



جمهوری اسلامی ایران
Islamic Republic of Iran
سازمان ملی استاندارد ایران

Iranian National Standardization Organization



استاندارد ملی ایران

۱۶۸۹۷-۲

چاپ اول

آذر ۱۳۹۲

INSO
16897-2
1st. Edition
Nov.2013

فناوری اطلاعات - تعامل پذیری و انقیادهای
مراکز ثبت فراداده (MDR-IB) - قسمت ۲:
انقیادهای کدنویسی

**Information technology - Metadata
Registries Interoperability and Bindings
(MDR-IB) - part 2: Coding bindings**

ICS:35.040

به نام خدا

آشنایی با سازمان ملی استاندارد ایران

مؤسسه استاندارد و تحقیقات صنعتی ایران به موجب بند یک ماده ۳ قانون اصلاح قوانین و مقررات مؤسسه استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱ تنها مرجع رسمی کشور است که وظیفه تعیین، تدوین و نشر استانداردهای ملی (رسمی) ایران را به عهده دارد.

نام موسسه استاندارد و تحقیقات صنعتی ایران به موجب یکصد و پنجاه و دومین جلسه شورای عالی اداری مورخ ۹۰/۶/۲۹ به سازمان ملی استاندارد ایران تغییر و طی نامه شماره ۲۰۶/۳۵۸۳۸ مورخ ۹۰/۷/۲۴ جهت اجرا ابلاغ شده است. تدوین استاندارد در حوزه های مختلف در کمیسیون های فنی مرکب از کارشناسان سازمان، صاحب نظران مراکز و مؤسسات علمی، پژوهشی، تولیدی و اقتصادی آگاه و مرتبط انجام می شود و کوششی همگام با مصالح ملی و با توجه به شرایط تولیدی، فناوری و تجاری است که از مشارکت آگاهانه و منصفانه صاحبان حق و نفع، شامل تولیدکنندگان، مصرف کنندگان، صادرکنندگان و وارد کنندگان، مراکز علمی و تخصصی، نهادها، سازمان های دولتی و غیر دولتی حاصل می شود. پیش نویس استانداردهای ملی ایران برای نظرخواهی به مراجع ذی نفع و اعضای کمیسیون های فنی مربوط ارسال می شود و پس از دریافت نظرها و پیشنهادهای در کمیته ملی مرتبط با آن رشته طرح و در صورت تصویب به عنوان استاندارد ملی (رسمی) ایران چاپ و منتشر می شود.

پیش نویس استانداردهایی که مؤسسات و سازمان های علاقه مند و ذی صلاح نیز با رعایت ضوابط تعیین شده تهیه می کنند در کمیته ملی طرح و بررسی و در صورت تصویب، به عنوان استاندارد ملی ایران چاپ و منتشر می شود. بدین ترتیب، استانداردهایی ملی تلقی می شوند که بر اساس مفاد نوشته شده در استاندارد ملی ایران شماره ۵ تدوین و در کمیته ملی استاندارد مربوط که سازمان ملی استاندارد ایران تشکیل می دهد به تصویب رسیده باشد.

سازمان ملی استاندارد ایران از اعضای اصلی سازمان بین المللی استاندارد (ISO)^۱، کمیسیون بین المللی الکتروتکنیک (IEC)^۲ و سازمان بین المللی اندازه شناسی قانونی (OIML)^۳ است و به عنوان تنها رابط^۴ کمیسیون کدکس غذایی (CAC)^۵ در کشور فعالیت می کند. در تدوین استانداردهای ملی ایران ضمن توجه به شرایط کلی و نیازمندی های خاص کشور، از آخرین پیشرفت های علمی، فنی و صنعتی جهان و استانداردهای بین المللی بهره گیری می شود.

سازمان ملی استاندارد ایران می تواند با رعایت موازین پیش بینی شده در قانون، برای حمایت از مصرف کنندگان، حفظ سلامت و ایمنی فردی و عمومی، حصول اطمینان از کیفیت محصولات و ملاحظات زیست محیطی و اقتصادی، اجرای بعضی از استانداردهای ملی ایران را برای محصولات تولیدی داخل کشور و/یا اقلام وارداتی، با تصویب شورای عالی استاندارد، اجباری نماید. سازمان می تواند به منظور حفظ بازارهای بین المللی برای محصولات کشور، اجرای استاندارد کالاهای صادراتی و درجه بندی آن را اجباری نماید. همچنین برای اطمینان بخشیدن به استفاده کنندگان از خدمات سازمان ها و مؤسسات فعال در زمینه مشاوره، آموزش، بازرسی، ممیزی و صدور گواهی سیستم های مدیریت کیفیت و مدیریت زیست-محیطی، آزمایشگاه ها و مراکز کالیبراسیون (واسنجی) و وسایل سنجش، سازمان ملی استاندارد ایران این گونه سازمان ها و مؤسسات را بر اساس ضوابط نظام تأیید صلاحیت ایران ارزیابی می کند و در صورت احراز شرایط لازم، گواهینامه تأیید صلاحیت به آن ها اعطا و بر عملکرد آن ها نظارت می کند. ترویج دستگاه بین المللی یکاها، کالیبراسیون (واسنجی) و وسایل سنجش، تعیین عیار فلزات گرانبها و انجام تحقیقات کاربردی برای ارتقای سطح استانداردهای ملی ایران از دیگر وظایف این سازمان است.

1- International Organization for Standardization

2 - International Electrotechnical Commission

3- International Organization of Legal Metrology (Organisation Internationale de Metrologie Legale)

4 - Contact point

5 - Codex Alimentarius Commission

کمیسیون فنی تدوین استاندارد

« فناوری اطلاعات - تعامل پذیری و انقیادهای مراکز ثبت فراداده (MDR-IB) - قسمت ۲:

انقیادهای کدنویسی »

رئیس:

مشرف، بهنوش

(فوق لیسانس مهندسی فناوری اطلاعات- شبکه‌های کامپیوتری)

سمت و / یا نمایندگی

کارشناس استاندارد

دبیر:

ترابی، مهرنوش

(فوق لیسانس مهندسی فناوری اطلاعات - تجارت الکترونیک)

کارشناس استاندارد

اعضاء: (اسامی به ترتیب حروف الفبا)

ذاکری، صفورا

(فوق لیسانس مهندسی کامپیوتر - نرم‌افزار)

عضو هیات علمی دانشگاه آزاد اسلامی

بندرعباس

زمانی، کرشنا

(فوق لیسانس مهندسی فناوری اطلاعات - تجارت الکترونیک)

کارشناس مرکز رایانه دانشگاه مازندران

سایانی، احمد

(فوق لیسانس مهندسی برق - قدرت)

مدیر دفتر تحقیقات و استانداردها شرکت

برق منطقه‌ای هرمزگان

صحراگرد، جلیل

(فوق لیسانس مهندسی برق - قدرت)

معاون دفتر برنامه‌ریزی و تحقیقات شرکت

برق منطقه‌ای هرمزگان

فهرست مندرجات

صفحه		عنوان
ب		آشنایی با سازمان ملی استاندارد
ج		کمیسیون فنی تدوین استاندارد
و		پیش گفتار
۱	۱	هدف و دامنه کاربرد
۱	۲	مراجع الزامی
۱	۳	اصطلاحات و تعاریف
۱	۴	موارد استفاده این استاندارد
۳	۵	مدل انتزاعی
۳	۱-۵	مروری بر اشیا داده
۳	۲-۵	مشخصات تبادل داده مرجع
۳	۳-۵	مدل ساختار داده
۵	۴-۵	تعیین‌ها، شناسه‌ها و پیمایش
۷	۶	قواعد معناشناسی
۷	۱-۶	انواع داده
۸	۲-۶	ساختار ذاتی
۸	۳-۶	نام‌گذاری سلسله مراتبی
۱۰	۴-۶	خواص مرتبط
۱۰	۵-۶	شناسه‌های پیمایش ادغام‌شده برای ویژگی‌ها
۱۰	۶-۶	قراردادهای نام‌گذاری بیرونی، منطقی و داخلی
۱۱	۷-۶	ویژگی value_
۱۱	۸-۶	واژه‌های کلیدی
۱۲	۷	انقیادها
۱۲	۸	اداره
۱۲	۱-۸	استفاده از انواع داده تعریف شده توسط مرکز ثبت
۱۲	۹	انطباق
۱۲	۱-۹	الگوی انطباق کدنویسی

ادامه فهرست مندرجات

صفحه	عنوان
۱۳	۲-۹ انطباق نمونه داده
۱۴	۳-۹ انطباق کاربرد داده
۱۵	۴-۹ برچسب‌های انطباق
۱۶	۱۰ محفوظ برای استانداردسازی آینده
۱۶	۱۱ انقیاد کدنویسی DIVP
۱۶	۱-۱۱ کلیات
۱۸	۲-۱۱ تولید و ایجاد DIVP
۲۰	۳-۱۱ مصرف و تفسیر DIVP
۲۱	۴-۱۱ نمایش انواع داده پایه
۲۴	۵-۱۱ کدنویسی نمایش‌های نویسه
۲۵	۶-۱۱ مدیریت استثناها و توسعه‌ها
۲۵	۷-۱۱ پیشوند برچسب انطباق
۲۶	۱۲ انقیاد کدنویسی XML
۲۶	۱-۱۲ کلیات
۲۶	۲-۱۲ تولید و ایجاد XML
۲۹	۳-۱۲ مصرف و تفسیر XML
۳۰	۴-۱۲ نمایش انواع داده پایه
۳۳	۵-۱۲ کدنویسی نمایش‌های نویسه
۳۳	۶-۱۲ مدیریت استثناها و توسعه‌ها
۳۴	۷-۱۲ پیشوند برچسب انطباق
۳۵	کتابنامه

پیش‌گفتار

استاندارد «فناوری اطلاعات- تعامل‌پذیری و انقیادهای مراکز ثبت فراداده (MDR-IB) - قسمت ۲: انقیادهای کدنویسی» که پیش‌نویس آن در کمیسیون فنی مربوط، توسط سازمان ملی استاندارد ایران تهیه و تدوین شده و در دویست و نود و سومین اجلاس کمیته ملی استاندارد رایانه و فرآوری داده مورخ ۱۳۹۲/۸/۲۰ مورد تصویب قرار گرفته است اینک به استناد بند یک ماده ۳ قانون اصلاح قوانین و مقررات موسسه استاندارد و تحقیقات صنعتی ایران مصوب بهمن ماه ۱۳۷۱ به عنوان استاندارد ملی منتشر می‌شود. برای حفظ همگامی و هماهنگی با تحولات و پیشرفت‌های ملی و جهانی در زمینه صنایع، علوم و خدمات، استانداردهای ملی ایران در مواقع لزوم تجدید نظر خواهند شد و هر گونه پیشنهادی که برای اصلاح و تکمیل این استانداردها ارائه شود، هنگام تجدید نظر در کمیسیون فنی مربوط مورد توجه قرار خواهد گرفت. منبع و ماخذی که در تهیه این استاندارد مورد استفاده قرار گرفته است به شرح زیر است:

ISO/IEC 20944-2:2013, Information technology - Metadata Registries Interoperability and Bindings (MDR-IB) - part 2: Coding bindings

فناوری اطلاعات - تعامل پذیری و انقیادهای مراکز ثبت فراداده (MDR-IB) ^۱ - قسمت ۲: انقیادهای کدنویسی

۱ هدف و دامنه کاربرد

مجموعه استانداردهای ISO/IEC 20944 به شرح کدنویسی‌ها، واسط‌های برنامه‌نویسی کاربردی (API) ^۲ و پروتکل‌هایی برای تعامل با مرکز ثبت فراداده استاندارد ISO/IEC 11179 می‌پردازد. هدف از تدوین این استاندارد، تعیین تمهیداتی است که میان انقیادهای کدنویسی مجموعه استانداردهای ISO/IEC 20944 مشترک است. همچنین شامل خود انقیادهای کدنویسی شخصی نیز می‌باشد.

۲ مراجع الزامی

مدارک الزامی زیر حاوی مقرراتی است که در متن این استاندارد ملی ایران به آن‌ها ارجاع داده شده است. بدین ترتیب آن مقررات جزئی از این استاندارد ملی ایران محسوب می‌شود. در صورتی که به مدرکی با ذکر تاریخ انتشار ارجاع داده شده باشد، اصلاحیه‌ها و تجدیدنظرهای بعدی آن موردنظر این استاندارد ملی ایران نیست. در مورد مدارکی که بدون ذکر تاریخ انتشار به آن‌ها ارجاع داده شده است، همواره آخرین تجدیدنظر و اصلاحیه‌های بعدی آن‌ها مورد نظر است. استفاده از مراجع زیر برای این استاندارد الزامی است:

- 2-1 ISO/IEC 11404:2007, Information technology — General-Purpose Datatypes (GPD)
- 2-2 ISO/IEC 20944-1:2013, Information technology — Metadata Registries Interoperability and Bindings (MDR-IB) — Framework, common vocabulary, and common provisions for conformance

۳ اصطلاحات و تعاریف

در این استاندارد، اصطلاحات و تعاریف استفاده‌شده در استاندارد ISO/IEC 20944-1، به کار می‌رود.

۴ موارد استفاده این استاندارد

انقیادهایی که نگاشت یک استاندارد (یا چارچوب) را به استاندارد (چارچوب) دیگر ^۳ مدنظر قرار می‌دهد. انقیادهای کدنویسی که نگاشت نمونه‌ای از مدل‌های داده را به عناصر کد ^۴ (نمایش‌های داده) مدنظر قرار می‌دهد.

1 - Metadata Registries Interoperability and Bindings

2 - Application Programming Interfaces

۳ - مثلاً، انقیاد ASN.1 از "مشخصات فنی XYZ"، به معنی نگاشت ویژگی‌ها و الزامات "مشخصات فنی XYZ" به ویژگی‌ها و قابلیت‌های استاندارد ASN.1 است.

4 - Code Elements

بیش از یک استاندارد (چارچوب) ممکن است برای تکمیل نگاشت، استفاده شود.^۱ مجموعه استانداردهای ISO/IEC 20944 از حداقل سه لایه نگاشت استفاده می‌کنند. اولین لایه مربوط به نوع اصلی نگاشت برای انقیاد است: انقیادهای کدنویسی، انقیادهای واسط برنامه‌نویسی کاربردی (API)^۲ و انقیادهای پروتکل. دومین لایه نگاشت، مربوط به نگاشت نمونه‌های مدل داده به نمایش مستقل از کدنویسی (CIR)^۳ داده است؛ استاندارد ISO/IEC 11404 قواعد نحوی^۴ و قواعد معناشناسی^۵ این CIR را مشخص می‌کند. سومین لایه از نگاشت، مربوط به نگاشت CIR به نمایش ویژه کدنویسی (CSR)^۶ - بندها ۱۱ و بعد از آن، نگاشت‌های ویژه کدنویسی را شرح می‌دهد. لایه‌های اضافی نگاشت ممکن است، مانند مشخصات کدنویسی نگاشت‌ها^۷.

هدف از یک CIR مشترک داده، پشتیبانی از قواعد معناشناسی مشترک و تعامل بین انقیادهای کدنویسی و سایر انقیادها می‌باشد، مانند API و انقیادهای پروتکل.

یادآوری - انقیادهای XML، برای ادغام با سایر روش‌های XML، مفید است. انقیادهای XML، الزامات تبادل XML را بدون مشخص کردن روش‌های XML (مثلاً طرح XML^۸)، فراهم می‌کنند تا ویژگی‌ها را پیاده‌سازی کنند. انقیادهای جفت مقدار شناسه نقطه‌گذاری شده (DIVP)^۹ برای ادغام با روش‌های جفت نام - مقدار مفید هستند، مثلاً سامانه‌های اسکریپت نویسی، پست الکترونیک و سرویس‌دهنده‌های وب.

مثال: یک CIR، برای مدل داده ایجاد شده است. بر اساس این CIR، انتقال یک انقیاد کدنویسی (مثلاً انقیاد کدنویسی XML) به انقیاد کدنویسی دیگر (مثلاً انقیاد کدنویسی ASN.1) در حالی که قواعد معناشناسی مشترک (CIR) را بین انقیادهای کدنویسی به اشتراک می‌گذارد، امکان‌پذیر است. به همین ترتیب، یک نمایش کدنویسی (مثلاً XML) می‌تواند به نمایش ویژه کدنویسی دیگر (مثلاً ASN.1) تبدیل شود.

۱ - هنگامی که به‌عنوان مجموعه‌ای از لایه‌ها مشاهده می‌شود (مثلاً، data model = Standard XYZ, coding binding = ASN.1, encoding = ASN.1 Basic Encoding Rules (BER))، انقیادها ممکن است به‌عنوان «استانداردهای لایه لایه» یا «لایه‌بندی استاندارد» هم مشاهده شوند.

- 2 - Application Programming Interface
- 3 - Coding-Independent Representation
- 4 - Syntax
- 5 - Semantics
- 6 - Coding-Specific Representation

۷ - همه انقیادهای کدگذاری XML، ASN.1 و DIVP، برای کدگذاری لایه‌های اضافی دارند. XML دو لایه کدگذاری اضافی دارد: کدگذاری نویسه (مثلاً ASCII در مقابل UTF-8) و لایه پایین‌تر کدگذاری مرتب‌کردن بایت‌ها برای نمایش‌های چندبیتی (مثلاً ترتیب endian کوچک و endian بزرگ برای UTF-16 و UTF-32). ASN.1 یک لایه اضافی کدگذاری دارد، قواعد کدگذاری (BER)، قواعد کدگذاری متعارف (CER) و قواعد کدگذاری مشخص (DER). DIVP ممکن است قواعد کدگذاری داشته باشد که توسط پوشش بیرونی آن مشخص می‌شود، اما آنها قسمتی از انقیاد کدگذاری DIVP نیستند.

- 8 - XML Schema
- 9 - Dotted Identifier Value Pair

۵ مدل انتزاعی

انقیادهای کدنویسی در مفاهیم نمونه‌های داده و ساختار درونی آن‌ها، اشتراک دارند. مثلاً، ویژگی‌های مشترک شامل موارد زیر هستند:

- استفاده از انواع داده برای توصیف ماهیت و عملیات بر روی داده.
- استفاده از استاندارد ISO/IEC 11404 برای تعریف و اعلان^۱ نوع داده.
- استفاده از ساختارهای تجمیعی مشترک، مانند آرایه و رکورد برای توصیف مجموعه‌ای از داده.
- استفاده از توصیف‌های پیمایش مشترک برای مرجع‌دهی مولفه‌های درون یک مجموعه داده.

انقیادهای کدنویسی شخصی (از بند ۱۱ به بعد)، یک نداشت از قواعد نحوی داده مشترک را به الزامات انقیاد شخصی پیوند می‌دهد.

۵-۱ مروری بر اشیا داده

مدل مفهومی از اشیا داده، به دو قسمت تقسیم می‌شود: مدل ساختار داده و مدل پیمایش. مدل ساختار داده، ساختار منطقی داده را هنگامی که داده منتقل می‌شود توصیف می‌کند. مدل پیمایش، نداشت ساختار منطقی به سلسله مراتب پیمایش را توصیف می‌کند.

یادآوری - استفاده از نام‌گذاری سلسله مراتبی، به معنی مدل داده سلسله مراتبی یا مدل فراداده نمی‌باشد.

۵-۲ مشخصات تبادل داده مرجع

مجموعه استانداردهای ISO/IEC 20944 بوسیله مشخصات تبادل داده ضمنی، سازماندهی می‌شود. این مشخصات تبادل داده، برای انقیادهای کدنویسی، API و پروتکل، بیرونی^۲ می‌باشد.

۵-۳ مدل ساختار داده

۵-۳-۱ اشیا داده

۵-۳-۱-۱ کلیات

یک واحد یا مجموعه داده یک «شی داده»^۳ نامیده می‌شود. اشیا داده، داده ساختاریافته است که توسط زیربندهای زیر مشخص می‌شود.

۵-۳-۱-۲ اشیا داده تفکیک‌ناپذیر^۴

یک شی ممکن است «شی داده تفکیک‌ناپذیر» باشد اگر شی داده دارای مقادیری باشد که به‌طورذاتی غیرقابل تقسیم باشد. مثلاً، یک نوع پایه مثل عدد صحیح، حقیقی، نویسه.

مثال: 17، -1.7E8، 0D19980831235959، "hello world"

1 - Declare

2 - External

۳ - با "اشیا" از تجزیه و تحلیل، طراحی و برنامه نویسی شی‌گرا اشتباه نشود.

4 - Atomic Data Objects

۳-۱-۳-۵-۳-۵ تجمیع‌ها^۱

یک تجمیع، مجموعه‌ای از اشیا داده (تجزیه‌پذیر یا تفکیک‌ناپذیر) است. هر عضو تجمیع، یک «مولفه» نامیده می‌شود. مولفه‌ها ممکن است مرتب یا نامرتب باشند، دارای نام یا بدون نام باشند، دارای نوع یا بدون نوع باشند. یک تجمیع ممکن است تودرتو باشد، یعنی یک مولفه از تجمیع می‌تواند خودش یک تجمیع باشد.

مثال: فهرست { X: 17 Y: { 18 19 20 } Z: 0D19980831232359 } شامل سه قلم داده نام‌گذاری شده X، Y و Z است. اولین مولفه (X) یک شی داده تفکیک‌ناپذیر است. دومین مولفه (Y) تجمیع سه مولفه بدون نام است. سومین مولفه (Z) یک شی داده تفکیک‌ناپذیر از نوع تاریخ- زمان است. تفاوت بین تجمیع و ساختار زبان برنامه‌نویسی C عبارتست از: ساختار C همیشه مرتب‌شده، نام‌گذاری شده و دارای نوع است؛ درحالی‌که یک تجمیع ممکن است مرتب‌شده، نام‌گذاری شده و/یا دارای نوع باشد.

۳-۱-۴-۵-۳-۵ تشکیلات سلسله مراتبی

تودرتویی بر تشکیلات سلسله مراتبی فقط از جنبه دسترسی داده نه از جنبه پیاده‌سازی اشاره دارد. برای مثال، آرایه چندبعدی ممکن است از روش دسترسی^۲ مشابهی به‌عنوان لیست تودرتو استفاده کند. به‌عنوان مقایسه، هنگامیکه در زبان برنامه‌نویسی C، آرایه دو بعدی `char x[10][20]` با فهرست تودرتوی `char *y[10]` مقایسه می‌شود، هر دو نوع توسط روش سلسله مراتبی مشابه یعنی `x[a][b]` و `y[a][b]` در دسترس قرار می‌گیرند.

۳-۱-۵-۵-۳-۵ انواع داده

اشیا داده ممکن است یک نوع داده داشته باشند. استاندارد ISO/IEC 11404 شامل نوع‌های پایه (مثلاً، Boolean، عدد صحیح، عدد حقیقی، تاریخ- زمان، رشته)، پارامترهای نوع (مثلاً، دقت) و عملگرهای نوع (مثلاً فهرست‌ها، رکوردها، مجموعه‌ها) می‌باشد. پیاده‌سازی باید انواع داده در استاندارد ISO/IEC 11404 را پشتیبانی کند^۳. انواع داده پشتیبانی شده اضافه، هنگام پیاده‌سازی تعریف می‌شوند.

۳-۱-۲-۵-۳-۵ ویژگی‌ها

۳-۱-۲-۱-۵-۳-۵ کلیات

ویژگی‌ها می‌توانند با اشیا داده مرتبط باشند. ویژگی‌ها، صفت‌هایی^۴ از شی داده هستند.

۳-۱-۲-۲-۵-۳-۵ فهرست‌های ویژگی‌ها

هر شی داده ممکن است فهرست ویژگی مرتبط داشته باشد. فهرست ویژگی‌ها، یک فهرست مرتب و نام‌گذاری شده از اشیا داده است.

- 1 - Aggregates
- 2 - Access Method
- 4 - Attribute

۳ - استاندارد ISO/IEC 11404 شامل توصیفی از انطباق نوع داده است.

مثال: شی داده { 1 2 3 } ممکن است فهرست ویژگی { read_write_access: read_only size: 123 } داشته باشد.

۳-۲-۳-۵ سامانه و ویژگی‌های کاربر

برخی ویژگی‌ها توسط سامانه اساسی مثل `_type`, `_shape`, `_last_modified` ایجاد می‌شوند. برخی از ویژگی‌ها توسط کاربر (یا برنامه کاربردی) ایجاد می‌شوند، مانند `read_write_access`, `size`. قواعد تعیین (مثلاً، زیرخط ابتدایی برای ویژگی‌های سامانه) کمک می‌کند که بین ویژگی‌های کاربر/برنامه کاربردی و ویژگی‌های سامانه تصادم رخ ندهد.

۳-۲-۴-۵ ویژگی‌های ایستا و پویا

ویژگی‌های ایستا برای طول زمان^۱ [وجود] شی وجود دارند و می‌توانند فهرست شوند، مثل `_type`, `color`. ویژگی‌های پویا ممکن است شمارشی^۲ نباشند و برای مدت شی وجود نداشته باشند.

۳-۲-۵-۵ ویژگی‌های ذخیره‌شده و محاسبه‌شده

برخی از ویژگی‌ها انباره صریح مرتبط با شی دارند، مثلاً `color` و `_type` (احتمالاً). برخی ویژگی‌ها ممکن است ضمنی یا محاسبه شده باشند یا انباره نداشته باشند، مثل `_type` و `_shape`.

۳-۲-۶-۵ ویژگی‌های حافظه و فرآر

برخی از ویژگی‌ها مانند حافظه رفتار می‌کنند یعنی اگر ویژگی بیش از یکبار خوانده شود، همیشه مقدار مشابه برمی‌گرداند و نوشتن مقدار مشابه بیش از یکبار، تاثیری همانند هنگامی دارد که یکبار نوشته می‌شود.

مثال: ویژگی‌های رنگ و قیمت، ویژگی از نوع حافظه هستند زیرا آن‌ها بصورت یکسان می‌مانند. ویژگی `_last_modified` ویژگی از نوع فرآر است زیرا ممکن است تغییر کنند. ویژگی `usage_payment` ویژگی از نوع فرآر است زیرا مقادیرش اگر بیش از یکبار تنظیم می‌شود (مثلاً هزینه استفاده از 0.50 USD بیش از یکبار پرداخت می‌شود) ممکن است تاثیرش متفاوت از زمانی باشد که فقط یکبار تنظیم شود.

۳-۲-۷-۵ ویژگی‌های نوع داده

برخی ویژگی‌ها خود ذاتا نوع داده هستند، مثل مرتب در مقابل نامرتب. GPD برخی از این ویژگی‌ها را مشخص می‌کند. سایر ویژگی‌ها ممکن موجود باشند.

۴-۵ تعیین‌ها^۲، شناسه‌ها و پیمایش

۴-۵-۱ کلیات

اشیا تفکیک‌ناپذیر به بیش از یک مولفه اشاره دارند. روش‌های تعیین، پیمایش داده ساختاریافته در اشیا را تسهیل می‌کنند.

1 - Duration
2 - Enumerable
3 - Designations

۴-۵-۲ شناسه‌ها برای پیمایش

شناسه، یک مشخصه است که برای مرجع‌دهی استفاده می‌شود.^۱ مفهوم پیمایش توسط قواعد نحوی URI تعریف می‌شود که با اندیس‌ها و سلسله مراتب قرار دارد.

۴-۵-۲-۱ تعیین‌ها برای اندیس‌ها^۲

اقلام در فهرست‌های مرتب ممکن است بوسیله اعداد قابل دسترس باشند. مثلاً از فهرست مقابل، قلم شماره ۲، برابر با ۱۷ است: { 15 16 17 }. اقلام در فهرست‌ها ممکن است بوسیله شناسه دسترسی شود. مثلاً قلم شماره Z در فهرست مقابل، ۱۷ است: { x: 15 16 z: 17 }. اقلام برچسب‌دار (یعنی اقلام با یک شناسه) از فهرست‌های مرتب، ممکن است بوسیله شناسه عدد دسترسی شود. مثلاً سومین قلم از { x:15 16 z: 17 } ممکن است توسط شناسه Z یا عدد شماره ۲ دسترسی شود. اعداد اندیس از صفر شروع می‌شود.

۴-۵-۲-۲ شناسه‌های سلسله مراتبی

شناسه‌ها سلسله مراتبی هستند زیرا اشیایی که آن‌ها پیمایش می‌کنند، نام‌گذاری سلسله مراتبی دارند. این قواعد نام‌گذاری سلسله مراتبی همانند همه شکل‌های دیگر شناسه‌های داده بیرونی که پیمایش سلسله مراتبی را اجازه می‌دهند، مسیرهای پیمایش سلسله مراتبی نیز نامیده می‌شوند.

مثال: همه نام‌های C/Z, C/2, 2/Z و 2/2 قلم ۱۷ را در شی داده { X: 15 Y: 16 Z: 17 } { A: 10 B: 11 C: } تعیین می‌کنند.

۴-۵-۲-۳ شناسه‌های پیمایش ادغام شده^۳

شناسه‌های پیمایش شی، ویژگی، پیوند، کنترل و غیره، با شناسه‌های پیمایش شی داده ادغام می‌شوند تا تعداد روش‌های دسترسی را ساده کنند. یعنی به عملیات جداگانه `getvalue` و `getProperty` نیاز نباشد.

مثال: ویژگی Y از شی X از طریق نام X.Y قابل دسترس است. برای پیوند نرم L که به D اشاره می‌کند: (۱) گرفتن مقدار L، پیوند را دنبال می‌کند و مقدار D را برمی‌گرداند (۲) گرفتن مقدار L/ به خود پیوند اشاره می‌کند. یعنی ارجاع به D، نه مقدار D (۳) گرفتن مقدار L/ پیوند را دنبال می‌کند و D را بازیابی می‌کند.

۴-۵-۲-۴ قراردادهای تعیین^۴

چندین قرارداد تعیین، برای افزایش تعامل و کاهش هزینه یکپارچگی استفاده می‌شود. کاربر (برنامه کاربردی) از شناسه‌های بیرونی استفاده می‌کند. سامانه (پیاده‌سازی) از شناسه‌های داخلی استفاده می‌کند. شناسه‌های بیرونی به/از شناسه‌های منطقی نگاشت می‌شوند. شناسه‌های منطقی ترتیبی از جداکننده‌های نام مسیر و قطعه‌های نام مسیر هستند. هر قطعه نام مسیر شامل کلماتی است که با جداکننده کلمات، جدا می‌شود. شناسه‌های منطقی به/از شناسه‌های داخلی نگاشت می‌شوند.

۱ - همانطور که در این متن استفاده می‌شود، اصطلاح "شناسه" یک تعیین است که برای پیمایش استفاده می‌شود.

2 - Indexes
3 - Merged
4 - Conventions

مثال: قراردادهای تعیین بیرونی از نام‌های MIME استفاده می‌کنند، در حالی که شناسه‌های داخلی از قراردادهای زبان برنامه‌نویسی C استفاده می‌کنند. نام MIME بیرونی `Abc-Def-Ghi/3/Jkl-Mno` به مسیر نام منطقی نگاشت می‌شود: اولین قطعه مسیر کلمات `Abc`، `Def` و `Ghi` است؛ دومین قطعه مسیر کلمه `3` است؛ سومین قطعه مسیر کلمه `Jkl Mno` می‌باشد. شناسه منطقی به شناسه داخلی زبان برنامه‌نویسی `C` `Abc_Def_Ghi[3].Jkl_Mno` یا احتمالاً `Abc_Def_Ghi[3]->Jkl_Mno` نگاشت می‌شود.

۵-۴-۲-۵ پیوندهای سخت

یک شی داده ممکن است بوسیله بیش از یک شناسه مورد دسترسی قرار گیرد. یک پیوند سخت، اولین ارجاع کلاس به شی داده است. یک شی داده تا زمانی که همه پیوندهای سخت آن پاک نشود، وجود دارد. پس از پاک شدن همه پیوندهای سخت، شی داده خراب می‌شود^۱.

۵-۴-۲-۶ پیوندهای نرم

شناسه دیگری می‌تواند برای شی داده ایجاد شود. یک پیوند نرم، دومین ارجاع کلاس به شی است: پیوند حتی ممکن است بعد از اینکه شی داده مقصد خراب شود، وجود داشته باشد. پیوندهای نرم، بطور ضمنی دنبال می‌شوند (بطور خودکار رجوع می‌شوند).

مثال: اگر پیوند نرم `L` به `D` اشاره کند، سپس (۱) گرفتن مقدار `L`، پیوند را دنبال می‌کند و مقدار `D` را برمی‌گرداند (۲) گرفتن مقدار `L/` به خود پیوند اشاره می‌کند. یعنی ارجاع به `D`، نه مقدار `D` (۳) گرفتن مقدار `L/` پیوند را دنبال می‌کند و را بازیابی می‌کند.

۵-۴-۲-۷ ارجاع

یک شناسه یا اشاره‌گر به شی داده است. یک ارجاع، دومین نام کلاس است: پیوند ممکن است حتی بعد از اینکه شی داده مقصد خراب شود، وجود داشته باشد. ارجاع‌ها باید، بطور ضمنی دنبال شوند (رجوع شوند).

۵-۴-۲-۸ منظرها^۲

یک منظر، زیرمجموعه‌ای از داده ساختار را نشان می‌دهد. یک منظر ساده ممکن است زیردرختی از داده ساختار باشد. یک منظر پیچیده ممکن است یک پرسمان انتخاب `SQL` برای توصیف زیرمجموعه باشد. منظرها، زیرمجموعه‌های اطلاعات را توصیف می‌کنند. یک منظر ممکن است برای آدرس‌دهی نیاز به ایجاد زیرمجموعه‌های کارکردی استفاده شود.

۶ قواعد معناشناسی

۶-۱ انواع داده

انواع داده استاندارد `ISO/IEC 20944`، بر مبنای استاندارد `ISO/IEC 11404` است.

۲-۶ ساختار ذاتی^۱

اشیا شامل داده‌هایی هستند که از طریق سامانه شناسه پیمایش سلسله مراتبی پیمایش می‌شوند.

مثال:

```
struct
{
    int id;
    char name[100];
    time_t datetime;
};
```

ساختار زبان برنامه‌نویسی C بطور مفهومی نمایش داده شده است:

Name	Value
id	11223344
name	John Doe
datetime	19980102

ساختار مفهومی مشابه ممکن است برای XML استفاده شود:

```
<NAMEINFO>
  <ID>11223344</ID>
  <NAME>John Doe</NAME>
  <DATETIME>19980102</DATETIME>
</NAMEINFO>
```

۳-۶ نام‌گذاری سلسله مراتبی

داده ساخت‌یافته^۲ است و از طریق سامانه شناسه پیمایش سلسله مراتبی مورد دسترسی واقع می‌شود.

مثال:

```
struct
{
    int id;
    struct
    {
        char last[40];
        char first[30];
        char middle[30];
        char title[10];
    } name;
    time_t datetime;
};
```

-
- 1 - Inherent
 - 2 - Structured

};

ساختار تودرتوی C ممکن است بطور مفهومی نشان داده شود:

Name	Value
id	11223344
name	last Doe
name	first John
name	middle Q.
name	title Dr.
datetime	19980102

مهمترین ویژگی، شناسه پیمایش سلسله مراتبی است. با استفاده از قواعد نحوی C به عنوان مثال، id, name.last, name.first, name.middle, name.title, datetime است، نه پیاده‌سازی ساختار، لذا شرح ذیل یک مدل مفهومی معادل از منظر MDRIB است:

Name	Value
id	11223344
name	*----- -----+
datetime	19980102

V

Name	Value
last	Doe
first	John
middle	Q.
title	Dr.

ساختارهای داده مفهومی بالا را می‌توان توسط قرارداد تعیین سامانه فایل‌ی UNIX^۱ (یعنی name/last, id, name/first, name/middle, name/title, datetime) یا توسط قرارداد تعیین انباره پنجره^۲ (یعنی id, name\last, name\first, name\middle, name\title, datetime) نمایش داد.

1 - UNIX Filesystems
2 - Windows Registry Designation Convention

۴-۶ خواص مرتبط

اشیا داده می‌توانند خواص مرتبط باشند. بطور مفهومی، ویژگی‌ها فهرستی از جفت شناسه-مقدار هستند. ویژگی‌ها ممکن است توسط سیستم^۱ (مثلا، آدرس اشاره گر) یا توسط کاربر^۲ (مثلا، امنیت) تعریف شوند، ایستا باشند (مثلا، نوع) یا پویا (مثلا، زمان اصلاح قبلی)، به انباره^۳ نیاز داشته باشند (مثلا، امنیت) یا بنابر تقاضا محاسبه شوند (مثلا، طول).

Name	Value
id	11223344
Property List	
name	John Doe
datetime	19980102
V	
Name	Value
address	0x12345678
security	read-write
type	string
modified	19980103
length	???

۵-۶ شناسه‌های پیمایش ادغام‌شده برای ویژگی‌ها

شناسه‌های پیمایش ویژگی‌ها با شناسه‌های پیمایش اشیا داده ادغام می‌شوند تا دسترسی به اطلاعات آسان‌تر باشد. مثلا به جای عملیات "get value" و "get property value"، فقط "getvalue" مورد نیاز است. بنابر فرض‌های قراردادهای تعیین، «/» به عنوان جداکننده مسیر نام و «.» به عنوان معرفی ویژگی، ویژگی "_type" از شناسه پیمایش شی داده، از طریق "name/._type" دسترسی می‌شود.

۶-۶ قراردادهای نام‌گذاری بیرونی، منطقی و داخلی

قراردادهای تعیین ممکن است تغییر کند. به طور مفهومی «شناسه بیرونی» نامی است که در تبادل اطلاعات استفاده می‌شود. مثلا شناسه پیمایش در خدمت "get" استفاده می‌شود. «شناسه بیرونی» به یک «شناسه منطقی» نگاشت می‌شود. یک «شناسه منطقی»، از فهرستی از قطعات نام مسیر تشکیل می‌شود. هر قطعه، فهرستی از قطعات کلمات است. شناسه منطقی «در حین پیاده‌سازی داده ساخت یافته، به شناسه داخلی» نگاشت می‌شود.

1 - System-defined
2 - User-defined
3 - Storage

مثالهایی از شناسه‌های بیرونی با قراردادهای تعیین متغیر، با استفاده از جداکننده‌های مسیر (مثلاً، ".", "/", "\"") و جداکننده‌های کلمات (مثلاً، "-", "_", "'") یا تغییر کوچک و بزرگی حروف)

Abc-Def/Ghi-Jkl/Mno-Pqr-Stu/123/456 # MIME or UNIX filesystem

abc_def.ghi_jkl.mno_pgr_stu[123][456] # C language

AbcDef\GhiJkl\MnoPqrStu\123\456 # Windows Registry

((abc def) (ghi jkl) (mno pqr stu) 123 456) # LISP

نام‌های بیرونی بالا به «شناسه‌های منطقی» زیر نگاشت می‌شوند:

path segment #1:

word #1: abc

word #2: def

path segment #2:

word #1: ghi

word #2: jkl

path segment #3:

word #1: mno

word #2: pqr

word #3: stu

path segment #4:

word #1: 123

path segment #5:

word #1: 456

نام‌های منطقی بالا ممکن است به هر یک از «شناسه‌های داخلی» زیر نگاشت شوند:

Abc-Def/Ghi-Jkl/Mno-Pqr-Stu/123/456 # MIME or UNIX filesystem

abc_def.ghi_jkl.mno_pgr_stu[123][456] # C language

AbcDef\GhiJkl\MnoPqrStu\123\456 # Windows Registry

((abc def) (ghi jkl) (mno pqr stu) 123 456) # LISP

قراردادهای تعیین برای شناسه‌های پیمایش بیرونی، مستقل هستند و به طور خودکار به قراردادهای تعیین داخلی، نگاشت می‌شوند. مثلاً، قرارداد نام‌گذاری MIME ممکن است بصورت بیرونی استفاده شود ولی یک قرارداد زبان C بصورت داخلی استفاده می‌شود.

یادآوری - قراردادهای تعیین بالا فقط مثال هستند. قراردادهای تعیین از طریق مشخصات پارامتری، تعریف می‌شوند.

۶-۷ ویژگی `_value`

ویژگی `_value` شی داده، همان ارزش شی داده است. مثلاً، خواندن یا نوشتن مقدار "object"، با خواندن یا نوشتن مقدار ویژگی `_value` مرتبط با شی، یکسان است. یعنی، گرفتن `object`، همان گرفتن `object/_value` است.

۶-۸ واژه‌های کلیدی

واژه‌های کلیدی، شناسه‌هایی هستند که توسط پیشوند "._" شناسایی می‌شوند. واژه‌های کلیدی، معانی ویژه‌ای دارند که در زیر تعریف شده اند:

`_typesas` : مقداری را که به نوع مشخصی تبدیل شده است، برمی‌گرداند. مثلاً،
"zipcode/_typesas/integer"

value: : ویژگی "value" است که همانند دسترسی به خود شی داده است. مثلا، "foo/bar" با "foo/bar/._value" یکسان است.
prop: : فهرستی از ویژگی‌ها است. مثلا همه واژه‌های کلیدی "._".
type: : نوع داده شی است.
label: : قسمت برچسب شی است. مثلا، مقدار "foo/bar/._label" رشته "foobar" است. (منظور مقدار foobar نمی‌باشد)

۷ انقیادها

این استاندارد، مدل مفهومی مشترک و قواعد معناشناسی مشترک در میان تمام انقیادهای کدنویسی توصیف می‌کند. انقیاد کدنویسی انطباقی، باید مطابق با الزامات توصیف شده در بندهای ۴، ۵، ۶، ۸ و ۹ باشد.

۸ اداره

۸-۱ استفاده از انواع داده تعریف شده توسط مرکز ثبت^۱

یک پیاده‌سازی مطابق است که از انواع داده که توسط مرکز ثبت تعریف می‌شوند استفاده می‌کند. این پیاده‌سازی باید مطابق با الزامات آن مرکز ثبت (وابسته) باشد. زمان به‌روزرسانی برای مرکز ثبت وابسته باید در زمان پیاده‌سازی تعریف شوند.

مثال: برای یک پیاده‌سازی که از اقلام داده‌ای استفاده می‌کند که دامنه مقدارش بر اساس مرکز ثبت است، آن پیاده‌سازی باید چگونگی زمان‌بندی مورد انتظار برای اعمال تغییرات مرکز ثبت به پیاده‌سازی را مستندسازی کند (مثلا، اگر یک دامنه مقدار بر اساس کدهای کشور استاندارد ISO 3166-1 باشد و یک کد کشور جدید اضافه شود، زمان‌بندی مورد انتظار برای ترکیب کردن کد کشور در پیاده‌سازی چیست؟).

۹ انطباق

۹-۱ الگوی انطباق کدنویسی

الگوی انطباق برای استانداردهای ISO/IEC 20944 شامل نقش‌های انطباقی زیر می‌باشد: نمونه داده^۲، خواننده داده^۳، نویسنده داده^۴ و انباره داده^۵.

-
- 1 - Registry
 - 2 - Data Instance
 - 3 - Data Reader
 - 4 - Data Writer
 - 5 - Data Repository

۲-۹ انطباق نمونه داده

۲-۹-۱ نمونه داده

انطباق نمونه داده مستقل از انقیاد است.

یک نمونه داده بسیار منطبق باید مجموعه داده‌ای باشد که: (۱) مستقل از انقیاد ساخته شده است، (۲) بسیار منطبق با کارکرد، مدل مفهومی و قواعد نحوی مشخصات تبادل داده ارجاع شده است، (۳) باید شامل همه اقلام داده اجباری باشد، (۴) ممکن است شامل اقلام داده اختیاری باشد و (۵) نباید شامل اقلام داده توسعه‌یافته باشد.

یک نمونه داده منطبق باید مجموعه داده‌ای باشد که: (۱) مستقل از انقیاد ساخته شده است، (۲) منطبق با کارکرد، مدل مفهومی و قواعد نحوی مشخصات تبادل داده ارجاع شده است، (۳) باید شامل همه اقلام داده اجباری باشد، (۴) ممکن است شامل اقلام داده اختیاری باشد و (۵) نباید شامل اقلام داده توسعه‌یافته باشد. ارزیابی انطباق نمونه‌های داده باید توسط (۱) ترجمه نمونه داده در نحوه نوشتاری^۱ استاندارد ISO/IEC 11404 و (۲) بررسی الزاماتی که توسط مشخصات تبادل داده ارجاع شده توصیف شده است، انجام شود.

۲-۹-۲ نمونه داده مرز^۲

یک نمونه داده مرز دقیقاً منطبق باید (۱) یک نمونه داده دقیقاً منطبق باشد و (۲) دقیقاً منطبق با حداقل یک انقیاد کدنویسی باشد.

یک نمونه داده مرز منطبق باید (۱) یک نمونه داده منطبق باشد و (۲) منطبق با حداقل یک انقیاد کدنویسی باشد.

یادآوری- اختلاف بین نمونه داده SC/C، کدنویسی SC/C و نمونه داده مرزی SC/C عبارتست از: (۱) نمونه داده، نمونه‌ای از داده مستقل از انقیاد است، (۲) کدنویسی می‌تواند به نمونه‌ای از داده، مجموعه‌ای از نمونه‌های داده یا قواعد نحوی نمونه‌های داده اشاره کند و (۳) نمونه داده بسیار منطبق / نمونه داده مرز بسیار منطبق با انقیاد خاص، مرتبط است.

تعریف‌ها: پشتیبانی^۳، استفاده^۴

در محتوای انطباق، کلمات پشتیبانی و استفاده، در هر انقیاد کدنویسی بصورت جداگانه تعریف می‌شوند.

تعریف‌ها: آزمون^۵، دسترسی^۶، تفحص^۷

در محتوای انطباق، کلمات آزمون، دسترس و تفحص، به عنوان عملیات پوچ تعریف می‌شوند. یعنی برای انطباق نمونه داده، عملیات آزمون، دسترسی و تفحص هیچ عملیاتی انجام نمی‌دهند و تاثیری ندارند.

-
- 1 - Notation
 - 2 - Bound Data Instance
 - 3 - Support
 - 4 - Use
 - 5 - Test
 - 6 - Access
 - 7 - Probe

۳-۹ انطباق کاربرد داده

دو نوع انطباق کاربرد وجود دارد: دقیقاً منطبق و منطبق.

کاربرد داده دقیقاً منطبق، یک کاربرد داده است که کاملاً منطبق بر مشخصات تبادل داده ارجاع شده است.

یادآوری ۱- کاربرد داده دقیقاً منطبق باید حداقل منطبق اما حداکثر سازگار با مشخصات تبادل داده ارجاع شده باشد. انطباق دقیق مربوط است با (۱) ارزیابی، اندازه‌گیری و/یا قابلیت دسترسی مجموعه حداقل ویژگی‌ها؛ (۲) کاربرد داده بدون استفاده ویژگی تفحص و (۳) کاربرد داده بدون استفاده مجموعه‌های ویژگی توسعه یافته.

کاربرد داده منطبق، یک کاربرد داده است که منطبق بر مشخصات تبادل داده ارجاع شده است.

یادآوری ۲- کاربرد داده منطبق ممکن است مفیدتر باشد اما ممکن است کمتر سازگار با مشخصات تبادل داده ارجاع شده باشد. انطباق مربوط است با (۱) ارزیابی، اندازه‌گیری و/یا قابلیت دسترسی مجموعه حداقل ویژگی‌ها؛ (۲) ویژگی تفحص، طبق توافق قبلی با وجود (یا قابلیت دسترسی) ویژگی‌های توسعه یافته که توسط پیاده‌سازی مجاز است و (۳) ویژگی‌های توسعه یافته که برای این استاندارد به عنوان بیرونی تعیین شده است.

سه نوع کاربرد داده SC/C وجود دارد: خواننده داده، نویسنده داده و انباره داده.

۳-۹-۱ خواننده داده

خواننده داده دقیقاً مرتبط، باید داده‌ای را تفسیر کند^۱ که این داده، دقیقاً منطبق بر (۱) مشخصات تبادل داده ارجاع شده و (۲) حداقل یک انقیاد مشخصات تبادل داده ارجاع شده، است.

یادآوری ۱- خواننده داده دقیقاً مرتبط، اقلام داده توسعه یافته را تفسیر نمی‌کند.

یادآوری ۲- بسته به انقیاد مشخصات تبادل داده ارجاع شده، خواننده داده دقیقاً مرتبط، ممکن است توسعه‌های داده را نادیده بگیرد. به عنوان مثال، خواننده داده دقیقاً مرتبط ممکن است توسعه‌های داده را مصرف کند اما خواننده داده می‌تواند این توسعه‌ها را نادیده بگیرد (نه اینکه تفسیر کند)

خواننده داده دقیقاً، باید تفسیر کند داده‌ای را که منطبق است بر (۱) مشخصات تبادل داده ارجاع شده و (۲) حداقل یک انقیاد مشخصات تبادل داده ارجاع شده.

یادآوری ۳- خواننده داده مرتبط، ممکن است اقلام داده توسعه یافته را تفسیر کند.

۳-۹-۲ نویسنده داده

نویسنده داده دقیقاً مرتبط، باید داده‌ای را تولید کند^۲ که این داده، دقیقاً منطبق بر (۱) مشخصات تبادل داده ارجاع شده و (۲) حداقل یک انقیاد مشخصات تبادل داده ارجاع شده، است.

یادآوری ۱- نویسنده داده دقیقاً مرتبط، اقلام داده توسعه یافته را تولید نمی‌کند.

نویسنده داده دقیقاً، باید داده‌ای را تفسیر کند که این داده، منطبق بر (۱) مشخصات تبادل داده ارجاع شده و (۲) حداقل یک انقیاد مشخصات تبادل داده ارجاع شده، است.

1 - Interpret
2 - Generate

یادآوری ۲- نویسنده داده مرتبط، ممکن است اقلام داده توسعه یافته را تولید کند.

۳-۹-۳- انبار داده

انبار داده دقیقاً مرتبط، باید:

- نمونه‌های داده را برای بازیابی بعدی، دریافت کند.
- برای نمونه‌های داده دریافتی از تفسیر داده دقیقاً منطبق استفاده کند.
- نمونه‌های داده را در انبار ماندگار نگهداری کند بطوری که توسعه‌های داده ممکن است ماندگار نباشند.
- نمونه‌های داده قبلاً ذخیره شده را ارسال کند یا درخواست کند.
- برای ارسال نمونه‌های داده از تولید داده دقیقاً منطبق استفاده کند.
- برای حداقل یک انقیاد کدنویسی، دقیقاً انطباق دهد و
- برای حداقل یک انقیاد API یا انقیاد پروتکل، دقیقاً انطباق دهد.

یادآوری ۱- انبار داده دقیقاً مرتبط، الزامی به «حفظ» اقلام داده توسعه یافته ندارد. به عنوان مثال، تبادل داده نباید وابسته

به اقلام داده توسعه یافته مورد انتظار باشد که بر روی داده دقیقاً مرتبط اصرار دارد ولی آن را ممنوع هم نمی‌کند.

انبار داده مرتبط، باید:

- نمونه‌های داده را برای بازیابی بعدی، دریافت کند.
- برای نمونه‌های داده دریافتی از تفسیر داده منطبق استفاده کند.
- نمونه‌های داده را در انبار ماندگار نگهداری کند بطوری که توسعه‌های داده ممکن است ماندگار باشند.
- نمونه‌های داده قبلاً ذخیره شده را ارسال کند یا درخواست کند.
- برای ارسال نمونه‌های داده از تولید داده منطبق استفاده کند.
- برای حداقل یک انقیاد کدنویسی، انطباق دهد و
- برای حداقل یک انقیاد API یا انقیاد پروتکل، انطباق دهد.

یادآوری ۲- انبار داده منطبق ممکن است بر روی انبار، اقلام داده توسعه یافته را اضافه، حذف یا برای بازیابی بعدی تغییر دهد.

یادآوری ۳- انبار داده منطبق ممکن است برخی از توسعه‌های داده را ذخیره کند اما الزامی نیست که همه توسعه‌های داده را ذخیره و بازیابی کند.

یادآوری ۴- انبار داده منطبق ممکن است اشیا داده که نمونه‌های داده نیستند را ذخیره و بازیابی کند.

۴-۹-۴- برجسب‌های انطباق

برجسب‌های زیر انواع خاصی از انطباق با این بخش را نشان می‌دهد:

- الگوی ۱: "ISO/IEC 20944/B/prefix" که "prefix" نوع انقیاد را مشخص می‌کند همانطور که در بندهای ۱۱ و بعد از آن تعریف شده است، "ISO/IEC20944-2/B/XML" مطابق با انقیاد کدنویسی XML است.

- الگوی ۲: "ISO/IEC 20944-2/B/prefix/role" که "prefix" نوع انقیاد را مشخص می‌کند همانطور که در بندهای ۱۱ و بعد از آن تعریف شده است و "role" یکی از "D"، "R"، "W" یا "S" مطابق با نمونه داده، خواننده داده، نویسنده داده و انباره داده مطابق با تعریف این بند است. برای مثال، "ISO/IEC 20944-2/B/XML/S" مطابق با انباره داده با استفاده از مطابق با انقیاد کدنویسی XML است. محل قرار گرفتن انطباق برچسبها خارج از حوزه و دامنه کاربرد این استاندارد است.

۱۰ محفوظ برای استانداردسازی آینده

این بند برای استانداردسازی آینده خالی نگه داشته شده است.

۱۱ انقیاد کدنویسی DIVP

۱-۱۱ کلیات

انقیاد کدنویسی DIVP، نگاشت انواع داده به رشته‌های نویسه‌ای را تعریف می‌کند.

یادآوری - عبارت زیر استخراج شده، برگزیده و اقتباس شده و هماهنگ شده با استاندارد (HTTP/1.1) RFC 2068 است.

اندازه رکوردهای DIVP که تولید و مصرف می‌شوند، در زمان پیاده‌سازی تعریف می‌شوند. رکوردهایی که حداقل ۱۰۰۰۰۰ هشتتایی هستند، باید توسط پیاده‌سازی پشتیبانی گردند.

۱-۱۱-۱ ارقام واژه‌های^۱ پایه

نویسه خط جدید (CRLF) توسط کدنویسی آن، تعریف می‌شود.

یادآوری ۱- نویسه خط جدید ممکن است line-feed (مثلا Unix/Linux)، carriage-return (مثلا Macintosh) یا carriage-return (مثلا ترکیب line-feed) باشد.

برای حداکثر تعامل، پیاده‌سازی‌ها برای نویسه خط جدید هنگام تولید داده، باید ترکیبی از carriage-return و line-feed استفاده کنند. پیاده‌سازی‌هایی که فقط از یک نویسه، line-feed یا carriage-return، استفاده می‌کنند، هنگام مصرف داده برای نویسه خط جدید باید به ترتیب سایر نویسه‌های carriage-return یا line-feed را نادیده بگیرند.

فضای سفید پیشین (LWS)^۲ به عنوان حداقل یک یا تعداد بیشتری جای خالی^۳ یا نویسه تب^۴ در ابتدای خط تعریف می‌شود. مثلا اولین خط یا نویسه‌هایی که به دنبال خط جدید می‌آیند.

یادآوری ۲- LWS شامل خط جدید قبل نمی‌شود. ترتیبی از جای خالی یا نویسه‌های تب که در ابتدای خط نیستند، LWS نمی‌باشند.

1 - Lexical
2 - Leading Whitespace
3 - Space
4 - Tab

نویسه کنترل (CTL)^۱، نویسه‌ای در محدوده ۰-۳۱ (دهدهی) یا هشتایی ۱۲۷ (دهدهی) است ولی هشتایی ۹ (تب افقی (HT)^۲) نمی‌باشد.

نویسه‌های متنی، شامل همه نویسه‌ها به جز نویسه‌های کنترلی می‌باشد.

نویسه‌های زیر، نویسه‌های خاص نشانه^۳ هستند:

() < > @
, ; : \ "
/ [] ? =
{ } SP HT

یک نشانه^۴، یک یا چند نویسه است که شامل نویسه‌های کنترلی یا نویسه‌های نشانه خاص نمی‌باشد. رشته متنی، به عنوان یک نشانه واحد است (و نویسه‌های خاص نشانه، خاص نیستند) اگر در علامت دو نقل قول^۵ قرار بگیرد.

مثال ۱: نویسه‌های "[]" ، "abcd-" به عنوان یک نشانه واحد مصرف می‌شوند.

نویسه «» ممکن است برای علامت یک نقل قول^۶ استفاده شود. مثلاً، از بین بردن ماهیت خاص یک نویسه نویسه نشانه خاص، یک نویسه در هر زمان.

مثال ۲: نویسه‌های they're به عنوان یک نشانه واحد مصرف می‌شوند و نویسه یک نقل قول (')، معنی خاصی ندارد.

مثال ۳: نویسه‌های "say \"hello\"" به عنوان یک نشانه واحد مصرف می‌شوند، نشانه، قبل و بعد از کلمه hello، شامل نویسه دو نقل قول است و نویسه‌های نقل قول که کلمه hello را در برگرفته است، معنی خاصی ندارد.

۱۱-۱-۲ نام فیلد و مقدار فیلد

جفت نام-مقدار (NVP)^۷ باید شامل فیلد نام باشد که بعد از آن علامت دو نقطه^۸ «:» و بعد از آن مقدار فیلد فیلد قرار می‌گیرد. نام فیلد یک نشانه است. نام فیلدها باید به بزرگ و کوچک بودن حروف حساس باشند^۹.

یادآوری ۱- حساس بودن حروف در این انقیاد DVIP متفاوت از استاندارد RFC 822 است.

مقدار فیلد ممکن است توسط هر مقدار LWS مقدم شود. پیاده‌سازی‌های منطبق باید از یک فضای خالی واحد (SP)^{۱۰} به عنوان LWS استفاده کنند. مقدار فیلد صفر یا چندین نویسه است که شامل ترکیبی از نشانه‌ها و/یا نویسه‌های خاص نشانه می‌باشد.

- 1 - Control Character
- 2 - Horizontal Tab
- 3 - Token Special Characters
- 4 - Token
- 5 - Double-quote Marks
- 6 - Single-character Quoting
- 7 - Name-Value Pair
- 8 - Colon
- 9 - Case-sensitive
- 10 - Single Space

یادآوری ۲- جفت نام-مقدار در ابتدای خط شروع می‌شوند.

۱-۱۱-۳ فرآیند خط جدید

جفت نام-مقدار ممکن است روی چندین خط قبل از هر خط اضافه با حداقل یک SP یا HT توسعه یابد. همه فضاهای خالی خطی، قواعد معنانشناسی مشابه نویسه‌های SP دارند. خط جدید (CRLF) باید جفت نام-مقدار کامل را دنبال کند.

۱-۱۱-۴ خلاصه قواعد نحوی

یادآوری - خلاصه قواعد نحوی BNG ذیل، استخراج‌شده، برگزیده و اقتباس‌شده و هماهنگ‌شده با استاندارد RFC 2616 (HTTP/1.1) است.

CRLF	= <newline character>
CTL	= <any US-ASCII control character (characters 0 - 31) and DEL (127), except HT (9)>
LWS	= [CRLF] 1*(SP HT)
token	= 1*<any character except CTLs or tspecials>
tspecials	= "(" ")" "<" ">" "@" "," ";" ":" "\" <"> "/" "[" "]" "?" "=" "{" "}" SP HT
quoted-pair	= "\" character
quoted-string	= <">*<any char except non-quoted double quotes><">
message-header	= field-name ":" [field-value] CRLF
field-name	= token
field-value	= *(field-content LWS)
field-content	= <the characters making up the field-value and consisting of combinations of tokens or tspecials>

۱۱-۲ تولید^۱ و ایجاد^۲ DIVP

قواعد زیر تبدیل اقلام داده را به رکوردهای DIVP توصیف می‌کند، همانطور که در استانداردهای ISO/IEC 20944-2 و نوشتار استاندارد ISO/IEC 11404 توصیف شده است.

- قانون ۱: برای هر قلم داده، همه شناسه‌ها به نام فیلدهای کاملاً واجد شرایط^۳ نگاشت می‌شوند. نام کاملاً کاملاً واجد شرایط، ساختار تودرتوی قلم داده را نشان می‌دهد، همانطور که در «تولید نوع داده تجمیع» (استاندارد اصطلاحات ISO/IEC 11404) توصیف شده است. نقطه «.» باید هر مرحله از تودرتویی در یک نام فیلد کاملاً واجد شرایط را جدا کند. برای آرایه و تجمیع‌های ترتیبی، اقلام توسط نام‌های فیلد DIVP تکراری که مبتنی بر شناسه تجمیع است (نه اندیس قلم) نشان داده می‌شوند. نام فیلد که توسط دونقطه «.» دنبال می‌شود، توسط مقدار قلم داده مرتبطش دنبال می‌شود.

- قانون ۲: نام فیلدهای DIVP زیر را (نوشتار کلمات):
MDR_*

-
- 1 - Generating
 - 2 - Producing
 - 3 - Fully-qualified
 - 4 - Wildcard Notation

را به نام فیلدهای DIVP زیر تبدیل می‌کند (نوشتار کلمات):
ISO_IEC_11179_MDR_*
همه داده‌هایی که ایجاد می‌شوند باید به خوبی شکل گرفته^۱ DIVP باشند.

بنیاد و پایه^۲

مورد زیر بنیاد و پایه برای این دو قانون است.
باقیمانده این زیربنیاد اطلاعاتی است و الزامی نیست.

بنیاد و پایه برای قانون ۱

قانون ۱ تبدیل اصلی از انواع داده استاندارد ISO/IEC 11404 به نوشتار DIVP است. مثال‌های زیر برای روشن ساختن تبدیل‌ها از تعریف زیر استفاده می‌کند:

```
A: record
(
  B: integer,
  C: record
  (
    D: integer,
    E: characterstring(iso-10646),
  ),
  F_list: array (0..limit) of (integer),
  G: sample_mlstring_list_type,
)
```

سه جمله اول، «برای هر قلم داده، همه شناسه‌ها به نام فیلدهای کاملاً واجد شرایط^۳ نگاشت می‌شوند. نام کاملاً واجد شرایط، ساختار تودرتوی قلم داده را نشان می‌دهد، همانطور که در «تولید نوع داده تجمیع» (استاندارد اصطلاحات ISO/IEC 11404) توصیف شده است. نقطه «.» باید هر مرحله از تودرتویی در یک نام فیلد کاملاً واجد شرایط را جدا کند»، شناسه‌ها را به فیلدهای نام تبدیل می‌کند، مانند:

A.B

A.C.D

A.C.E

چهارمین جمله، «برای آرایه و تجمیع‌های ترتیبی، ارقام توسط نام‌های فیلد DIVP تکراری که مبتنی بر شناسه تجمیع است (نه اندیس قلم) نشان داده می‌شوند»، به آرایه‌ها و ترتیبات (فهرست‌ها) نیاز دارد که به عنوان چندین DIVPs با نام مشابه، نشان داده شود – مجاز در استاندارد RFC 822-like systems، اما با یک معنی کمی متفاوت (RFC 822) به این خطوط اجازه جستجو^۴ را می‌دهد، در حالی که این استاندارد تثبیت خودکار DIVP را اجازه نمی‌دهد. برای مثال، قلم داده F نشان داده می‌شود بصورت:

(انقیاد DIVP صحیح از F)

A.F: xxx

A.F: yyy

-
- 1 - Well-formed
 - 2 - Rationale
 - 3 - Fully-qualified
 - 4 - Comb

A.F: zzz

نه به صورت:

(انقیاد DIVP ناصحیح از F)

A.F.0: xxx

A.F.1: yyy

A.F.2: zzz

پنجمین جمله، «نام فیلد که توسط دونقطه «:» دنبال می‌شود، توسط مقدار قلم داده مرتبطش دنبال می‌شود»، فیلد نام را با مقدارش پیوند می‌زند و با دونقطه «:» از هم جدا می‌کند.

مثال:

A.B: 17

A.C.D: 34

A.C.E: yellow pigs

A.F: 51

A.F: 68

A.F: 85

بنیاد و پایه برای قانون ۲

این قانون برای دوباره‌نویسی برچسب‌ها جهت استفاده قراردادهای فضای نام^۱ خاص، به کار می‌رود.

۱۱-۳ مصرف و تفسیر DIVP

قواعد زیر تبدیل رکوردهای DIVP به اقلام داده را توصیف می‌کند، همانطور که در استانداردهای ISO/IEC 20944-2 و نوشتار استاندارد ISO/IEC 11404 توصیف شده است.

- قانون ۱: برچسب‌های DIVP زیر را (نوشتار کلمات):

ISO_IEC_11179_MDR_*

را به برچسب‌های DIVP زیر را تبدیل می‌کند (نوشتار کلمات):

MDR_*

- قانون ۲: برای هر نام فیلدی که مرتبط با شناسه‌ای است که توسط یک قلم داده در این قسمت تعریف شده است، هر نام فیلد به شناسه قلم داده مرتبط، نگاشت می‌شود. تودرتویی نام فیلدها، تودرتویی اقلام داده را نشان می‌دهد، یعنی برعکس عملیات قانون شماره ۱ از زیربند ۵-۱۱، تولید و ایجاد جفت‌های نام-مقدار نقطه‌گذاری شده. هر مقدار فیلد، به مقدار قلم داده مرتبط تبدیل شده است. مقدار فیلد خالی از یک تجمیع ممکن است وجود تجمیع را نشان دهد ولی نه مقدار مولفه‌اش.

بنیاد و پایه

مورد زیر بنیاد و پایه برای این دو قانون است.

باقیمانده این زیربند اطلاعاتی است و الزامی نیست.

بنیاد و پایه برای قانون ۱

این قانون در صورت لزوم، همه فضاهای نام را تبدیل می‌کند. در این شرح، پیشوند فضای کاری ("ISO_IEC_11179_MDR") برای کاهش احتمال تصادم فضاهای کاری استفاده شده است.

بنیاد و پایه برای قانون ۲

این قانون تبدیل‌های اصلی DIVPs به اقلام داده را مدیریت می‌کند. اولین جمله، «برای هر نام فیلدی که مرتبط با شناسه‌ای است که توسط یک قلم داده در این قسمت تعریف شده است، هر نام فیلد به شناسه قلم داده مرتبط، نگاشت می‌شود»، (۱) همه شناسه‌هایی که برای این راهنما ناشناخته است، نادیده می‌گیرد، (۲) به درستی نام فیلدها و شناسه‌های اقلام داده را جفت می‌کند و (۳) با اقلام داده ارتباط برقرار می‌کند ولی مقادیر اقلام داده را تخصیص نمی‌دهد.

دومین جمله، «تودرتویی نام فیلدها، تودرتویی اقلام داده را نشان می‌دهد، یعنی برعکس عملیات قانون شماره ۱ از زیربند ۵-۱۱، تولید و ایجاد جفت‌های نام-مقدار نقطه‌گذاری شده»، اطمینان می‌دهد که ساختار داخلی DIVPs، به میزان موردنیاز این راهنما، با ساختار داخلی اقلام داده توافق دارد. سومین جمله، «هر مقدار فیلد، به مقدار قلم داده مرتبط تبدیل شده است»، مقادیر فیلد اقلام داده را تبدیل می‌کند، یعنی اقلام داده را پر^۱ می‌کند.

چهارمین جمله، «مقدار فیلد خالی از یک تجمیع ممکن است وجود تجمیع را نشان دهد ولی نه مقدار مولفه‌اش»، فقط یک تجمیع را بوجود می‌آورد.

قطعه پایین، A.C، که مقدار خالی دارد، ممکن است برای شناسایی وجود یک تجمیع، استفاده شود:

A.B: 17

A.C:

A.C.D: 34

A.C.E: yellow pigs

مقدار خالی یک روش مناسب برای هنگامی که تجمیع‌ها شامل اقلام داده انتخابی هستند، یعنی علامت‌دهی^۲ وجود تجمیع ولی بدون مولفه‌های تجمیع:

A.B: 17

A.C:

مقدار خالی برای شناسایی نوع خالی^۳ استفاده نمی‌شود.

۴-۱۱ نمایش انواع داده پایه

زیربندهای زیر تبدیل مقادیر قلم داده به/از نمایش‌های نویسه‌ای برای تبادل اطلاعات به منظور استفاده در انقیاد DIVP، را توصیف می‌کنند.

۴-۱۱-۱ نویسه‌ها و رشته‌های نویسه

اقلام داده که از نوع نویسه هستند هنگامی که مطابق با قواعد استاندارد RFC 1522 کدنویسی می‌شوند، باید فقط مانند استاندارد ISO/IEC 8859-1 نمایش داده شوند.

1 - Populate
2 - Signaling
3 - Void

۴-۱۱-۲ اعداد صحیح

اقلام داده که از نوع صحیح هستند باید فقط مانند استاندارد ISO/IEC 9899:1999، زبان برنامه‌نویسی C، زیربند ۶-۴-۴-۱، ثابت‌های^۱ صحیح نمایش داده شوند؛ به استثنای پیشوندهای "U"، "L" و "LL" و انواع حروف کوچکشان است و ممکن است شامل علامت پیشین مثبت «+» یا منفی «-» ولی نه هر دو، باشد.

مثال:

```
0 // zero
23 // twenty-three
0x17 // same in hexadecimal
027 // same in octal
-34 // negative thirty-four
+34 // positive thirty-four
+34 // same
```

۴-۱۱-۳ اعداد حقیقی

اقلام داده که از نوع حقیقی هستند:

- اگر صحیح و کامل^۲ باشند، ممکن است همانطور که در بخش بالا ذکر شد، بصورت اعداد صحیح نمایش داده شوند.

- اگر صحیح و کامل نباشند یا همانند عدد صحیح نمایش داده نشوند، باید فقط مانند استاندارد ISO/IEC 9899:1999، زبان برنامه‌نویسی C، زیربند ۶-۴-۴-۲، ثابت‌های شناور^۳ صحیح نمایش داده شوند؛ به استثنای پیشوندهای "F" و "L" و انواع حروف کوچکشان است و ممکن است شامل علامت پیشین مثبت «+» یا منفی «-» ولی نه هر دو، باشد.

مثال:

```
0 // zero
0.0 // same
130.0 // one hundred thirty
1.3E2 // same
+1.3E2 // same
```

۴-۱۱-۴ مقادیر تاریخ و زمان

اقلام داده که از نوع زمان هستند باید فقط مانند استاندارد ISO/IEC 8601، ارقام داده و تبادل قالب‌ها-تبادل اطلاعات-نمایش تاریخ‌ها و زمان‌ها، نمایش داده شوند.

یادآوری ۱- استاندارد ISO 8601 تاریخ‌ها و زمان‌ها را از ۱ ژانویه ۲۰۰۱ تا ۳۱ دسامبر ۹۹۹۹ با استفاده از تقویم میلادی نمایش می‌دهد. کاملاً مشخص شده است که با این روش اختلالاتی وجود دارد. مثلاً، ماه سپتامبر ۱۷۵۲ کمتر از ۳۰ روز است، عدم ارتباط جغرافیایی^۴ قبل از معرفی خطوط راه آهن بین‌قاره‌ای، تغییر ماه آغازی تقویم، عدم توانایی نمایش تاریخ‌های قبل از عصر حاضر^۵، عدم توانایی نمایش تاریخ‌های بعد از ۳۱ دسامبر ۹۹۹۹.

- 1 - Constants
- 2 - Integral
- 3 - Floating Constants
- 4 - Timezones
- 5 - Before the Common Era

- یک تاریخ-زمان مطابق رخ نمون استاندارد ISO 8601:2000 زیر بیان می شود:
- یک تاریخ-زمان باید «نقطه زمانی»^۱ را نمایش دهد و نباید «بازه زمانی»^۲ یا زمان داخلی در محدوده زمانی معین^۳ را نمایش دهد.
 - یادآوری- به استاندارد ISO 8601 زیربندهای ۳-۸، ۳-۱۸، ۳-۲۶ و ۴-۱ مراجعه نمایید.
 - کوتاه‌سازی‌ها^۴ نباید استفاده شود.
 - یادآوری- به استاندارد ISO 8601 زیربند ۴-۶ مراجعه نمایید.
 - توسعه‌ها ممکن است برای نمایش سال‌های خارج از محدوده ۱ تا ۹۹۹۹ عصر حاضر استفاده شود (به استاندارد ISO 8601 زیربند ۴-۷ مراجعه نمایید)
 - یک تاریخ-زمان باید یک مولفه تاریخ (زیربند ۵-۲) یا ترکیبی از مولفه‌های تاریخ و زمان (زیربند ۵-۴) باشد.
 - یک ترکیب تاریخ-زمان باید نمایش کامل باشد (زیربند ۵-۴-۱، مثلا، "YYYY-MM-DDTHH:MM:SS±" یا نمایش سایر کامل (زیربند ۵-۴-۲، مثلا، "±YYYYYY-MM" یا "YYYY-MM-DDTHH:MM" ولی نه به صورت "MM-DD" یا "THH:MM:SS").
 - تاریخ‌ها باید همانند تاریخ‌های تقویم نمایش داده شوند و نباید مانند تاریخ‌های ترتیبی یا تاریخ‌های هفته.
 - یادآوری- به استاندارد ISO 8601 زیربند ۵-۲-۱ مراجعه نمایید.
 - تاریخ‌ها باید به صورت یکی از قالب‌های زیر نمایش داده شود:
 - تاریخ کامل در قالب توسعه‌یافته (زیربند ۵-۲-۱-۱، "YYYY-MM-DD")
 - تاریخ با دقت کاهش یافته برای یک ماه خاص در قالب اولیه (زیربند ۵-۲-۱-۲، "YYYY-MM")
 - تاریخ با دقت کاهش یافته برای یک سال خاص در قالب اولیه (زیربند ۵-۲-۱-۲، "YYYY")
 - تاریخ توسعه‌یافته برای روز خاص در قالب توسعه‌یافته (زیربند ۵-۲-۱-۴، "±YYYYYY-MM-DD")
 - تاریخ توسعه‌یافته برای ماه خاص در قالب توسعه‌یافته (زیربند ۵-۲-۱-۴، "±YYYYYY-MM")
 - تاریخ توسعه‌یافته برای سال خاص در قالب توسعه‌یافته (زیربند ۵-۲-۱-۴، "±YYYYYY")
 - زمان‌ها باید مانند زمان محلی (زیربند ۵-۳-۱) یا زمان هماهنگ جهانی^۵ (زیربند ۵-۳-۳) نمایش داده شود.

1 - Time-point
 2 - Time-interval
 3 - Recurring Time-interval
 4 - Truncations
 5 - Coordinated Universal Time

- زمان‌ها نباید از قالبی استفاده کند که تفاوت زمان محلی و زمان هماهنگ جهانی را نشان می‌دهد (زیربند ۵-۳-۴).
- زمان‌ها باید از نمایش کامل ("HH:MM:SS")، دقت کاهش یافته برای ساعت و دقیقه خاص ("HH:MM") و دقت کاهش یافته برای ساعت ("HH") خاص.
- زمان‌ها ممکن است از کسر اعشار استفاده کنند (زیربند ۵-۳-۱-۳، به عنوان مثال "HH:MM:SS.SSSS")
- نویسه نقطه «.» باید علامت اعشاری باشد.
- دقت تاریخ-زمان باید تاریخ-زمان‌هایی که از اعداد صحیح و کامل ثانیه‌ها استفاده می‌کنند پشتیبانی کند. مثال‌هایی از دقت تاریخ-زمان: ۱ ثانیه، ۱/۲ ثانیه، ۱/۶۰ ثانیه، ۱/۱۰۰ ثانیه. غیر منطبق: ۲ ثانیه، ۲/۳ ثانیه.
- یادآوری- این بدین معنی است که برنامه کاربردی باید دقت ثانیه دقیق را پشتیبانی کند اما ممکن است دقت‌های اعداد صحیح و کامل بخش‌هایی از ثانیه (مثلا 1/N ثانیه) را پشتیبانی کند.
- نقش‌دهنده زمان "T" باید هنگامی نمایش زمان، استفاده شود (زیربند ۵-۳-۱-۵).
- ساعت «۲۴» نباید برای نمایش نیمه شب به کار رود (زیربند ۵-۳-۲)

۱۱-۴-۵- نوع‌های خالی

یک نوع خالی نباید نه نمایش و نه کدنویسی داشته باشد.

مثال: رکورد زیر

```
A: record
(
  B: integer,
  C: void,
  D: characterstring(iso-10646),
)
```

در استاندارد RFC 822-like binding نمایش داده شده است:

```
(correct DIVP binding of C)
A.B: 17
A.D: hello
but not as:
(incorrect DIVP binding of C)
A.B: 17
A.C:
A.D: hello
```

۱۱-۵- کدنویسی نمایش‌های نویسه

انقیادهای کدنویسی DIVP باید نمایش‌های نویسه را به مقادیر نویسه‌ای مشخص شده توسط استاندارد ISO/IEC 8859-1 کدنویسی کنند. نویسه‌هایی که خارج از این فهرست نمایش^۱ هستند باید مطابق با قواعد استاندارد RFC 1522 کدنویسی شوند.

۱۱-۶ مدیریت استثناها و توسعه‌ها

۱۱-۶-۱ رفتار تعریف شده در هنگام پیاده‌سازی

موارد زیر رفتارهای تعریف شده در هنگام پیاده‌سازی هستند علاوه بر آنهایی که در جاهای دیگری از این راهنما توصیف شده‌اند.

موارد زیر رفتارهای تعریف شده در هنگام پیاده‌سازی در تولید و مصرف کدنویسی‌های DIVP هستند:

- کدنویسی نویسه خط جدید (در نویسه‌ها).

یادآوری - پیاده‌سازی‌ها می‌توانند از رفتار هنگام پیاده‌سازی بوسیله استفاده از نویسه‌های ترکیب carriage return line feed برای نویسه خط جدید جلوگیری کنند.

- حداکثر اندازه نمونه داده دقیقاً منطبق (در نویسه‌ها) که به عنوان DIVP کدنویسی می‌شود که می‌تواند به طور موفقیت‌آمیزی پردازش شود.

- اطلاعات جغرافیایی برای اقلام داده نوع زمان که جغرافیای نامشخص دارند.

۱۱-۶-۲ رفتار نامشخص

موارد زیر رفتارهای نامشخص هستند علاوه بر آنهایی که در جاهای دیگری از این راهنما توصیف شده‌اند.

موارد زیر رفتارهای نامشخص در هنگام پیاده‌سازی در تولید و مصرف کدنویسی‌های DIVP هستند:

- ترتیب پردازش اقلام داده.

۱۱-۶-۳ رفتار تعریف نشده

موارد زیر رفتارهای تعریف نشده هستند علاوه بر آنهایی که در جاهای دیگری از این راهنما توصیف شده‌اند.

موارد زیر رفتارهای تعریف نشده در هنگام پیاده‌سازی در تولید و مصرف کدنویسی‌های DIVP هستند:

- استفاده از نام فیلدهایی که با اقلام داده توسعه‌یافته مرتبط هستند.

- استفاده از نام فیلدهایی که با اقلام داده آماده به جانشین^۱ مرتبط هستند.

- استفاده از نام فیلدهایی که در این انقیاد کدنویسی DIVP مشخص نشده‌اند.

- استفاده از نویسه‌هایی که خارج از فهرست نمایش توصیف شده در این راهنما هستند.

۱۱-۷ پیشوند برچسب انطباق

پیشوند DIVP باید در رابطه با زیربند ۹-۴ برای نشان دادن انطباق با این زیربند، استفاده شود.

۱-۱۲ کلیات

اندازه رکوردهای XML که تولید و مصرف می‌شوند، در زمان پیاده‌سازی تعریف می‌شوند. رکوردهایی که حداقل ۱۰۰۰۰۰ هشتایی هستند، باید توسط پیاده‌سازی پشتیبانی گردند.

۱۲-۲ تولید و ایجاد XML

قواعد زیر تبدیل اقلام داده را به رکوردهای XML توصیف می‌کند، همانطور که در استانداردهای ISO/IEC 20944-2 و نوشتار استاندارد ISO/IEC 11404 توصیف شده است.

- قانون ۱: هر داده در نوشتار استاندارد ISO/IEC 11404، همه شناسه‌ها را به برچسب‌های XML نگاشت می‌کند، به جز مواردی که در قانون ۲ ذکر شده است. برچسب‌های XML متوازن رمز مقادیر مرتبط با قلم داده را محدود می‌کند. تودرتویی برچسب‌های XML، ساختار اقلام داده را نشان می‌دهد، همانطور که در «تولید نوع داده تجمیع» (استاندارد اصطلاحات ISO/IEC 11404) توصیف شده است. برای آرایه و تجمیع‌های ترتیبی، (۱) برچسب XML نام مشابه، به عنوان شناسه تجمیع، گروهی از تجمیع‌ها را نشان می‌دهد، (۲) اقلام داده‌های شخصی توسط برچسب‌های XML تکراری که مبتنی بر شناسه تجمیع منهای پسوند “_list” یا “_bucket” است (نه اندیس قلم) نشان داده می‌شوند.

- قانون ۲: همه انواع داده mcstring_type را به موارد زیر نگاشت می‌کند.

- قانون ۲الف: قلم محلی mcstring_type که صفت ^۱ LANG در قلم XML والد را تنظیم می‌کند.

- قانون ۲ب: قلم رشته که محتوای برچسب والد را تنظیم می‌کند (یعنی مقصد جاری)

- قانون ۳: برچسب‌های XML زیر را (نوشتار کلمات):

MDR_*

را به برچسب‌های XML زیر تبدیل می‌کند (نوشتار کلمات):

ISO_IEC_11179_MDR_*

همه داده‌هایی که ایجاد می‌شوند باید به خوبی شکل گرفته XML باشند.

یادآوری - استفاده از روش‌های متنی بر XML مانند طرح XML^۲ و RELAX-NG، مجاز است ولی الزام نیست.

بنیاد و پایه

مورد زیر بنیاد و پایه برای این سه قانون است.

باقیمانده این زیربنای اطلاعاتی است و الزامی نیست.

بنیاد و پایه برای قانون ۱

قانون ۱ تبدیل اصلی از انواع داده استاندارد ISO/IEC 11404 به قراردادهای برچسب‌گذاری XML است. مثال‌های زیر برای روشن ساختن تبدیل‌ها از تعریف زیر استفاده می‌کند:

A: record

1 - Attribute

2 - XML Schema


```
(
  B: integer,
  C: record
  (
    D: integer,
    E: characterstring(iso-10646),
  ),
  F_list: array (0..limit) of (integer),
  G: sample_mlstring_list_type,
)
```

جمله اول، «هر داده در نوشتار استاندارد ISO/IEC 11404، همه شناسه‌ها را به برچسب‌های XML نگاشت می‌کند»، شناسه‌ها را تبدیل می‌کند. مثلاً، "X:"->"X>"

دومین جمله، «برچسب‌های XML متوازن مرز مقادیر مرتبط با قلم داده را محدود می‌کند»، مستلزم آن است که (۱) برچسب‌ها متوازن باشند و (۲) مقدار قلم داده بین برچسب‌ها باشد. مثلاً، "X: 17">"<X>17</X>"

سومین جمله، «تودرتویی برچسب‌های XML، ساختار اقلام داده را نشان می‌دهد، همانطور که در تولید نوع داده تجمیع توصیف شده است»، مستلزم آن است که تودرتویی اشاره شده در تجمیع‌ها (رکوردها، آرایه‌ها، ترتیب‌ها/ فهرست‌ها) منتج به تودرتویی مشابه با برچسب‌های XML می‌شود. استفاده از تعریف الف، تودرتویی زیر برای اقلام B, C, D و E به کار می‌رود:

```
<A>
  <B>...</B>
  <C>
    <D>...</D>
    <E>...</E>
  </C>
  ...
</A>
```

چهارمین جمله، «برای آرایه و تجمیع‌های ترتیبی، اقلام داده‌ها توسط برچسب‌های XML تکراری که مبتنی بر شناسه تجمیع (نه اندیس قلم) نشان داده می‌شوند»، مستلزم آن است که آرایه‌ها و ترتیب (فهرست)ها، مانند چندین برچسب با نام مشابه نمایش داده شوند – قرارداد سبک XML معمولی. مثلاً، قلم داده F بصورت زیر نشان داده می‌شود:

```
<!-- correct XML binding of F_list -->
<A>
  ...
  <F_list>
    <F>...</F>
    <F>...</F>
    <F>...</F>
  </F_list>
  ...
</A>
```

نه به صورت:

```
<!-- incorrect XML binding of F_list -->
<A>
```

```

...
<F_list>
    <0>...</0>
    <1>...</1>
    <2>...</2>
</F_list>
...
</A>

```

بنیاد و پایه برای قانون ۲

برخی از رکوردها از چندین نوع داده ویژه استفاده می‌کنند، مثل انواع داده چندزبانه برای توصیف اقلام داده نوع نویسه رشته‌ای معین که باید در محتوای چندزبانه و چندفرهنگی نمایش داده شود که معمولا ویژگی‌های بین‌المللی‌سازی (L 10N) و محلی‌سازی (I18N) نامیده می‌شود. قسمت زیر، یک نسخه مثال از نوع داده چندزبانه است که با تعریف‌های سایر انواع داده چندزبانه که در دیگر قسمت‌های این استاندارد تعریف شده است، تضادی ندارد. در این تصویر، نوع داده `sample_mcstring_type`، یک جفت ساده را نمایش می‌دهد: یک رشته محلی شده و یک مشخصات محل (نگاشت L 10N). نوع داده `sample_mcstring_array_type`، آرایه‌ای از ای جفت‌های رشته‌ای نمایش می‌دهد. در این مثال، آرایه `example_remarks`، شامل سه قلم است که هر قلم، یک جفت رشته است. احتمالا یک برنامه کاربردی رشته مناسبی از استفاده `example_remarks` مبتنی بر کشور (محل)ی که برنامه کاربردی در آن محل عملیاتی شده است، انتخاب می‌کند. قسمت زیر، تعریف‌های نوع مثال و تعریف‌های مقدار، می‌باشد.

```

type sample_mcstring_type =
record
(
    L10N_string: characterstring(iso-10646),
    L10N_locale: string_type,
),
type sample_mcstring_array_type =
array (0..limit) of (sample_mcstring_type),

value example_remarks:
sample_mcstring_array_type =
(
(
    L10N_string: "abc abc abc",
    L10N_locale: "en-US",
),
(
    L10N_string: "def def def",
    L10N_map: "fr-CA",
),
(
    L10N_string: "ghi ghi ghi",
    L10N_map: "de-DE",
),
),
),

```

قانون ۲ به همراه مدیریت قانون ۱، این اقلام داده را به XML زیر تبدیل می‌کند:

```
<example_remarks LANG="en-US">abc abc abc</example_remarks>  
<example_remarks LANG="fr-CA">def def def</example_remarks>  
<example_remarks LANG="de-DE">ghi ghi ghi</example_remarks>
```

بنیاد و پایه برای قانون ۳

این قانون برای بازنویسی برچسب‌ها به منظور تبدیل‌های فضای نام معین استفاده می‌شود. این قانون می‌تواند فضاهای نام XML مشخص توسط انتخاب تبدیل فضای نام متفاوت (پیشوندها)، داشته باشد. بعد از قانون ۳، پیاده‌سازی مورد نیاز است تا اطمینان حاصل شود که نتیجه این تبدیل‌ها، به خوبی شکل گرفته XML است.

۱۲-۳ مصرف و تفسیر XML

قواعد زیر تبدیل رکوردهای XML به اقلام داده را توصیف می‌کند، همانطور که در استانداردهای ISO/IEC 20944-2 و نوشتار استاندارد ISO/IEC 11404 توصیف شده است.

- قانون ۱: برچسب‌های XML زیر را (نوشتار کلمات):
ISO_IEC_11179_MDR_*
را به برچسب‌های XML زیر را تبدیل می‌کند (نوشتار کلمات):
MDR_*
- قانون ۲: تبدیل به شرح زیر است:
- قانون ۲الف: ویژگی LANG از قلم XML، قلم محل مربوط به قلم داده mcstring_type را تنظیم می‌کند.
- قانون ۲ب: محتویات قلم برچسب‌زده، قلم رشته مربوط به قلم داده mcstring_type را تنظیم می‌کند.
- قانون ۳: برای هر یک از برچسب‌های XML (که توسط یک قلم داده این استاندارد با یک شناسه مرتبط است)، برچسب‌های XML متوازن باز و بسته کردن مربوطه، با هم تطابق دارند. برای هر یک از برچسب‌های XML، به استثنای مواردی که در قانون ۲ اصلاح شد، هر برچسپ XML به شناسه قلم داده مربوطه نگاشت می‌کند. برچسب‌های XML تودرتو، اقلام داده تودرتو را نمایش می‌دهد. به عنوان مثال، ذخیره عملیات در قانون شماره ۱ زیربند ۵-۱، تولید و ایجاد XML. محتوای هر قلم برچسب زده شده، به مقدار قلم داده مرتبط تبدیل می‌شود.

بنیاد و پایه

مورد زیر بنیاد و پایه برای این سه قانون است. باقیمانده این زیربند اطلاعاتی است و الزامی نیست.

بنیاد و پایه برای قانون ۱

قبل از پردازش، پیاده‌سازی اطمینان می‌دهد که مصرف و تفسیر به خوبی شکل گرفته XML هستند. این قانون پیشوندها و پسوندهای فضای کاری XML را در صورت نیاز پاک می‌کند. در این تصویر، فضاهای نام XML استفاده نمی‌شوند اما پیشوند فضای کاری ("ISO_IEC_11179_MDR") برای کاهش احتمال تصادم فضاهای کاری استفاده شده است.

بنیاد و پایه برای قانون ۲

این قانون، نگاشت را از ویژگی LANG به نوع داده mlstring_type، ذخیره نمی‌کند. این قانون مواظب است که فقط اقلام داده mcstring_type شناخته شده را تبدیل کند، زیرا سایر ویژگی‌های XML LANG، مرتبط با اقلام داده mcstring_type در این استاندارد نیستند.

بنیاد و پایه برای قانون ۳

این قانون تبدیل‌های اصلی برچسب‌های XML و محتویاتشان به اقلام داده را مدیریت می‌کند. اولین جمله، «برای هریک از برچسب‌های XML (که توسط یک قلم داده این استاندارد با یک شناسه مرتبط است)، برچسب‌های XML متوازن باز و بسته کردن مربوطه، با هم تطابق دارند»، (۱) همه شناسه‌هایی که برای این راهنما ناشناخته است، نادیده می‌گیرد، (۲) به درستی آن‌ها را جفت می‌کند. دومین جمله، «برای هریک از برچسب‌های XML، به استثنای مواردی که در قانون ۲ اصلاح شد، هر برچسپ XML به شناسه قلم داده مربوطه نگاشت می‌کند»، ارتباط با اقلام داده را ایجاد می‌کند ولی مقادیر اقلام داده را تخصیص نمی‌دهد.

سومین جمله، «برچسب‌های XML تودرتو، اقلام داده تودرتو را نمایش می‌دهد. به عنوان مثال، ذخیره عملیات در قانون شماره ۱ زیربند ۵-۱، تولید و ایجاد XML»، اطمینان می‌دهد که ساختار داخلی XML، به میزان موردنیاز این راهنما، با ساختار داخلی اقلام داده توافق دارد. چهارمین جمله، «محتوای هر قلم برچسب زده شده، به مقدار قلم داده مرتبط تبدیل می‌شود»، محتوای برچسب‌های XML را به مقادیر اقلام داده تبدیل می‌کند، یعنی اقلام داده را پر می‌کند.

۴-۱۲ نمایش انواع داده پایه

زیربندهای زیر تبدیل مقادیر قلم داده به/از نمایش‌های نویسه‌ای برای تبادل اطلاعات به منظور استفاده در انقیاد XML، را توصیف می‌کنند.

۴-۱۲-۱ نویسه‌ها و رشته‌های نویسه

اقلام داده که از نوع نویسه هستند باید فقط مانند مشخصات XML نمایش داده شوند.

یادآوری ۱- نویسه‌های خاص مانند “>” “<” “&” نیاز به روش‌های ترجمه دارند و البته ترجمه‌ای که باعث زیان نشود.

یادآوری ۲- برخی از کدنویسی‌ها مثل استانداردهای ISO-8859-1 و UTF-8، کدنویسی مستقیم از نمایش نویسه‌هایی مانند “©” (نشانه قانون حق نسخه برداری) را اجازه می‌دهند. بعد از کدنویسی‌ها، مانند ASCII، توسعه‌های کدنویسی، مانند “#169” جهت نمایش این نشانه‌ها، لازم است.

۴-۱۲-۲ اعداد صحیح

اقلام داده که از نوع صحیح هستند باید فقط مانند استاندارد ISO/IEC 9899:1999، زبان برنامه‌نویسی C، زیربند ۴-۴-۶، ثابت‌های صحیح نمایش داده شوند؛ به استثنای پیشوندهای “U”، “L” و “LL” و انواع حروف کوچکشان است و ممکن است شامل علامت پیشین مثبت “+» یا منفی «-» ولی نه هر دو، باشد.

مثال:

0 // zero
23 // twenty-three
0x17 // same in hexadecimal
027 // same in octal
-34 // negative thirty-four
+34 // positive thirty-four
+34 // same

۴-۱۲-۳ اعداد حقیقی

اقدام داده که از نوع حقیقی هستند:

- اگر صحیح و کامل باشند، ممکن است همانطور که در بخش بالا ذکر شد، بصورت اعداد صحیح نمایش داده شوند.
- اگر صحیح و کامل نباشند یا همانند عدد صحیح نمایش داده نشوند، باید فقط مانند استاندارد ISO/IEC 9899:1999، زبان برنامه‌نویسی C، زیربند ۶-۴-۲، ثابت‌های شناور صحیح نمایش داده شوند؛ به استثنای پیشوندهای "F" و "L" و انواع حروف کوچکشان است و ممکن است شامل علامت پیشین مثبت «+» یا منفی «-» ولی نه هر دو، باشد.

مثال:

0 // zero
0.0 // same
130.0 // one hundred thirty
1.3E2 // same
+1.3E2 // same

۴-۱۲-۴ مقادیر تاریخ و زمان

اقدام داده که از نوع زمان هستند باید فقط مانند استاندارد ISO/IEC 8601، اقدام داده و تبادل قالب‌ها-تبادل اطلاعات-نمایش تاریخ‌ها و زمان‌ها، نمایش داده شوند.

یادآوری ۱- استاندارد ISO 8601 تاریخ‌ها و زمان‌ها را از ۱ ژانویه ۲۰۰۱ تا ۳۱ دسامبر ۹۹۹۹ با استفاده از تقویم میلادی نمایش می‌دهد. کاملاً مشخص شده است که با این روش اختلالاتی وجود دارد. مثلاً، ماه سپتامبر ۱۷۵۲ کمتر از ۳۰ روز است، عدم ارتباط جغرافیایی قبل از معرفی خطوط راه آهن بین‌قاره‌ای، تغییر ماه آغازی تقویم، عدم توانایی نمایش تاریخ‌های قبل از عصر حاضر، عدم توانایی نمایش تاریخ‌های بعد از ۳۱ دسامبر ۹۹۹۹.

یک تاریخ-زمان مطابق رخ نمون استاندارد ISO 8601:2000 زیر بیان می‌شود:

- یک تاریخ-زمان باید «نقطه زمانی» را نمایش دهد و نباید «بازه زمانی» یا زمان داخلی در محدوده زمانی معین را نمایش دهد.

یادآوری - به استاندارد ISO 8601 زیربندهای ۳-۸، ۳-۱۸، ۳-۲۶ و ۴-۱ مراجعه نمایید.

- کوتاه‌سازی‌ها نباید استفاده شود.

یادآوری - به استاندارد ISO 8601 زیربند ۴-۶ مراجعه نمایید.

- توسعه‌ها ممکن است برای نمایش سال‌های خارج از محدوده ۱ تا ۹۹۹۹ عصر حاضر استفاده شود (به استاندارد ISO 8601 زیربند ۴-۷ مراجعه نمایید)
- یک تاریخ-زمان باید یک مولفه تاریخ (زیربند ۵-۲) یا ترکیبی از مولفه‌های تاریخ و زمان (زیربند ۵-۴) باشد.
- یک ترکیب تاریخ-زمان باید نمایش کامل باشد (زیربند ۵-۴-۱، مثلا، "YYYY-MM-DDTHH:MM:SS" یا "±YYYYYY-MM-DDTHH:MM:SS") یا نمایش سایر کامل (زیربند ۵-۴-۲، مثلا، "±YYYYYY-MM" یا "YYYY-MM-DDTHH:MM" ولی نه به صورت "MM-DD" یا "THH:MM:SS").
- تاریخ‌ها باید همانند تاریخ‌های تقویم نمایش داده شوند و نباید مانند تاریخ‌های ترتیبی یا تاریخ‌های هفته.
- یادآوری - به استاندارد ISO 8601 زیربند ۵-۲-۱ مراجعه نمایید.
- تاریخ‌ها باید به صورت یکی از قالب‌های زیر نمایش داده شود:
- تاریخ کامل در قالب توسعه‌یافته (زیربند ۵-۲-۱-۱، "YYYY-MM-DD")
- تاریخ با دقت کاهش یافته برای یک ماه خاص در قالب اولیه (زیربند ۵-۲-۱-۲، "YYYY-MM")
- تاریخ با دقت کاهش یافته برای یک سال خاص در قالب اولیه (زیربند ۵-۲-۱-۲، "YYYY")
- تاریخ توسعه‌یافته برای روز خاص در قالب توسعه‌یافته (زیربند ۵-۲-۱-۴، "±YYYYYY-MM-DD")
- تاریخ توسعه‌یافته برای ماه خاص در قالب توسعه‌یافته (زیربند ۵-۲-۱-۴، "±YYYYYY-MM")
- تاریخ توسعه‌یافته برای سال خاص در قالب توسعه‌یافته (زیربند ۵-۲-۱-۴، "±YYYYYY")
- زمان‌ها باید مانند زمان محلی (زیربند ۵-۳-۱) یا زمان هماهنگ جهانی (زیربند ۵-۳-۳) نمایش داده شود.
- زمان‌ها نباید از قالبی استفاده کند که تفاوت زمان محلی و زمان هماهنگ جهانی را نشان می‌دهد (زیربند ۵-۳-۴).
- زمان‌ها باید از نمایش کامل ("HH:MM:SS")، دقت کاهش یافته برای ساعت و دقیقه خاص ("HH:MM") و دقت کاهش یافته برای ساعت ("HH") خاص.
- زمان‌ها ممکن است از کسر اعشار استفاده کنند (زیربند ۵-۳-۱-۳، به عنوان مثال "HH:MM:SS.SSSS")
- نویسه نقطه «.» باید علامت اعشاری باشد.
- دقت تاریخ-زمان باید تاریخ-زمان‌هایی که از اعداد صحیح و کامل ثانیه‌ها استفاده می‌کنند پشتیبانی کند. مثال‌هایی از دقت تاریخ-زمان: ۱ ثانیه، ۱/۲ ثانیه، ۱/۶۰ ثانیه، ۱/۱۰۰ ثانیه. غیر منطبق: ۲ ثانیه، ۲/۳ ثانیه.
- یادآوری - این بدین معنی است که برنامه کاربردی باید دقت ثانیه دقیق را پشتیبانی کند اما ممکن است دقت‌های اعداد صحیح و کامل بخش‌هایی از ثانیه (مثلا 1/N ثانیه) را پشتیبانی کند.

- نقش دهنده زمان "T" باید هنگامی نمایش زمان، استفاده شود (زیربند ۵-۳-۱-۵).
- ساعت «۲۴» نباید برای نمایش نیمه شب به کار رود (زیربند ۵-۳-۲)

۴-۱۲-۵ نوع‌های خالی

یک نوع خالی نباید نه نمایش و نه کدنویسی داشته باشد.

مثال: رکورد زیر

```
A: record
(
  B: integer,
  C: void,
  D: characterstring(iso-10646),
)
```

در XML به صورت زیر نمایش داده می‌شود:

```
<!-- correct XML representation of C -->
<A>
  <B>17</B>
  <D>hello</D>
</A>
```

نه به صورت:

```
<!-- incorrect XML representation of C -->
<A>
  <B>17</B>
  <C></C>
  <D>hello</D>
</A>
```

۵-۱۲ کدنویسی نمایش‌های نویسه

کدنویسی نمایش نویسه‌ها به مقادیر هشتایی توسط روش کدنویسی XML مشخص می‌شود. انطباق نمونه‌های داده با انقیاد کدنویسی XML، باید در یکی از موارد زیر کدنویسی شود: ASCII، استاندارد ISO/IEC 8859-1، UTF-8 یا استاندارد ISO/IEC 10646 یا UTF-16 از استاندارد ISO/IEC 10646. انطباق کاربردهای داده با انقیاد کدنویسی XML، باید همه کدنویسی‌های زیر را پشتیبانی کند: ASCII، استاندارد ISO/IEC 8859-1، UTF-8 یا استاندارد ISO/IEC 10646 یا UTF-16 از استاندارد ISO/IEC 10646.

۶-۱۲ مدیریت استثناها و توسعه‌ها

۶-۱۲-۱ رفتار تعریف شده در هنگام پیاده‌سازی

موارد زیر رفتارهای تعریف شده در هنگام پیاده‌سازی هستند علاوه بر آن‌هایی که در جاهای دیگری از این راهنما توصیف شده‌اند.

موارد زیر رفتارهای تعریف شده در هنگام پیاده‌سازی در تولید و مصرف کدنویسی‌های XML هستند:

- حداکثر اندازه (به صورت هشتایی) نمونه داده دقیقاً منطبق، که به عنوان XML کدنویسی می‌شود که ممکن است بصورت موفقیت آمیزی پردازش شود.
- عمق تودرتویی حداکثر رکوردهای XML.

- اطلاعات جغرافیایی برای اقلام داده نوع زمان که جغرافیای نامشخص دارند.

۱۲-۶-۲ رفتار نامشخص

موارد زیر رفتارهای نامشخص هستند علاوه بر آنهایی که در جاهای دیگری از این راهنما توصیف شده‌اند.

موارد زیر رفتارهای نامشخص در تولید یا تفسیر کدنویسی‌های XML هستند:

- ترتیب پردازش اقلام داده.

موارد زیر رفتارهای نامشخص در ایجاد یا مصرف کدنویسی‌های XML هستند:

- استفاده از نویسه‌های اضافی فضای سفید، خارج از مقادیر قلم داده و خارج از آنهایی که توسط مشخصات XML موردنیاز است.

۱۲-۶-۳ رفتار تعریف نشده

موارد زیر رفتارهای تعریف نشده هستند علاوه بر آنهایی که در جاهای دیگری از این راهنما توصیف شده‌اند.

موارد زیر رفتارهای تعریف نشده در هنگام پیاده‌سازی در تولید و مصرف کدنویسی‌های DIVP هستند:

- استفاده از برچسب‌های XML که با اقلام داده توسعه‌یافته مرتبط هستند.

- استفاده از برچسب‌های XML که با اقلام داده ذخیره‌شده مرتبط هستند.

- استفاده از برچسب‌های XML یا ویژگی‌هایی که در این انقیاد کدنویسی XML مشخص نشده‌اند.

- استفاده از نویسه‌هایی که خارج از فهرست نمایش توصیف شده در این استاندارد هستند.

۱۲-۷ پیشوند برچسب انطباق

پیشوند XML باید در رابطه با زیربند ۹-۴ برای نشان دادن انطباق با این زیربند، استفاده شود.

کتابنامه

- [1] ISO 3166-1, Codes for the representation of names of countries and their subdivisions — Part 1: Country codes
- [2] ISO 8601:2000, Data elements and interchange formats — Information interchange — Representation of dates and times
- [3] ISO/IEC 8859-1, Information technology — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1
- [4] ISO/IEC 9899:1999, Programming languages — C
- [5] ISO/IEC 10646, Information technology — Universal Coded Character Set (UCS)
- [6] ISO/IEC 11179 (all parts), Information technology — Metadata registries (MDR)
- [7] RFC 822, Standard for the Format of ARPA Internet Text Messages, August 1982
- [8] RFC 1522, MIME (Multipurpose Internet Mail Extensions) Part Two: Message Header Extensions for Non-ASCII Text, September 1993
- [9] RFC 2616, Hypertext Transfer Protocol — HTTP/1.1, June 1999