

UML	min baş verir	maxOccurs	Lazimi element bəyannaməsi
1 (defolt)	1 (defolt)	1 (defolt)	—
0..1	0	1 (defolt)	minOccurs="0"
0..*	0	sərhədsiz	minOccurs="0" maxOccurs="məhdudiyetsiz"
1..*	1 (defolt)	sərhədsiz	maxOccurs = "məhdudiyetsiz"
2..8	2	8	minOccurs="2" maxOccurs="8"

NÜMUNƏ Məlumat növü Nümunə UML-də elan edilmişdir. Onun üç atributu var "başlıq", "nömrə" və "altMisal", bax Şəkil C.16. Həm "başlıq", həm də "nömrə" atributları atribut bəyannaməsinə çevrilə bilər, halbuki "altNümunə" mürəkkəb tiplidir və onun element bəyannaməsinə uyğunlaşdırılması zəruridir.

<<Məlumat Tipi>> Misal
+ başlıq: CharacterString + sayı [0..1] : Tam ədəd + altMisal[0..*] : Nümunə

Şəkil C.16 — Nümunə atribut

Standart element bəyannaməsi aşağıdakıları verir:

```
<xs:complexType name="Nümunə" >
  <xs:ardıcillıq>
    <xs:element name="title" type="CharacterString"/>
    <xs:element name="number" type="Tam" minOccurs="0"/>
    <xs:element name="subExample" type="Nümunə" minOccurs="0" maxOccurs="məhdudiyətsiz"/> </
  <xs:sequence>
</xs:complexType>
```

Burada "başlıq" və "nömrə" atribut elanı qaydasına uyğun olaraq çevrilir.

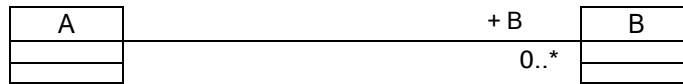
```
<xs:complexType name="Nümunə" >
  <xs:ardıcillıq>
    <xs:element name="subExample" type="Nümunə" minOccurs="0" maxOccurs="məhdudiyətsiz"/> </
  <xs:sequence>
    <xs:atribut adı="title" type="CharacterString" isteğe bağlı="false"/>
    <xs:atribut adı="nömrə" type="Tam ədəd"/>
</xs:complexType>
```

C.5.3.2 Assosiasiya

Assosiasiya iki sinif arasında ümumi əlaqəni müəyyən edir. Aşağıda siniflərdən biri mənbə sinif, digəri isə hədəf sinif adlanır. Mənbə obyektləri hədəf obyektlərə istinadları saxlayır və əksinə.

- Əgər assosiasiya naviqasiya edilə biləndirsə və hədəf sinif rol adı ilə müəyyən edilirsə, mənbə sinfinə uyğun gələn kompleks tip element bəyannaməsini ehtiva etməlidir. Elementin adı hədəf sinfi müəyyən edən rol adı, növü isə hər ikisi olmalıdır *IM_Object Reference*, və ya müəyyən edilmiş atributlara əsaslanan və ya olan bir növ *IM_Object Reference*. Cədvəl C.5-ə uyğun olaraq element bəyannaməsi çoxalmalıdır.
- Hədəf sinfinə uyğun gələn kompleks tip, əgər assosiasiya naviqasiya edilə biləndirsə və mənbə sinfi rol adı ilə müəyyən edilirsə, element bəyannaməsini ehtiva etməlidir. Elementin adı mənbə sinfini müəyyən edən rol adı, növü isə hər ikisi olmalıdır *IM_Object Reference*, və ya müəyyən edilmiş atributlara əsaslanan və ya olan bir növ *IM_Object Reference*. Cədvəl C.5-ə uyğun olaraq element bəyannaməsi çoxalmalıdır.

NÜMUNƏ A və B sinifləri arasında assosiasiya müəyyən edilmişdir, bax Şəkil C.17. B haqqında yalnız birindən yalnız A bilir rolun adı müəyyən edilir.



Şəkil C.17 — Assosiasiya nümunəsi

XML Sxema bəyannamələri aşağıdakı kimidir:

```
<xs:complexType name="A">
  <xs:ardıcılıq>
    <xs:element adı="theB" type="ref_B" minOccurs="0" maxOccurs="məhdudiyətsiz"/> </
    xs:sequence>>
</xs:complexType>

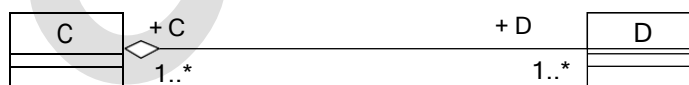
<xs:complexType name="ref_B">
  <xs:attributeGroup ref="IM_ObjectReference"/> </
  xs:complexType>
```

C.5.3.3 Ümumiləşdirmə

Aqreqasiya məcmu (bütöv) və tərkib hissəsi arasında zəif tam hissə əlaqəsini müəyyən edir. Sahiblik zəifdir ki, hissələr eyni anda birdən çox məcmuənin üzvü ola bilər. Beləliklə, bir hissə obyektini birdən çox ümumi obyekt tərəfindən paylaşa bilər. Beləliklə, məcmu ümumi olaraq yalnız onun hissələrinə istinad edə bilər, lakin tam mülkiyyət olduqda müvafiq hissələri ehtiva edə bilər.

- Ümumi sinifə uyğun olan kompleks tip element elanını ehtiva etməlidir, burada adı hissə sinfini müəyyən edən rol adına uyğun gəlir. Bu elementin çoxluğu Cədvəl C.5-ə uyğun olmalıdır. Elementin növü əsaslanmalıdır *IM_Object Reference* və hissə sinfinə uyğun gələn növün sıfır və ya bir elementini ehtiva edə bilər.
- Əgər assosiasiya naviqasiya edilə biləndirsə və hədəf sinif rol adı ilə müəyyən edilirsə, hissə sinfinə uyğun olan kompleks tip element bəyannaməsini ehtiva etməlidir. Elementin adı məcmu sinifi müəyyən edən rol adına uyğun gəlməlidir və növü əsaslanmalıdır *IM_Object Reference*. Cədvəl C.5-ə uyğun olaraq element bəyannaməsi çoxalmalıdır.

NÜMUNƏ C aqreqatı ilə D hissəsi arasında birləşmə müəyyən edilmişdir, Şəkil C.18-ə baxın. C D-ni müəyyən edir rol adı "theD" və D "theC" rol adı ilə C-ni müəyyən edir.



Şəkil C.18 — Toplama nümunəsi

```
<xs:complexType name="C">
  <xs:ardıcılıq>
    <xs:element name="theD" minOccurs="1" maxOccurs="məhdudiyətsiz">
      <xs:complexType>
        <xs:ardıcılıq>
          <xs:element name="D" type="D" minOccurs="0" maxOccurs="1"/>
        </xs:sequence>
        <xs:attributeGroup ref="IM_ObjectReference"/> </
        xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
```

```
<xs:complexType name="D">
  <xs:ardıcılıq>
    <xs:element name="theC" type="IM_ObjectReference" minOccurs="1" maxOccurs="məhdudiyətsiz"/> </
    xs:sequence>
</xs:complexType>
```

C.5.3.4 Tərkibi

Kompozisiya kompozit (bütöv) və tərkib hissəsi arasında güclü tam hissə əlaqəsini müəyyən edir. Mülkiyyət güclüdür ki, bir hissə tam olaraq bir kompozit obyektin üzvü ola bilər. Beləliklə, kompozit öz müvafiq hissələrini ehtiva etməlidir.

- Mürəkkəb sinfə uyğun olan kompleks tip element elanını ehtiva etməlidir, burada ad hissə sinfini müəyyən edən rol adına uyğundur və tip hissə sinfinin növünə uyğundur. Cədvəl C.5-ə uyğun olaraq element bəyannaməsi çoxalmalıdır.
- Hissə sinfinə uyğun olan kompleks tip heç bir element bəyannaməsini ehtiva etməməlidir, hətta rol adı kompozit sinfi müəyyən etsə belə. Bu, gizlidir, çünki bir hissə həmişə kompozit sinifdə yerləşir.

NÜMUNƏ E və F sinifləri arasında kompozisiya müəyyən edilmişdir, Şəkil C.19-a baxın. E, F-ni hədəf rol adı ilə müəyyən edir "theF".



Şəkil C.19 — Nümunə kompozisiya

```
<xs:complexType name="E">
  <xs:ardıcılıq>
    <xs:element adı="theE" type="F" minOccurs="2" maxOccurs="8"/> </
    xs:sequence>
</xs:complexType>
```

C.5.4 Element bəyannamələri

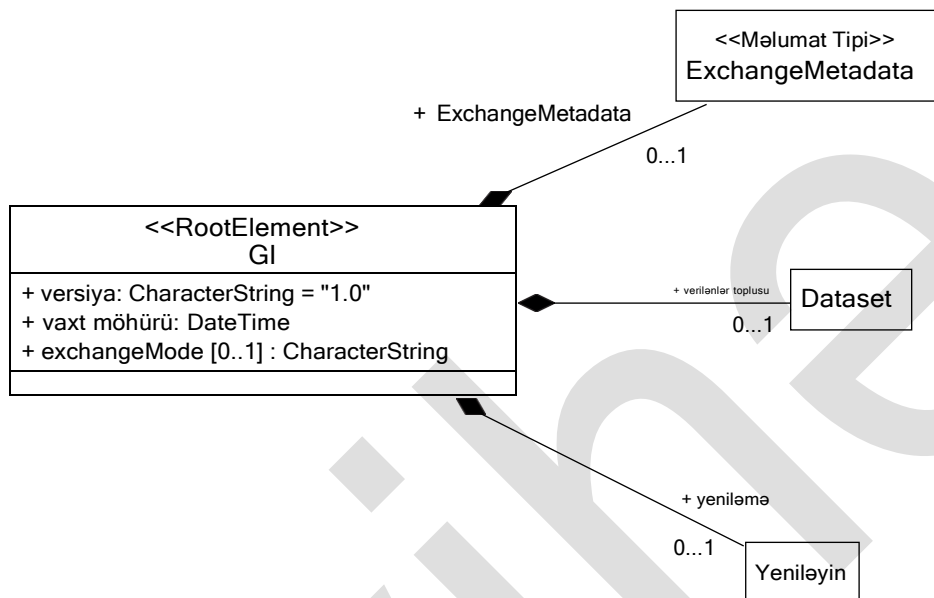
C.5.4.1 Sənədin strukturu

GI elementi mübadilə faylının kök elementi olmalıdır. O, üç elementdən ibarətdir: exchangeMetadata, dataset və update; Şəkil C.20-yə baxın. Verilənlər toplusu və yeniləmə elementləri identikliyə malik ola bilər, mübadilə məlumatları isə belə olmamalıdır.

GI elementinin atributları aşağıdakılardır:

- versiya: CharacterString = "1.0";
- vaxt möhürü: DateTime;
- exchangeMode [0..1] : CharacterString.

The **versiya** atributun "1.0" olaraq təyin edilməsi tələb olunur. Bu, mübadilə faylının bu Beynəlxalq Standartın bu versiyasına uyğun olduğunu göstərir. Bu Beynəlxalq Standartın yenidən işlənmiş versiyalarında onlarla əlaqəli başqa nömrə olacaq. The **vaxt damgası** atribut məlumatın kodlaşdırıldığı tarix və vaxtı göstərir. Yeni mübadilə faylı istehsal olunduqda. The **mübadilə rejimi** atribut istifadəçi tərəfindən müəyyən edilir və onun dəyəri mübadilə faylının kontekstini və ya rejimini göstərə bilər.



Şəkil C.20 — Sənədin strukturu

GI elementinin bəyannaməsi aşağıdakı kimidir.

```

<xs:element adı="GI">
  <xs:complexType>
    <xs:ardıcılıq>
      <xs:element name="exchangeMetadata" type="ExchangeMetadata" minOccurs="0"/>
      <xs:element name="dataset" type="Dataset" minOccurs="0"/>
      <xs:element name="yeniləmə" növü="Yeniləmə" minOccurs="0"/> </
    <xs:sequence>
      <xs:atribut adı="versiya" type="CharacterString" use="tələb olunur" fixed="1.0"/>
      <xs:atribut name="timeStamp" type="TarixTime" use="tələb olunur"/> <xs: atribut
      adı="exchangeMode" type="CharacterString"/>
    </xs:complexType>
  </xs:element>
  
```

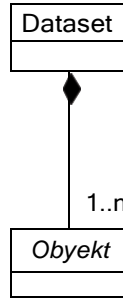
C.5.4.2 Verilənlər toplusu və obyekt elementləri

Verilənlər toplusu elan ediləcək obyekt elementləri adlanan obyektləri kodlayan bir və ya bir neçə elementdən ibarətdir.

- İdentifikasiya atributlarına malik bütün kompleks tip tərifləri obyekt elementi elanı üçün namizəddir. Bəziləri müstəqil obyektlər hesab olunmaya bilər və buna görə də müəyyən edilə bilməz.
- Elementin adı növün adı ilə eyni olmalıdır və ya bu tip üçün müəyyən edilmiş teq adı ola bilər. Obyekt elementi yerli və ya qlobal element kimi elan edilə bilər. Defolt yerli element kimidir.

- Bütün obyekt elementləri adlandırıla bilən seçim qrupunda qruplaşdırılmalıdır **Obyekt**. Beləliklə, seçim qrupu ya qlobal elan edilmiş elementlərə istinad edir, ya da obyekt elementlərini lokal olaraq elan edir. Bu seçim qrupu verilənlər bazasındakı hüquqi obyektləri məhdudlaşdırmaq üçün istifadə edilməlidir.

NÜMUNƏ Obyekt qrupu nümunənin obyekt elementlərini C.5.2.8.3-də müəyyən edir. Verilənlər dəsti növü an-a aiddir obyekt qrupunun qeyri-məhdud ardıcılığı (Şəkil C.21). Bu o deməkdir ki, verilənlər bazası yalnız T1, T2, T3 və T4 dörd elementdən ibarət ola bilər.



Şəkil C.21 — Dataset obyektləri ehtiva edir

```

<xs:complexType name="Dataset">
  <xs:sequence maxOccurs="bounded">
    <xs:group ref="Obyekt"/>
  </xs:sequence>
  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

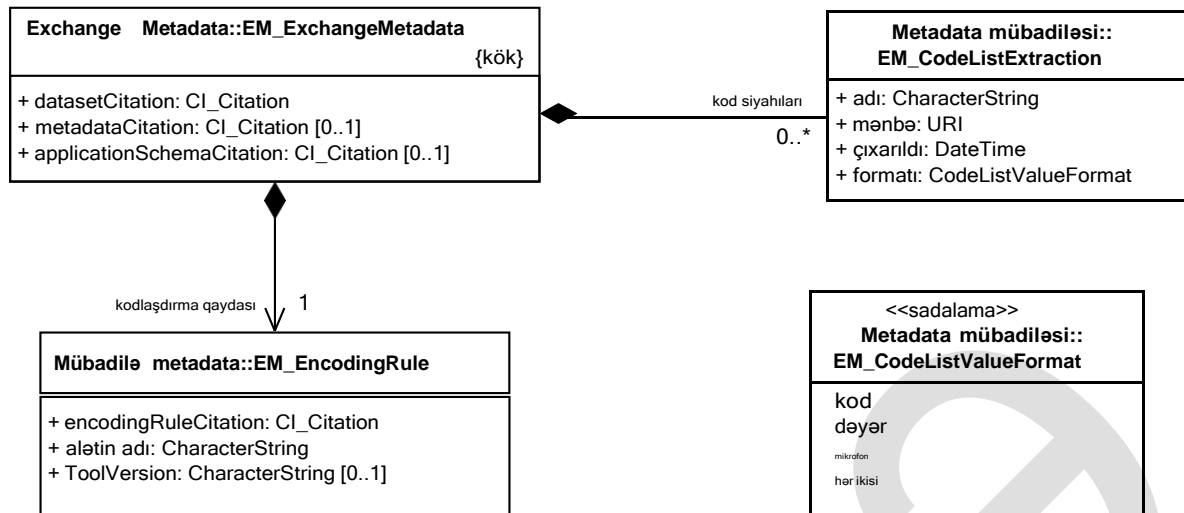
<xs:group name="Obyekt">
  <xs:seçim>
    <xs:element adı="T1" növü="T1"/>
    <xs:element adı="T2" növü="T2"/>
    <xs:element adı="T3" növü="T3"/>
    <xs:element adı="T4" type="T4"/>
  </xs:seçim>
</xs:group>

```

C.5.4.3 Metadata mübadiləsi

Mübadilə məlumatlarının növləri Şəkil C.22-də müəyyən edilmişdir. CI_Citation növü ISO 19115-dən idxal edilir və təkrar istifadə olunur.

ExchangeMetadata verilənlər toplusunu təsvir edən məlumatları ehtiva etməlidir. "DatasetCitation" atributu verilənlər toplusunun yaradıcısını təsvir edir. "MetadataCitation" atributu verilənlər toplusu üçün müvafiq metadətaya, "applicationSchemaCitation" istifadə olunan proqram sxeminə, "configFileCitation" isə istifadə edilən konfigurasiya faylına istinad edir. "EncodingRule" kompozisiyası verilənlər toplusunu yaratmaq üçün istifadə olunan kodlaşdırma qaydasını təsvir edir. Əgər verilənlər toplusunda kod siyahılarının atributları varsa, o, hansı kod siyahılarının istifadə edildiyini və onların etibarlılığını göstərməlidir. Bu, "codeLists" kompozisiyası ilə edilir.



Şəkil C.22 — Metadata mübadiləsi

EncodingRule verilənlər toplusunu yaratmaq üçün tətbiq olunan kodlaşdırma qaydası haqqında məlumatı ehtiva edir. “encodingRuleCitation” istifadə edilən kodlaşdırma qaydasını təsvir edir. “ToolName” verilənlər toplusunu yaratmaq üçün istifadə edilən kodlaşdırma xidməti alətinin adını, “toolVersion” isə onun versiya nömrəsini göstərir.

CodeListExtraction kod siyahısını “ad” və “mənbə” atributu ilə müəyyən edir. O, həmçinin bu xüsusi kod siyahısı üçün istifadə edilən kodların nə vaxt “çıxarıldığını” və bununla da kodların etibarlılığını göstərir. “Format” atributu, atributların kodu və ya kod siyahısının dəyər hissəsini istifadə etdiyini göstərir. Onlar həmçinin ikisini qarışdırma və ya həm kodu, həm də dəyəri istifadə edə bilərlər.

```

<xs:complexType name="ExchangeMetadata">
  <xs:ardıcılıq>
    <xs:element name="datasetCitation" type="CI_Citation"/>
    <xs:element name="metadataCitation" type="CI_Citation" minOccurs="0"/> <xs:element
    name="applicationSchemaCitation" type="CI_Citation" minOccurs="0"/> <xs:element
    name="configFileCitation" type="CI_Citation" minOccurs="0"/> <xs:element
    name="encoding" type="EncodingRule"/>
    <xs:element name="odelist" type="CodeListExtraction" minOccurs="0" maxOccurs="unbounded"/> </
    xs:sequence>
  </xs:complexType>

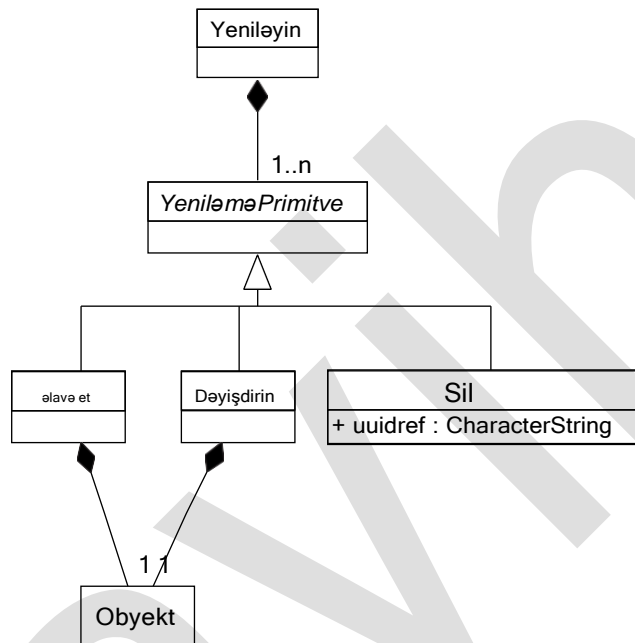
  <xs:complexType name="EncodingRule">
    <xs:ardıcılıq>
      <xs:element name="ruleCitation" type="CI_Citation"/>
      <xs:element name="toolName" type="CharacterString"/>
      <xs:element name="toolVersion" type="CharacterString"/> </
      xs:sequence>
    </xs:complexType>

    <xs:complexType name="CodeListExtraction">
      <xs:ardıcılıq>
        <xs:element name="name" type="CharacterString"/>
        <xs:element name="mənbə" type="URI"/>
        <xs:element name="timeStamp" type="DateTime"/>
        <xs:element name="format" type="CodeListValueFormat"/> </
        xs:sequence>
      </xs:complexType>
  
```

```
<xs:simpleType name="CodeListValueFormat">
  <xs:restriction base="CharacterString">
    <xs:enumeration value="code"/>
    <xs:enumeration value="value"/>
    <xs:enumeration value="mix"/> </
  </xs:restriction>
</xs:simpleType>
```

C.5.4.4 Yeniləmə

Üç yenilənmə primitivi Şəkil C.23-də müəyyən edilmişdir.



Şəkil C.23 — Primitivləri yeniləyin

```
<xs:complexType name="Əlavə et">
  <xs:ardıcılıq>
    <xs:group ref="Object"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="Modify">
  <xs:ardıcılıq>
    <xs:group ref="Object"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="Sil">
  <xs:attribute name="uuidref" type="CharacterString" use="required"/> </
  </xs:complexType>

<xs:group name="UpdatePrimitive">
  <xs:seçim>
    <xs:element name="əlavə et" növü="Əlavə et"/>
    <xs:element adı="dəyişdirin" növü="Dəyişdirin"/>
  </xs:seçim>
</xs:group>
```



```

    <xs:element name="delete" type="Delete"/> </
    xs:choice>
</xs:group>

<xs:complexType name="Yeniləmə">
  <xs:choice maxOccurs="bounded">
    <xs:group ref="UpdatePrimitive"/>
  </xs:choice>
</xs:complexType>

```

C.5.5 Əsas növlər

C.5.5.1 Obyektin identifikasiyası

Bütün obyekt növləri nümunə modelində müəyyən edilmiş IM_Object sinfinə uyğun gələn IM_Object növündən alınmalıdır. O, obyektin identifikasiyası üçün istifadə olunacaq iki atribut müəyyən edir. "id" atributu XML sənədində unikal olmalıdır və XML "ID" məlumat tipi semantikasına uyğun olmalıdır. "uuid" atributu tətbiq domenində müəyyən edilmiş kainat daxilində universal unikal identifikator təmin etmək üçün istifadə oluna bilər.

```

<xs:attributeGroup name="IM_ObjectIdentification">
  <xs:attribute name="id" type="xs:ID"/>
  <xs:attribute name="uuid" type="xs:string"/>
</xs:attributeGroup>

<complexType name="IM_Object">
  <attributeGroup ref="IM_ObjectIdentification"/>
</complexType>

```

C.5.5.2 Obyekt istinadı

IM_ObjectReference növü obyektə istinad etmək üçün istifadə olunacaq üç atributun siyahısını müəyyən edir. "idref" atributu eyni verilənlər toplusunda XML elementi ilə təmsil olunan obyektə istinad etmək üçün istifadə edilməlidir. "uuidref" atributu tətbiq sahəsinin kainatındakı obyektə istinad etmək üçün istifadə edilməlidir. "uuid" dəyərlərini həll etmək üçün ad serverindən istifadə edilə bilər. "Uuref" atributundan İstinad [5] ilə müəyyən edilmiş vahid resurs identifikatorunun (URI) sintaksisindən istifadə edərək digər verilənlər dəstlərində yerləşən obyektlərə istinad etmək üçün istifadə edilməlidir.

IM_ObjectReference tipli nümunə mövcud üç atributdan yalnız biri üçün qiymətlərə malik olmalıdır.

```

<xs:attributeGroup name="IM_ObjectReference">
  <xs:attribute name="idref" type="xs:IDREF"/>
  <xs:attribute name="uuidref" type="xs:string"/>
  <xs:attribute name="uriref" type="URI" /> </
  xs:attributeGroup>

  xs:complexType name="IM_ObjectReference">
    <xs:attributeGroup ref="IM_ObjectReference"/> </
    xs:complexType>

```

NÜMUNƏ 1 Obyekt elementi e öz "id" və "uuid" atributları üçün aşağıdakı dəyərlərə malikdir:

```
<e id="i05" uuid="dce:F6A120B3"> ... </e>
```

NÜMUNƏ 2 l elementi "idref" atributundan istifadə edərək e-yə istinad edə bilər:

```
<l idref="i05"/>
```

NÜMUNƏ 3 "uuidref" atributundan istifadə edərək e-yə istinad edə bilər:

```
<l uuidref="dce:F6A120B3"/>
```

NÜMUNƏ 4 Və ya XPointer ilə birlikdə xarici XML sənədinə istinad edən keçiddən istifadə edərək e-yə istinad edə bilər fraqmentin identifikasiyası. Burada "i05" "id(i05)" üçün qısadır.

```
<l uriref="http://www.example.org/data.xml|i05"/>
```

C.5.6 Dil etiketləmə mexanizmləri

İki dil etiketləmə mexanizmi dəstəklənir.

- Birincisi, qorunan "xml:lang" atributunun istifadəsidir. Qorunan "xml:lang" atributu dil təqini tələb edən elementlərin atributu kimi elan edilməlidir. "xml:lang" mexanizmi elementin atributlarının və məzmununun müəyyən bir dildə olmasını göstərir.
- İkincisi, ISO/IEC 10646:2011 təyyarə 14-ün dil işarələmə mexanizmindən istifadə etməkdir. Əgər simvol sətirində birdən çox dil istifadə olunursa, ISO/IEC 10646:2011 təyyarə 14 mexanizmindən istifadə edilməlidir.

Dil təq simvollarından istifadə qəti şəkildə tövsiyə edilmir. Onlar standartda kodlaşdırılmışdır və yalnız hüquqlu işarələmə mexanizmlərindən istifadə etmədən qısa sətirlər üçün dil etiketlənməsini təmin etmək üçün lazım ola bilən xüsusi protokollar tərəfindən məhdud istifadə üçün. Mətni dil identifikasiyası ilə etiketləmək tələbi olan əksər istifadəçilər HTML, XML və ya digər zəngin mətn mexanizmləri tərəfindən təmin edilənlər kimi standart işarələmə mexanizmlərindən istifadə etməlidirlər. Verilənlər bazası kontekstlərində dil, daxil edilmiş dil təqləri və ya işarələmə ilə deyil, ümumiyyətlə müvafiq məlumat sahələri ilə göstərilməlidir.

NÜMUNƏ Burada CharacterStringWithLanguageTag adı ilə mürəkkəb tip müəyyən edilmişdir. Bilin ki, belədir xml: ad sahəsini təyin etmək lazımdır.

```
<xs:complexType name="CharacterStringWithLanguageTag">  
  <xs:simpleContent>  
    <xs:extension base="xs:string">  
      <xs:attribute ref="xml:lang"/>  
    </xs:extension>  
  </xs:simpleContent>  
</xs:complexType>
```

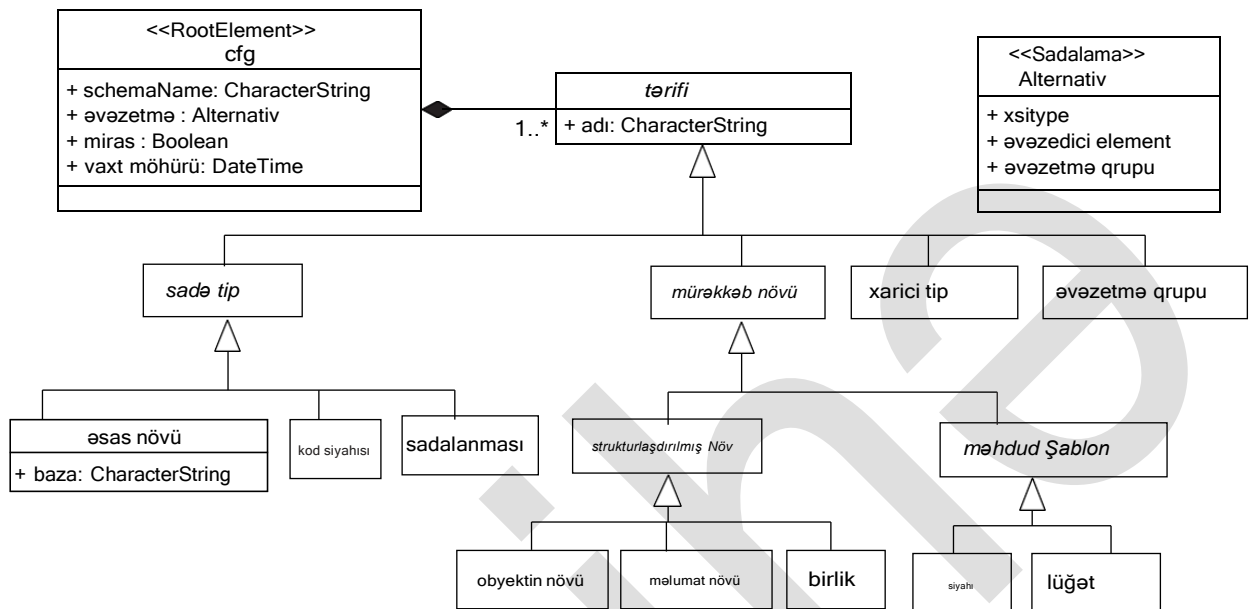
C.5.7 Kodlaşdırma seçimləri

Bir sıra seçimlər indiyə qədər sxemlərin çevrilməsi qaydalarında təsvir edilmişdir. Onlar sənədləşdirilməli və C.5.7 onları sadalayır və bu seçimləri konfigurasiya faylında necə saxlamağı təsvir edir.

- Adlandırma siyasəti. Defolt adlandırma siyasəti XSD-də element adlarını elan edərkən tətbiq sxeminin adından istifadə etməkdir. UML-də sinif, atribut və rol adlarının müəyyən edilməsi qaydaları ISO/TS 19103-də verilmişdir. Alternativ element adları dəstini müəyyən etməklə fərqli adlandırma siyasətinə icazə verilir.
- Müstəqil obyektlər. Defolt, heç bir stereotip və ya <<Tip>> stereotipi olmayan bütün sinifləri obyekt elementləri kimi elan etməkdir. Hansı obyektlərin müstəqil və ya müstəqil olmadığını müəyyən etməklə onlar ləğv edilə bilər.
- XML Sxema miras mexanizmi və ya kopyalayın. Yalnız tək miras istifadə edilərsə, o zaman XML Sxema mirası istifadə edilə bilər. Əks halda, surəti aşağı mexanizmdən istifadə edilməlidir. İstifadə olunan mexanizm təsvir edilməlidir.
- Əvəzetmə mexanizmi. Konfigurasiya faylında istifadə edilən əvəzetmə mexanizminin forması göstərilməlidir.

-Qaydalara istisna. Əgər bəzi siniflər ümumi qaydadan fərqli olaraq xəritələnərsə, bu təsvir edilməlidir.

Konfiqurasiya faylı hər bir sinif üçün elementi olan XML sənədində saxlanılır. Konfiqurasiya faylı müxtəlif siniflərin və xassələrin kodlaşdırılmasının necə baş verəcəyini dəqiq müəyyən etmək üçün istifadə edilməlidir. Konfiqurasiya faylı həmçinin standart element adlarının dəyişdirilməsi mexanizmini təmin edir. Konfiqurasiyanın yuxarı elementləri Şəkil C.24-də təsvir edilmişdir.

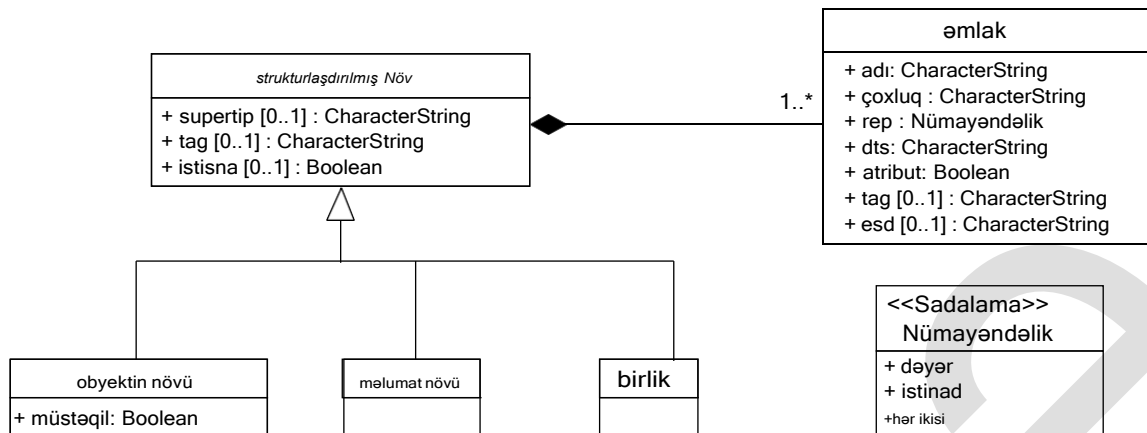


Şəkil C.24 — Konfiqurasiya faylı: yuxarı elementlər

The **cfg** element bir sıra tərif elementlərini təmsil edir. O, sxemin adı, istifadə edilən əvəzetmə mexanizmi, əgər mirasdan istifadə edilərsə və faylın vaxt möhürü (fayl yaradıldığı zaman) haqqında məlumatları saxlayan atributlara malikdir. Fərqli tərif elementləri aşağıdakılardır.

- basicType: <<BasicType>> kimi stereotipli sinfi təmsil edir.
- codelist: <<KodList>> kimi stereotipləşdirilmiş sinfi təmsil edir.
- enumeration: <<Sayadalama>> kimi stereotipləşdirilmiş sinfi təmsil edir.
- objectType: obyekt tipini təmsil edir (stereotipi olmayan və ya <<Tip>> kimi stereotipli sinif). O, tətbiq sxemi sinfi ilə eyni ada və tətbiq sxemi sinfinin atributlarına və assosiasiyalarına uyğun gələn xassələrin siyahısına malik olmalıdır. Müstəqil atribut bu obyektin müstəqil obyekt ola biləcəyini müəyyən edir və buna görə də ayrıca element bəyannaməsi olmalıdır. Teq atributunda element və tip üçün XML adı var və standart dəyər sinif adının özüdür.
- dataType: <<DataType>> kimi stereotipləşdirilmiş sinfi təmsil edir. O, tətbiq sxemi sinfi və xassələrin siyahısı ilə eyni ada malik olmalıdır.
- ittifaq: <<Birlik>> kimi stereotipli sinfi təmsil edir. O, tətbiq sxemi sinfi və xassələrin siyahısı ilə eyni ada malik olmalıdır.
- siyahı: Dəst, Ardıcılıq, Çanta və ya CircularSequence növünün məhdud şablon növünü təmsil edir.
- lüğət: Lüğət növünün məhdud şablon növünü təmsil edir.
- externalType: <<ExternalType>> kimi stereotipli sinfi təmsil edir.
- substitutionGroup: əvəzetmə qrupunu təmsil edir. Bu, konkret ani alt tiplərə malik supertipdir.

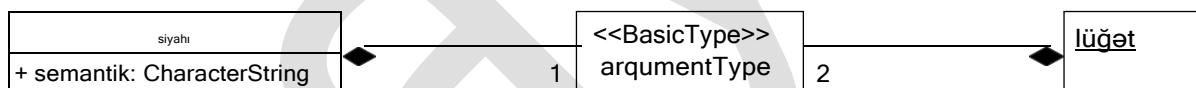
Strukturlaşdırılmış tip obyektType, dataType və birləşmənin abstrakt supertipidir və Şəkil C.25-də təsvir edilmişdir.



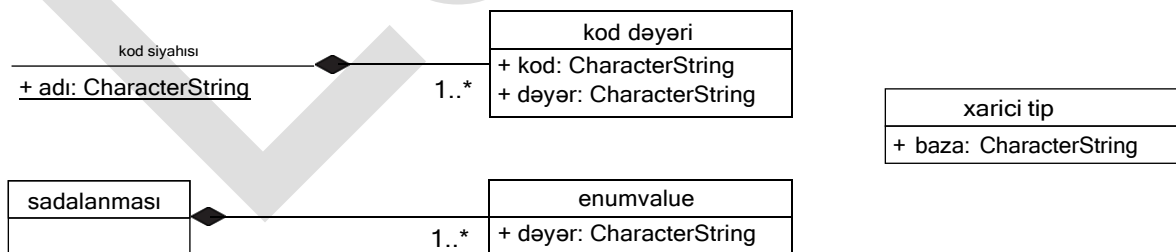
Şəkil C.25 — Konfiqurasiya faylı: strukturlaşdırılmış tiplər

Mülkiyyət elementi sinifin atributunu və ya assosiasiyasını təmsil edir. O, tətbiq sxemindəki müvafiq atribut və ya assosiasiyanın rol adı ilə eyni ada malik olmalıdır. Çoxluq atributunda atributun çoxluğu olmalıdır. rep atributunun üç qiyməti ola bilər: “dəyər” o deməkdir ki, nümunə dts atributunda verilmiş növün dəyər domenindən götürülməlidir; “istinad” o deməkdir ki, atributun nümunəsi onun identifikatorunu dəyər kimi verməklə obyektə istinad etməlidir; və “hər ikisi” o deməkdir ki, o, ya istinad, ya da dəyər ola bilər. dts atributunda proqram sxemindəki hədəf məlumat növünə uyğun gələn növün adı olmalıdır. Əgər rep “istinad”ı göstərsə, o zaman dts-də verilənlər tipləri identikliyə malik strukturlaşdırılmış tiptə olmalıdır.

Əsas tip daha ətraflı Şəkil C.26-da təsvir edilmişdir və kod siyahıları, siyahılar və xarici növlər Şəkil C.27-də daha ətraflı izah edilmişdir.



Şəkil C.26 — Konfiqurasiya faylı: məhdud şablon növləri



Şəkil C.27 — Konfiqurasiya faylı: kod siyahısı, siyahı və xarici tip

C.5.8 Konfigurasiya faylı və XSD—Misal

C.5.8.1 Giriş

C.5.8.2 və C.5.8.3 C.3.3-də verilmiş nümunə tətbiq sxemi üçün müvafiq olaraq konfigurasiya faylı və XSD faylı göstərir.

C.5.8.2 Konfigurasiya faylı

Şəkil C.24-dəki sxemə uyğun konfigurasiya faylı

```
<?xml version="1.0" ?>
<!DOCTYPE cfg SİSTEM "cfg.dtd">

<cfg schemaName="Ex" miras="yanlış" əvəzetmə="əvəzetmə Qrupu" timeStamp="2001-05-31T15:13">

<objectType id="i1" name="C0" müstəqil="true" tag="C0">
  <property id="i2" name="a1" multiplicity="0..1" rep="value" dts="Real" atribut="false" tag="a1"/> <property id="i3" adı
  ="a2" multiplicity="0..*" rep="dəyər" dts="CharacterString" atribut="false" tag="a2"/> <property id="i4" name="a3"
  rep="dəyər" dts="Dictionary_Integer_CharacterString_" atribut="false" tag="a3"/> <xüsusiyyət id="i5" adı="rol1"
  çoxluq="1..*" rep="referans" dts="C1" atribut="false" tag="role1"/> </objectType>

<objectType id="i7" name="C1" müstəqil="true" tag="C1">
  <property id="i8" name="a1" rep="dəyər" dts="Bütün" atribut="false" tag="a1"/>
  <property id="i9" name="main" multiplicity="1" rep="reference" dts="C0" atribut="false" tag="main"/> </
  objectType>

<objectType id="i10" name="C2" müstəqil="true" tag="C2">
  <property id="i11" name="no_items" rep="dəyər" dts="Bütün" atribut="yanlış" tag="no_items"/>
  <property id="i12" name="pos" rep="dəyər" " dts="Nöqtə" atribut="false" tag="pos"/>
  <property id="i13" name="part" multiplicity="0..*" rep="value" dts="C3" atribut="false" tag="part"/> </
  objectType>

<objectType id="i14" name="C3" müstəqil="true" tag="C3">
  <property id="i15" name="a1" rep="dəyər" dts="Tam ədəd" atribut="false" tag="a1"/>
  <property id="i16" name="main" multiplicity="1" rep="reference" dts="C0" atribut="false" tag="main"/>
  <property id="i17" name="a2" " rep="dəyər" dts="Tarix" atribut="false" tag="a2"/>
</objectType>

<objectType id="i18" name="Nöqtə" müstəqil="true" tag="Nöqtə">
  <property id="i19" name="position" rep="value" dts="DirectPosition" atribut="false" tag="position"/> </
  objectType>

<basicType id="i20" name="DirectPosition" base="CharacterString"/>

<dictionary id="i21" name="Dictionary_Integer_CharacterString_">
  <argumentType name="Integer"/>
  <argumentType name="CharacterString"/> </
  dictionary>

<əvəzetmə Qrupu id="i22" adı="C1">
  <type name="C1"/><type name="C3"/> </
  substitutionGroup>
</cfg>
```

C.5.8.3 XSD

Birincisi, Ex.xsd faylı var.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema version="1.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">

<!--PART 1 - Şəxsiyyəti olan obyektlər-->
  <xs:group name="Obyekt">
    <xs:seçim>
      <xs:element name="C0" type="C0"/>
      <xs:element name="C1" type="C1"/>
      <xs:element name="C2" type="C2"/>
      <xs:element adı="C3" type="C3"/>
      <xs:element adı="Nöqtə" növü="Nöqtə"/> </
    xs:choice>
  </xs:group>
<!--PART 2 - Bəyannamələri yazın-->
  <xs:complexType name="C0">
    <xs:ardıcılıq>
      <xs:element name="a1" type="Real" minOccurs="0"/>
      <xs:element name="a2" type="CharacterString" minOccurs="0" maxOccurs="bounded"/>
      <xs:element name="a3" type="Dictionary_Integer_CharacterString_"/>
      <xs:element name="role1" type="ref_C1" maxOccurs="məhdudiyətsiz"/>
      <xs:element name="theC2" minOccurs="0" maxOccurs="məhdudiyətsiz">
        <xs:complexType>
          <xs:sequence minOccurs="0">
            <xs:element name="C2" type="C2"/> </
          xs:sequence>
          <xs:attributeGroup ref="IM_ObjectReference"/> </
        xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attributeGroup ref="IM_ObjectIdentification"/>
  </xs:complexType>

  <xs:complexType name="C1">
    <xs:ardıcılıq>
      <xs:element name="a1" type="Integer"/>
      <xs:element name="main" type="ref_C0"/> </
    xs:sequence>
    <xs:attributeGroup ref="IM_ObjectIdentification"/>
  </xs:complexType>

  <xs:complexType name="C2">
    <xs:ardıcılıq>
      <xs:element name="no_items" type="Integer"/>
      <xs:element name="pos" type="Nöqtə"/>
      <xs:element name="part" type="C3" minOccurs="0" maxOccurs="unbounded"/> </
    xs:sequence>
    <xs:attributeGroup ref="IM_ObjectIdentification"/>
  </xs:complexType>

  <xs:complexType name="C3">
    <xs:ardıcılıq>
      <xs:element name="a1" type="Integer"/>
      <xs:element name="main" type="ref_C0"/>
      <xs:element name="a2" type="Tarix"/> </
    xs:sequence>
    <xs:attributeGroup ref="IM_ObjectIdentification"/>
  </xs:complexType>

```

```

<xs:complexType name="Nöqtə">
  <xs:ardıcılıq>
    <xs:element name="position" type="DirectPosition"/> </
    xs:sequence>
  <xs:attributeGroup ref="IM_ObjectIdentification"/>
</xs:complexType>

<xs:simpleType name="DirectPosition">
  <xs:restriction base="CharacterString"/> </
  xs:simpleType>
<xs:complexType name="Dictionary_Integer_CharacterString">
  <xs:sequence maxOccurs="bounded">
    <xs:element name="Integer" type="Integer"/>
    <xs:element name="CharacterString" type="CharacterString"/> </
    xs:sequence>
</xs:complexType>

<xs:grup adı="C1">
  <xs:seçim>
    <xs:element name="C1" type="C1"/>
    <xs:element name="C3" type="C3"/> </
    xs:choice>
</xs:grup>
<!--PART 3 - Obyekt İstinad növləri-->
<xs:complexType name="ref_C1">
  <xs:attributeGroup ref="IM_ObjectReference"/> </
  xs:complexType>
<xs:complexType name="ref_C0">
  <xs:attributeGroup ref="IM_ObjectReference"/> </
  xs:complexType>
</xs:schema>

```

Sonra proqram sxemi var: Ex_AS.xsd, bütün faylları bir-birinə bağlayan.

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema
  targetNamespace="Ex"
  xmlns="Ex"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  version="1.0">
  <xsd:include schemaLocation="19103.xsd"/>
  <xsd:include schemaLocation="19115.xsd"/>
  <xsd:include schemaLocation="Ex.xsd"/>
  <xsd:include schemaLocation="19118.xsd" /> </
  xsd:schema>

```

C.6 Nümunələrin çevrilməsi qaydaları

C.6.1 XML sənəd strukturu

Nümunələrin çevrilməsi qaydaları C.6-da verilmişdir. Bunlar nümunə modelindəki obyektlərin müvafiq XML elementlərinə necə çevrilməsini təsvir edir. Nümunənin çevrilməsinin nəticəsi etibarlı XML sənədi olan XML sənədidir. Bu faylda aşağıdakılar olmalıdır:

- XML versiyasının emal təlimatı və istəyə bağlı kodlaşdırma bəyannaməsi;
- NÜMUNƏ 1 <?xml version="1.0" encoding="UCS-2" ?>
- XSD faylına uyğun gələn ad sahəsinə istinadla kök element;
 - XSD faylındakı bəyannamələrə uyğun gələn elementlər.

Ümumi fikir ondan ibarətdir ki, nümunə modelindəki obyektlər XSD faylındakı bəyannamələrə uyğun olaraq XML faylındakı elementlərə çevrilir.

NÜMUNƏ 2 XML sənədi adətən aşağıdakı struktura malikdir. Nəzərə alın ki, Ex.xsd faylı hədəf müəyyən etmir ad sahəsi.

```
<?xml version="1.0"?>
<GI    versiya="1.0"
      timeStamp="2001-06-07T23:59" xmlns:xsi="http://
      www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="Ex.xsd">
<exchangeMetadata> ... </exchangeMetadata>
<dataset>
   <objectElement id="o1"> ... </objectElement>
   ...
   ...
</dataset>
<yeniləmə>
   <əlavə> ... </əlavə>
   <dəyişdirin> ... </dəyişdirin>
   <delete uuid="dcd:FD3243"/> </
   update>
</GI>
```

C.6.2 Obyekt

Nümunə modelindəki obyekt müvafiq sinif və element bəyannaməsinə uyğun olaraq obyekt elementinə çevrilməlidir. Elementin məzmunu onun məzmun elementləri olmalıdır. Müstəqil obyekt öz id atributuna malik olmalıdır.

Ümumi yanaşma hər bir obyektə element kimi kodlaşdırmaq və obyektin xassələrinin obyektin məzmun elementləri kimi kodlanmasına imkan verməkdir. Element, prinsipcə, sinif adı ilə eyni ada, məzmun elementləri isə öz xassələri ilə eyni ada malik olmalıdır.

C.6.3 Simvolların kodlaşdırılması

XSD faylları və XML sənədləri bu Beynəlxalq Standartda müəyyən edilmiş simvol kodlaşdırma sxemlərinə uyğun olmalıdır. XML "UTF-8", "UTF-16", "ISO/IEC-10646-UCS-2" və "ISO/IEC-10646-UCS-4" bütün hüquqi xarakter kodlaşdırma sxemlərini dəstəkləyir. Simvolların kodlaşdırılması XML faylının ilk emal bəyannatında müəyyən edilməlidir. Əgər kodlaşdırma verilmirsə, defolt XML "UTF-8"ə uyğundur.

NÜMUNƏ

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

C.6.4 XML kodlaşdırması—Misal

Aşağıda C.3.3-də göstərilən tətbiq sxemi və nümunə modeli və C.5.8-də verilmiş konfigurasiya faylı və XSD faylı ilə müəyyən edilmiş nümunənin XML kodlaşdırılması verilmişdir. Tətbiq sxeminə uyğun gələn XSD faylı "Ex" adlı hədəf ad məkanını müəyyən edir.

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

```
<mes:Gl
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ex="Ex"
  xsi:schemaLocation="Ex Ex_AS.xsd"
  timeStamp="1999-05-31T13:20-05:00"
  version="1.0">
<dataset>
  <C0 id="o1">
    <a1>2.14</a1>
    <a2>A</a2><a2>siyahı</a2><a2></a2><a2>sətirlər</a2>
    <a3><Integer>1</Integer><CharacterString>bir</CharacterString >
      <Integer>2</Integer><CharacterString>iki</CharacterString>
      <Integer>3</Integer><CharacterString>üç</CharacterString> </a3>

    <role1 idref="o8"/><role1 idref="o5"/>
  </C0>
  <C0 id="o2">
    <a2>a</a2><a2>b</a2><a2>c</a2>
    <a3>
      <Integer>1024</Integer><CharacterString>1KB</CharacterString>
      <Integer>2048</Integer><CharacterString>2KB</CharacterString> </
      a3>
    <role1 idref="o3"/><role1 idref="o6"/><role1 idref="o7"/> </
    C0>
  <C1 id="o3">
    <a1>255</a1>
  </C1>
  <C1 id="o8">
    <a1>255</a1>
  </C1>
  <C2 id="o4">
    <no_items>1</no_items>
    <pos id="p">
      <position>235553 3553234 32</position> </
      pos>
    <part id="o7">
      <a1>20</a1>
      <main idref="o2"/>
      <a2>25-02-2000</a2>
    </part>
  </C2>
  <C3 id="o5">
    <a1>20</a1>
    <main idref="o1"/>
    <a2>21-02-2000</a2>
```

```
</C3>  
<C3 id="o6">  
  <a1>30</a1>  
  <main idref="o2"/>  
  <a2>31-12-2000</a2>  
</C3>  
</dataset>  
</ex:Gl>
```

C.7 Abstrakt test toplusu

Bu əlavədəki kodlaşdırma qaydalarına uyğun olaraq kodlaşdırmanın həyata keçirilməsi, Əlavə B-də müəyyən edilmiş abstrakt test paketlərinə əlavə olaraq C.8-də mücərrəd test paketlərinin yerinə yetirilməsini nəzərdə tutur.

Kodlaşdırma qaydası kodlaşdırma xidmətlərinin imkanlarını təsvir edir.

Kodlaşdırma xidmətinin iki növü var. İxtisaslaşdırılmış tip bir və ya daha çox xüsusi proqram sxemlərini dəstəkləyən kodlaşdırma xidmətidir. Ümumi kodlaşdırma xidməti istənilən proqram sxemini dəstəkləyir. Kodlaşdırma xidməti 1-ci uyğunluq səviyyəsinə uyğun olmalıdır və 2-ci uyğunluq səviyyəsinə uyğun ola bilər.

1-ci səviyyə uyğunluğu Əlavə B-də tam təsvir edilmişdir.

Əlavə B-də təsvir edilən 1-ci uyğunluq səviyyəsinə cavab verən kodlaşdırma xidmətinin C.8-dəki bütün tələblərə cavab verən kodlaşdırma qaydası var. 2-ci uyğunluq səviyyəsinə cavab verən kodlaşdırma xidmətinin birbaşa bu əlavədə təsvir olunan XML əsaslı kodlaşdırma qaydasına əsaslanan kodlaşdırma qaydası var. 2-ci uyğunluq səviyyəsinə cavab vermək üçün kodlaşdırma xidmətlərinin C.8-də göstərilənlərə əlavə olaraq B Əlavəsində təsvir olunan tələblərə cavab verməsi lazımdır.

C.8 Səviyyə 2 uyğunluğu

C.8.1 Tətbiq sxemi

- Test məqsədi: Kodlaşdırma xidmətinin UML-də yazılmış bir və ya bir neçə proqram sxemi ilə konfigurasiya edilə biləcəyini və onun bu əlavəyə uyğun olaraq etibarlı XML Sxemlərini yazma biləcəyini yoxlayın.
- Test metodu: Sınaq ediləcək proqram sxemini müəyyən edin və bu proqram sxemini dəstəklədiyini görmək üçün kodlaşdırma xidmətini yoxlayın. Kodlaşdırma xidməti tərəfindən hazırlanmış XML Sxemini yoxlayın və onun çevrilmə qaydalarına əməl edib-etmədiyini və etibarlı XML Sxem yaratdığını yoxlayın.
- İstinad: 6, 7, 8, 9-cu bəndlər və Əlavə C.
- Test növü: Qabiliyyət testi.

C.8.2 Məlumat strukturu

- Test məqsədi: Kodlaşdırma xidmətinin bu Beynəlxalq Standarta uyğun olaraq etibarlı XML sənədlərini oxuya və yazma bildiyini yoxlayın.
- Test metodu: Tətbiq məlumatlarını, onun proqram sxeminə xas məlumat strukturuna necə uyğunlaşdırıldığını, nümunə modelini və çevirmə qaydalarının düzgün tətbiq edilib-edilmədiyini yoxlayın. Bu, XML sənədinin bir hissəsini yoxlamaq və hazırlanmış XML sənədini oxumaq üçün ümumi XML prosessorundan istifadə etməklə həyata keçirilir.
- İstinad: Əlavə C.
- Test növü: Əsas test.

C.8.3 Test məlumatları

- a) Test məqsədi: Kodlaşdırma xidmətinin proqram sxemini müşayiət edən test məlumat dəstini oxuya və yazma bildiyini yoxlayın. Tətbiq sxemi mücərrəd verilənlər bazası əsasında hazırlanmış düzgün XML sənədindən ibarət mücərrəd verilənlər toplusunu və test məlumatlarını birləşdirməlidir.
- b) Test üsulu: Nümunə modelini mücərrəd verilənlər toplusu ilə doldurun və XML sənədi yaradın. Tətbiq sxeminin test məlumatları ilə məntiqi olaraq ekvivalent olub olmadığını görmək üçün bu sənədi müqayisə edin. Test məlumatlarını oxuyun və onları əldə edilən nümunə modeli ilə müqayisə edin. İki növ məlumat arasında uyğunsuzluqlar varsa, test uğursuz olur. Dönüşüm zamanı simvol məlumatlarının itirilmədiyini yoxlayın.
- c) İstinad: Əlavə C.
- d) Test növü: Saxtalaşdırma testi.

Biblioqrafiya

- [1] ISO/IEC 2022:1994, *İnformasiya texnologiyası — Simvol kodu strukturu və genişləndirmə üsulları*
- [2] ISO/IEC 2382-1:1993, *İnformasiya texnologiyaları — Lüğət — 1-ci hissə: Əsas terminlər*
- [3] ISO/IEC 8211:1994, *İnformasiya texnologiyası — Məlumat mübadiləsi üçün verilənlərin təsviri faylı üçün spesifikasiya*
- [4] ISO/IEC 8859 (bütün hissələr), *İnformasiya texnologiyası — 8 bitlik bir bayt kodlu qrafik simvol dəstləri*
- [5] ISO 8879:1986, *İnformasiya emalı — Mətn və ofis sistemləri — Standart Ümumiləşdirilmiş İşarələmə Dili (SGML)*
- [6] ISO/IEC 9834-8, *İnformasiya texnologiyaları — Açıq Sistemlərin Qarşılıqlı Bağlantısı — ACİ Qeydiyyat Orqanlarının fəaliyyəti üçün prosedurlar: Universal Unikal İdentifikatorların (UUID) yaradılması və qeydiyyatı və onların ASN.1 Obyekt İdentifikatoru komponentləri kimi istifadəsi*
- [7] ISO 10303-21, *Sənaye avtomatlaşdırma sistemləri və inteqrasiyası — Məhsul məlumatlarının təqdimatı və mübadiləsi — Hissə 21: İcra üsulları: Mübadilə strukturunun aydın mətn kodlaşdırması*
- [8] ISO/IEC 11179-3:2003, *İnformasiya texnologiyası — Metadata registrləri (MDR) — 3-cü hissə: Reyestr metamodeli və əsas atributları*
- [9] ISO/IEC 11578, *İnformasiya texnologiyası — Açıq Sistemlərin Qarşılıqlı Əlaqəsi — Uzaqdan Prosedur Zəngi (RPC)*
- [10] ISO 19101:2002, *Coğrafi məlumat — İstinad modeli*
- [11] ISO/TS 19104:2008, *Coğrafi məlumat — Terminologiya*
- [12] ISO 19105:2000, *Coğrafi məlumat - Uyğunluq və sınaq*
- [13] ISO 19106:2004, *Coğrafi məlumat — Profillər*
- [14] ISO 19107:2003, *Coğrafi məlumat — Məkan sxemi*
- [15] ISO 19108:2002, *Coğrafi məlumat - Müvəqqəti sxem*
- [16] ISO 19110:2005, *Coğrafi məlumat — Xüsusiyyətlərin kataloqlaşdırılması üçün metodologiya*
- [17] ISO 19111:2007, *Coğrafi məlumat — Koordinatlar üzrə məkan istinadı*
- [18] ISO 19112:2003, *Coğrafi məlumat — Coğrafi identifikatorlarla məkan istinadı*
- [19] ISO 19113:2002, *Coğrafi məlumat - Keyfiyyət prinsipləri*
- [20] ISO 19114:2003, *Coğrafi məlumat — Keyfiyyətin qiymətləndirilməsi prosedurları*
- [21] ISO 19115:2003, *Coğrafi məlumat — Metadata*
- [22] ISO 19116:2004, *Coğrafi məlumat — Yerləşdirmə xidmətləri*
- [23] ISO 19117:2005, *Coğrafi məlumat - Təsvir*
- [24] ISO 19119:2005, *Coğrafi məlumat — Xidmətlər*

- [25] ISO/TR 19120:2001, *Coğrafi məlumat — Funksional standartlar*
- [26] ISO 19135:2005, *Coğrafi məlumat — Elementlərin qeydiyyatı üçün prosedurlar*
- [27] ISO 19136:2007, *Coğrafi məlumat — Coğrafiya İşarələmə Dili (GML)*
- [28] ISO/TS 19139:2007, *Coğrafi məlumat - Metadata - XML sxeminin həyata keçirilməsin*
- [29] ISO/IEC 19501, *İnformasiya texnologiyası — Açıq paylanmış emal — Vahid Modelləşdirmə Dili (UML) Versiya 1.4.2*
- [30] *Sənəd Obyekt Modeli (DOM) Səviyyə 1 Spesifikasiyası, Versiya 1.0, W3C Təvsiyəsi. Oktyabr 1998. Bu ünvanda mövcuddur: <<http://www.w3.org/TR/REC-DOM-Level-1> >*
- [31] IETF STD 1:1998, *İnternet Rəsmi Protokol Standartları*
- [32] KUHN, M. UTF-8 və Unicode tez-tez verilən suallar. < ünvanında mövcuddur <http://www.cl.cam.ac.uk/~mgk25/unicode.html> >
- [33] *XML-də ad boşluqları, W3C Təvsiyəsi. < ünvanında mövcuddur <http://www.w3.org/TR/REC-xml-names> >*
- [34] *Resurs Təsviri Çərçivəsi (RDF) Modeli və Sintaksis Spesifikasiyası, W3C Təvsiyəsi. Fevral 1999. Burada mövcuddur: <<http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/> >*
- [35] *Resurs Təsviri Çərçivəsi (RDF) Sxem Spesifikasiyası 1.0, W3C Namizəd Təvsiyəsi. Mart 2000. <<http://www.w3.org/TR/2000/CR-rdf-schema-20000327/> >*
- [36] RFC 2396, *Vahid Resurs İdentifikatorları (URI): Ümumi Sintaksis. < ünvanında mövcuddur <http://www.ietf.org/rfc/rfc2396.txt> >*
- [37] *Ölçəklənən Vektor Qrafikası (SVG) 1.1, W3C İşçi Layihəsi May 2011. Bu ünvanda mövcuddur: <<http://www.w3.org/TR/SVG/> >*
- [38] Açıq Qrup, CAE Spesifikasiyası, *DCE 1.1: Uzaqdan Prosedur Zəngi, Sənəd Nömrəsi C706. < ünvanında mövcuddur <http://www.opengroup.org/publications/catalog/c706.htm> >*
- [39] www.w3.org/TR/charmod-norm/
- [40] *XML Bağlantı Dili (XLink) Versiya 1.0, W3C Təvsiyəsi. < ünvanında mövcuddur <http://www.w3.org/TR/xlink/> >*
- [41] *XML Metadata Mübadiləsi (XMI), Versiya 2.0, Obyekt İdarəetmə Qrupu. < ünvanında mövcuddur <http://www.omg.org/cgi-bin/doc?formal/2003-05-02> >*
- [42] *XML Path Language (XPath), Versiya 1.0, W3C Təvsiyəsi. Noyabr 1999. Bu ünvanda mövcuddur: <<http://www.w3.org/TR/xpath> >*
- [43] *XML Göstərici Dili (XPointer), W3C İşçi Layihəsi. < ünvanında mövcuddur <http://www.w3.org/TR/xptr> >*
- [44] *XSL Transformations (XSLT), Versiya 1.0, W3C Təvsiyəsi. Noyabr 1999. Bu ünvanda mövcuddur: <<http://www.w3.org/TR/xslt> >*

Əlavə

ICS 35.240.70

Qiymət 69 səhifəyə əsaslanır