

МСЭ-R

Сектор радиосвязи МСЭ

Рекомендация МСЭ-R ВТ.1563-1
(03/2011)

**Протокол кодирования данных
с использованием значения длины ключа**

**Серия ВТ
Радиовещательная служба
(телевизионная)**



Предисловие

Роль Сектора радиосвязи заключается в обеспечении рационального, справедливого, эффективного и экономичного использования радиочастотного спектра всеми службами радиосвязи, включая спутниковые службы, и проведении в неограниченном частотном диапазоне исследований, на основании которых принимаются Рекомендации.

Всемирные и региональные конференции радиосвязи и ассамблеи радиосвязи при поддержке исследовательских комиссий выполняют регламентарную и политическую функции Сектора радиосвязи.

Политика в области прав интеллектуальной собственности (ПИС)

Политика МСЭ-R в области ПИС излагается в общей патентной политике МСЭ-T/МСЭ-R/ИСО/МЭК, упоминаемой в Приложении 1 к Резолюции 1 МСЭ-R. Формы, которые владельцам патентов следует использовать для представления патентных заявлений и деклараций о лицензировании, представлены по адресу: <http://www.itu.int/ITU-R/go/patents/en>, где также содержатся Руководящие принципы по выполнению общей патентной политики МСЭ-T/МСЭ-R/ИСО/МЭК и база данных патентной информации МСЭ-R.

Серии Рекомендаций МСЭ-R

(Представлены также в онлайн-форме по адресу: <http://www.itu.int/publ/R-REC/en>.)

Серия	Название
BO	Спутниковое радиовещание
BR	Запись для производства, архивирования и воспроизведения; пленки для телевидения
BS	Радиовещательная служба (звуковая)
BT	Радиовещательная служба (телевизионная)
F	Фиксированная служба
M	Подвижная спутниковая служба, спутниковая служба радиоопределения, любительская спутниковая служба и относящиеся к ним спутниковые службы
P	Распространение радиоволн
RA	Радиоастрономия
RS	Системы дистанционного зондирования
S	Фиксированная спутниковая служба
SA	Космические применения и метеорология
SF	Совместное использование частот и координация между системами фиксированной спутниковой службы и фиксированной службы
SM	Управление использованием спектра
SNG	Спутниковый сбор новостей
TF	Передача сигналов времени и эталонных частот
V	Словарь и связанные с ним вопросы

Примечание. – Настоящая Рекомендация МСЭ-R утверждена на английском языке в соответствии с процедурой, изложенной в Резолюции 1 МСЭ-R.

Электронная публикация
Женева, 2011 г.

© ITU 2011

Все права сохранены. Ни одна из частей данной публикации не может быть воспроизведена с помощью каких бы то ни было средств без предварительного письменного разрешения МСЭ.

РЕКОМЕНДАЦИЯ МСЭ-R ВТ.1563-1

Протокол кодирования данных с использованием значения длины ключа

(Вопрос МСЭ-R 130/6)

(2002-2011)

Сфера применения

В настоящей Рекомендации определяется протокол кодирования данных на уровне байтов, предназначенный для представления элементов данных и групп данных. Данный протокол определяет структуру данных, которая не зависит от используемого метода применения или транспортировки.

В Рекомендации определяется триплет "ключ-длина-значение" (KLV – key-length-value) в качестве протокола обмена данными применительно к элементам данных или группам данных, где Key идентифицирует данные, Length специфицирует длину данных, а Value – это собственно данные. Протокол KLV предоставляет единый механизм обмена для всех совместимых приложений, не зависящий от способа реализации или транспортировки.

Ассамблея радиосвязи МСЭ,

учитывая,

- a) что многие страны создали производственные комплексы цифрового телевидения, основанные на использовании цифровых компонентов видеосигналов, соответствующих Рекомендациям МСЭ-R ВТ.601, МСЭ-R ВТ.656 и МСЭ-R ВТ.799;
- b) что были созданы производственные системы телевидения высокой четкости (ТВЧ) с использованием цифровых интерфейсов ТВЧ, соответствующих Рекомендации МСЭ-R ВТ.1120;
- c) что сигнал, соответствующий Рекомендации МСЭ-R ВТ.656 или МСЭ-R ВТ.799, обладает определенной пропускной способностью, позволяющей мультиплексировать сигналы дополнительных данных в последовательном потоке данных;
- d) что мультиплексирование сигналов вспомогательных данных в последовательном потоке данных обеспечивает эксплуатационные и экономические выгоды;
- e) что эксплуатационные выгоды возрастают, если для сигналов вспомогательных данных используется минимальное число различных форматов;
- f) что форматирование пакетов вспомогательных данных определено в Рекомендации МСЭ-R ВТ.1364;
- g) что обобщенное форматирование данных различного типа с использованием пакетов вспомогательных данных как одна из форм передачи окажется полезным для операций радиовещательной передачи,

рекомендует

1 использовать форматирование данных вида "ключ-длина-значение" (KLV) – (протокол кодирования данных с использованием Key-Length-Value), определенное в Приложении 1, в качестве метода для данных различного типа в последовательном цифровом интерфейсе;

2 чтобы соблюдение положений данной Рекомендации осуществлялось на добровольной основе. Однако данная Рекомендация может содержать некоторые обязательные положения (например, для обеспечения функциональной совместимости или возможности применения), и в таком случае соблюдение Рекомендации достигается при выполнении всех указанных положений. Для выражения требований используются слова "следует", "должен" ("shall") или некоторые другие обязывающие выражения, такие как "обязан" ("must"), а также их отрицательные формы. Употребление таких слов никоим образом не следует интерпретировать как основание для частичного или полного соблюдения положений данной Рекомендации.

Приложение 1

1 Протокол KLV¹

Таблица 1 и рисунок 1 позволяют предварительно ознакомиться с протоколом KLV для кодирования данных. Кодированные данные могут представлять собой один элемент данных или группу данных.

Протокол KLV-кодирования составлен из "ключа" идентификации в виде универсальной метки UL (Universal Label), за которым следуют числовая "длина" (длина значения) и данные, то есть "значение".

Длина полного "ключа" должна составлять 16 байтов. "Значение" – это последовательность байтов типа данных согласно спецификации в соответствующей Рекомендации, которая не подвергается дополнительному кодированию Протоколом KLV. Поле "значение" имеет переменную длину, причем какие-либо ограничения установлены в соответствующей определяющей Рекомендации.

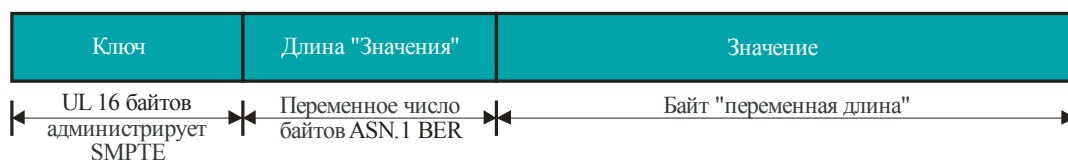
ТАБЛИЦА 1

Поля KLV для кодирования данных

Поле	Описание	Длина	Контент/Формат
Ключ	UL для идентификации значения	16 байтов	Раздел 1.1
Длина	Длина поля "значение"	Определена в соответствующем регистре, ядре (essence), стандарте применения, но имеет переменную длину	Раздел 1.2
Значение	Значение, связанное с ключом	Переменная	Раздел 1.3

РИСУНОК 1

KLV-кодирование



ВТ.1563-01

1.1 Ключ на основе универсальной метки (UL)

В протоколе KLV-кодирования в качестве ключа, идентифицирующего данные в поле "значение", должна использоваться исключительно универсальная метка UL (SMPTE 298M) фиксированной длины 16 байтов, администрируемая SMPTE. Сокращение UL используется во всей настоящей Рекомендации как ссылка на Universal Label, администрируемую SMPTE (см. Дополнение 2).

Полный ключ должен содержать поле в 16 байтов, включающее идентификатор объекта (0x06) и размер UL (0x0E, указывающий полный размер ключа 16 байтов) с последующей серией субидентификаторов, начиная от UL-кода (0x2B) и указателя SMPTE (0x34). Субидентификаторы определяют указатель UL (байты 3~8 включительно) и указатель элемента (байты 9~16 включительно), как показано в таблице 2.

¹ SMPTE ведет регистр значений универсальной метки, значений типа и записей в словарь метаданных. Читателям настоящей Рекомендации предлагается проверять зарегистрированные значения на сайте <http://www.smpte-ra.org/> и ознакомиться с последними записями.

ТАБЛИЦА 2

Описание полей для ключа, используемого при KLV-кодировании данных

Номер	Поле	Описание	Длина	Контент/Формат
	Заголовок UL			
1	OID	Идентификатор объекта	1 байт	Всегда 0x06
2	Размер UL	Размер UL 16 байтов	1 байт	Всегда 0x0E
	Указатель UL			
3	Код UL	Конкатенация субидентификаторов ISO, ORG	1 байт	Всегда 0x2B
4	Указатель SMPTE	Субидентификатор SMPTE	1 байт	Всегда 0x34
5	Указатель категории	Указатель категории, идентифицирующий категорию описанного реестра (например, словари)	1 байт	См. таблицу 3
6	Указатель реестра	Указатель реестра, идентифицирующий конкретный регистр в категории (например, словари метаданных)	1 байт	См. таблицу 3
7	Указатель структуры	Указатель варианта структуры, входящий в указатель заданного реестра	1 байт	Раздел 4.1.3
8	Номер версии	Версия данного регистра, который первый определяет элемент, специфицированный указателем элемента	1 байт	Нарастающий номер
9–16	Указатель элемента	Уникальная идентификация конкретного элемента в контексте указателя UL	8 байтов	См. соответствующую Рекомендацию и версию

Первые два субидентификатора, следующие за указателем SMPTE, должны иметь зарезервированные значения для протокола KLV-кодирования согласно настоящей Рекомендации.

Каждое слово в SMPTE 298M UL кодировано с использованием базовых правил кодирования (BER) ASN.1 применительно к кодированию идентификатора объекта, описываемому в ISO/IEC 8825-1.

Значение каждого байта в указателе UL должно быть ограничено диапазоном 0x01–0x7F, который в кодировании BER идентификатора объекта представлен одним байтом.

Значение указателя элемента кодировано с использованием базовых правил кодирования (BER) ASN.1 применительно к кодированию идентификатора объекта и должно иметь длину 8 байтов.

Значимость разрядов в субидентификаторах указателя UL и указателя элемента должна возрастать слева направо, причем первый субидентификатор должен иметь наибольший вес. Субидентификатор со значением 0x00, самый левый в ключе, должен обозначать конец метки, причем все субидентификаторы меньшего веса должны быть также установлены на 0x00. Субидентификаторы с весом 0x00 не должны существенно влиять на значение ключа.

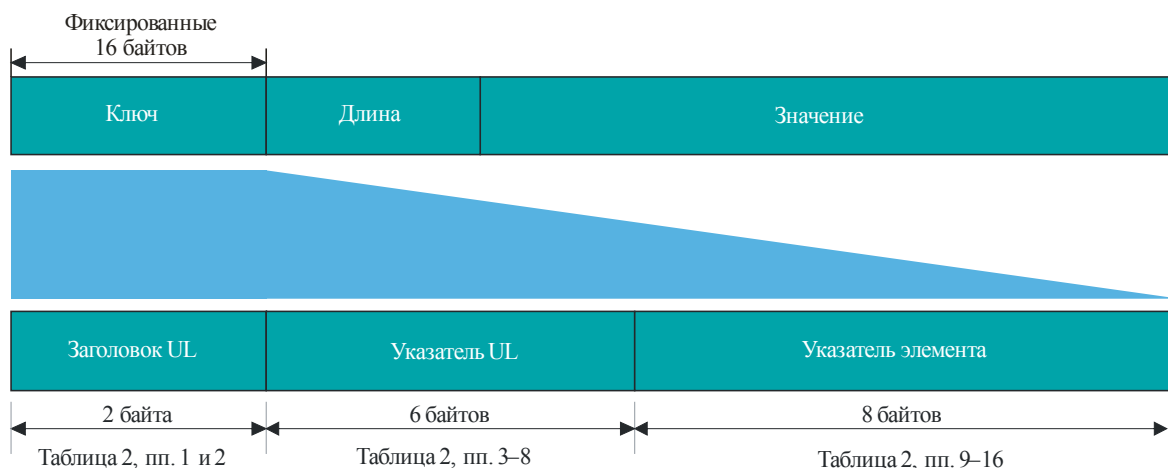
Документ SMPTE 298M определяет только первые четыре байта UL: ID объекта, размер UL, код UL и указатель SMPTE. В настоящей Рекомендации устанавливается применение меток UL согласно SMPTE 298M применительно к KLV-кодированию и определяется семантика остальных субидентификаторов указателя UL (байты 5–8). Семантика указателя элемента (байты 9–16) определена в ряде отдельных документов, совокупность которых охватывает все определенные значения указателя UL.

Декодеры, которые распознали ключ, но не намерены или не могут декодировать связанное с ним значение, могут проигнорировать элемент, но должны продолжить процесс декодирования последующих элементов, используя значение "длина" для "перепрыгивания" значения недекодированного элемента. Если декодеры запоминают и передают дальше этот элемент, они должны передать его без изменения.

Байты 5 и 6 ключа используются для идентификации содержимого значения, а также для определения интерпретации значения для всех значений указателя элемента в данной категории и в данном указателе реестра. Использование байтов 5 и 6 определено в таблице 3. Если байты 5 и 6 не соответствуют ни одному из значений таблицы 3, синтаксический анализатор должен прекратить интерпретацию содержимого байтов значения ("V") и должен сделать "K", "L" и "V" доступными для обработки согласно приложению. Синтаксический анализ байта, следующего непосредственно за окончанием "V", должен быть продолжен.

ПРИМЕЧАНИЕ 1. – Авторы применений должны помнить, что в числовых полях SMPTE UL существуют общественные и частные регистры и что эти регистры будут содержать действующие ключи KLV, с которыми синтаксический анализатор возможно не знаком. Для обеспечения функциональной совместимости важно обеспечивать на уровне применений интерпретацию нераспознанных ключей KLV.

РИСУНОК 2
Структура ключа



ВТ.1563-02

1.1.1 Указатель UL

В таблице 3 определяются значения байтов для указателей, которые должны использоваться в байтах 5–7 указателя UL. Рекомендации и Рекомендуемые практики (RP – Recommended Practice) SMPTE, которые определяют ключ со значением байта 5 (указатель категории реестра) в диапазоне 0x01–0x04, должны регистрировать полный ключ или ключи, используемые Центром регистрации SMPTE в реестре, который идентифицируется байтами 6 и 7 (указатель реестра и указатель структуры).

1.1.1.1 Словари

Стандарты и RP SMPTE, которые определяют значение слова 5 ключа как 0x01, являются рекомендациями по словарям и должны использоваться для определения единичных элементов данных с конструкцией KLV-данных.

1.1.1.2 Группы (наборы и пакеты)

Стандарты и RP SMPTE, которые определяют значение слова 5 ключа как 0x02, являются Рекомендациями по наборам и пакетам и должны использоваться для определения групп KLV-кодированных элементов данных.

ТАБЛИЦА. 3
Указатель UL для байтов 5–7

Указатель категории	Указатель реестра	Определено в:	Указатель структуры	Внешние ссылки (для сведения)
Байт 5	Байт 6		Байт 7	
0x01 – Словари		Раздел 5		
	01 – Словари метаданных	Раздел 5.1.1	0x01~0x7F	0x01: SMPTE 335M
	02 – Словари ядра	Раздел 5.1.2	0x01~0x7F	
	03 – Словари управления	Раздел 5.1.3	0x01~0x7F	
	04 – Словари типов	Раздел 5.1.4	0x01~0x7F	0x01: Проект типов CD2003
0x02 – Группы (наборы и пакеты)		Раздел 6		SMPTE 395M
	01 – Универсальные наборы	Раздел 6.1, таблица 4	0x01~0x7F	
	02 (по умолчанию) – Глобальные наборы	Раздел 6.2, таблица 6	Раздел 6.2, таблица 5	
	03 (по умолчанию) – Локальные наборы	Раздел 6.3, таблица 8	0x01~0x7F	
	04 (по умолчанию) – Пакеты переменной длины	Раздел 6.4, таблица 10	0x01~0x7F	
	05 – Пакеты определенной длины	Раздел 6.5, таблица 11	0x01~0x7F	
	06 – Зарезервировано	Раздел 6.6	0x01~0x7F	
03 – Оболочки и контейнеры		Раздел 7		
	01 – Простые оболочки и контейнеры	Раздел 7.1	0x01~0x7F	
	02 – Сложные оболочки и контейнеры	Раздел 7.2	0x01~0x7F	
04 – Метки		Раздел 8		
	Регистр меток	Раздел 8	0x01~0x7F	0x01: SMPTE 400M
05 – Зарегистрированная частная информация		Раздел 9		RP225
06 – 7E – Зарезервировано				

1.1.1.3 Оболочки и контейнеры

Стандарты и RP SMPTE, которые определяют значение слова 5 ключа как 0x03, являются Рекомендациями по оболочкам и контейнерам и пользуются ключом для идентификации оболочки или контейнера и его содержимого. Определения терминов "оболочка" и "контейнер" даны в Дополнении А.

1.1.1.4 Метки

Кодирование меток определено в разделе 5.

1.1.1.5 Зарегистрированная частная информация

Кодирование зарегистрированной частной информации определено в разделе 6.

1.1.1.6 Зарегистрировано

Резервированные категории реестров резервированы для расширения настоящей Рекомендации в будущем или для положений, которые определены другими стандартами SMPTE. Никакие другие спецификации не должны использовать эти резервированные значения.

1.1.2 Указатель реестра

Как показано в таблицах 2 и 3, байт 6 должен определить указатель реестра, идентифицируя конкретный регистр в категории (например, словарь метаданных). Глобальные наборы, локальные наборы и пакеты переменной длины используют более одного значения для идентификации длин в поле "длина", а в случае локальных наборов – также поле "локальный тег".

Использование значений указателя реестра определено в таблице 3.

1.1.3 Указатель структуры

Как показано в таблицах 2 и 3, байт 7 должен определить указатель структуры для данного реестра.

Значения указателя структуры назначаются, чтобы различать несовместимые версии одного и того же регистра. Их можно было бы принять за основной номер версии.

Использование значений указателя структуры определено в таблице 3.

1.1.4 Номер версии

В случае элементов, зарегистрированных в SMPTE, байт 8 должен определить номер версии данного регистра, который впервые определяет элемент, специфицированный указателем элемента.

В случае элементов, не зарегистрированных в SMPTE, организация, ответственная за регистрацию элементов, должна определить собственные правила нумерации версий.

ПРИМЕЧАНИЕ 1. – SMPTE назначит первый узел, идентифицирующий организацию, не входящую в SMPTE, и этому узлу будет дан номер версии SMPTE. Значения номера версии, которые предоставляет узел, назначенный SMPTE, будет затем назначен соответствующей не входящей SMPTE организацией, а не SMPTE.

Регистры могут дополняться новыми элементами уже после первоначального утверждения управляющей Рекомендации, стандарта или RP. Каждый раз при добавления набора определений элементов текущий номер версии конкретного регистра увеличивается на единицу. Каждая запись в регистре содержит номер версии, в которой элемент был впервые определен. Именно этот номер переносится байтом 8.

Синтаксические анализаторы могут игнорировать номер версии или использовать номер версии как вспомогательный ориентир для проверки последовательности проверять в процессе синтаксического анализа ключа.

1.1.5 Указатель элемента

Указатель элемента состоит из байтов 9–16 ключа.

Поле "указатель элемента" должно иметь фиксированную длину 8 байтов. Значения указателя элемента имеют длину от одного до 8 байтов, дополняемых справа нулевыми байтами для заполнения поля в 8 байтов. Кодирование выполняется с использованием ASN.1 BER идентификатора объекта, как описано в разделе 4.1.

Точное значение и конструкция указателя элемента зависят от конкретного регистра и значения указателя структуры, как описано подробнее в последующем разделе.

1.2 Кодирование поля "длина KLV"

В протоколе KLV-кодирования значение поля "длина" должно кодироваться с использованием Базовых правил кодирования (BER) применительно к кодированию байтов длины короткой или длинной формы. Соответствующие спецификации см. в ISO/IEC 8825-1, пп. 8.1.3, 8.1.3.3–8.1.3.5 (см. Приложение К). Этот самодостаточный метод кодирования поля "длина" позволяет производить эффективный синтаксический анализ KLV-кодированных данных. Когда протокол кодирования KLV

применяется для групп KLV-кодированных единиц, поле "длина" для отдельных единиц может принять другой метод, как определено Рекомендацией по кодированию этой группы (см. раздел 3).

Когда это уместно, Рекомендации и Рекомендуемые практики для отдельных применений могут установить максимальную длину в байтах для поля "длина" или же наложить ограничения на диапазон значений поля "длина", чтобы упростить требования к декодеру.

ПРИМЕЧАНИЕ 1. – Хотя в настоящей Рекомендации отсутствуют ограничения на максимальное число байтов поля "длина", наличие длинных полей "длина" может быть обусловлено первым байтом ASN.1 BER для кодирования длины длинной формы.

ПРИМЕЧАНИЕ 2. – Рекомендуется пользоваться короткой формой ASN.1 BER для всех полей "значение", длина которых меньше или равна 127 (0x7F).

При реализациях следует делать все возможное, чтобы назначить для поля "длина" правильное значение. Однако при выполнении некоторых операций практически не удастся установить длину поля "значение". Это имеет место при поступлении потока данных с ключом и полем "длина", назначенными в исходной точке. В этом случае значение поля "длина" нельзя переназначить до прекращения поступления потока, но в этот момент может оказаться невозможным вернуть поле "длина", чтобы ввести требуемое значение. В этих случаях поле "длина" должно быть установлено на (0x80), что должно указывать недерминистическую длину поля "значение". Любой документ для приложения, который допускает неопределенную длину поля "значение", должен определить альтернативный метод нахождения конца поля "значение".

ПРИМЕЧАНИЕ 3. – Значение длины (0x80) используется, потому что оно обычно не имеет смысла как значение длинной формы ASN.1 BER, поскольку указывает на нулевые последующие байты.

1.3 Кодирование значений данных

Значения данных могут быть отдельными элементами данных либо группами данных. В любом случае данные – это последовательность байтов, длина которой указана значением поля "длина". Последний байт в поле "значение" должен быть байтом конца последовательности байтов.

1.4 Пустые элементы данных

Спецификации по обеспечению непрерывности KLV-пакетов, включая любые зазоры между KLV-пакетами, не входят в сферу действия настоящей Рекомендации и рассматриваются в соответствующих документах транспортного уровня.

Вместе с тем, если применения это потребуют, зазоры в последовательности данных можно ввести, применив специфический "пустой" элемент данных. Использование "пустых" элементов данных не обязательно.

"Пустой" элемент данных – это KLV-кодированный пакет, который должен определить значение длины, за которым следует пустое поле "значение". Не следует предпринимать попытки интерпретировать данные в поле "значение".

Элементы "пустых" данных могут кодироваться либо как отдельные элементы либо в составе наборов, когда это допускает конкретное определение набора.

Применения могут удалить или пропустить любые или все "пустые" элементы данных при получении. Применения могут вставить "пустые" элементы данных, но не должны требовать от других применений сохранения таких элементов.

"Пустой" элемент данных должен быть определен в словаре метаданных, а может быть определен и в других словарях.

ПРИМЕЧАНИЕ 1. – Раздел 4.1.1 содержит руководство по использованию байта номера версии. В частном случае пустого элемента данных используются реализации с различными значениями версий, поэтому в декодерах рекомендуется игнорировать значение номера.

ПРИМЕЧАНИЕ 2. – "Пустой" элемент данных имеет другое широко распространенное название – "заполняющий элемент".

2 KLV-кодирование отдельных элементов данных

KLV-кодирование отдельных элементов данных – это простое применение ключа, длины и значения, как определено в разделе 4.

Ключи отдельных элементов данных определены в регистре вместе с диапазонами длины и спецификацией собственно значения. Для отдельных элементов данных значение байта 5 ключа должно быть равно 0x01.

2.1 Регистры отдельных элементов данных

Отдельные элементы данных, подвергнутые KLV-кодированию, собираются в регистрах в целях управления данными.

В настоящей Рекомендации определяются четыре различных регистра, указанных в таблице 3. Эти регистры являются "словарями", которые идентифицируются значением байта 6 ключа согласно следующему.

2.1.1 Словари метаданных

Словари метаданных должны идентифицироваться байтом 6, имеющим значение 0x01. Словари метаданных должны быть регистрами элементов метаданных.

Метаданные – это информация, отличная от ядра, которая не имеет неотъемлемого отдельного значения, но связана с ядром (то есть они контекстуальны и не имеют значения вне связи с взаимодействующим ядром). Примеры метаданных: URL, URI, временной код, MPEG-2 PCR, имя файла, программные метки, информация об авторском праве, управление версией, цифровой водяной знак, ключи условного доступа и пр.

2.1.2 Словари ядра

Словари ядра должны идентифицироваться байтом 6, имеющим значение 0x02. Словари ядра должны быть регистрами элементов ядра.

Ядро – это данные, которые представляют изображения, звук и текст. Типы ядра: видео, аудио и данные различного рода, в том числе титры, графика, неподвижные изображения, текст, улучшенные продукты и другие данные, которые требуются каждому из применений.

2.1.3 Словари управления

Словари управления должны идентифицироваться байтом 6, имеющим значение 0x03. Словари Управления должны быть регистрами элементов данных управления.

2.1.4 Словари типов

Словари типов должны идентифицироваться байтом 6, имеющим значение 0x04. Словари типов должны быть регистрами типов данных.

Многие значения регистров совместно пользуются общим набором определений для множественных представлений данных. Чтобы упростить определения регистров, должен использоваться словарь типов для определения этих представлений данных. Словарь типов должен быть совместно используемым ресурсом для всех других словарей.

2.2 Идентификация представлений данных "значение"

Значение многих элементов данных можно представить более чем одним способом. Например, в словаре метаданных время запуска можно представить строкой символов временного кода или эффективным форматом с плотно упакованными битами. Первый способ обеспечивает прямое отображение на дисплее, второй – высокую эффективность передачи, ориентированную на узкополосные каналы данных. Многие такие элементы словаря метаданных имеют множественные представления данных для одного и того же указателя.

Если элемент данных имеет более одного представления данных для значения, одно из них должно быть определено как представление по умолчанию и ему должен быть назначен ключ, имеющий по крайней мере один нулевой байт в конце строки. Остальным представлениям должны быть назначены ключи, где самый левый нулевой байт в конце строки заменен ненулевыми значениями, которые должны назначаться последовательно. Каждое представление должно быть документироваться в регистре отдельной записью.

Пример для сведения:

- 01.02.03.04.00.00.00.00 – это "имя" (представление данных по умолчанию с использованием 16-бит символов Unicode).
- 01.02.03.04.01.00.00.00 – это "имя" (альтернативное представление данных с использованием 7-бит символов ISO).
- 01.02.03.04.02.00.00.00 – это "имя" (еще одно представление данных с использованием символов UTF-8 Unicode).

Синтаксический анализатор трактует все представления как один и тот же элемент данных, то есть он опознает 01.02.03.04.00, затем ищет xx на месте "00", чтобы идентифицировать различные кодировки. Поскольку представление по умолчанию определено, лишний ненулевой член в 5-й позиции свидетельствует о новом представлении данных элемента регистра по умолчанию. Следует заметить, что различные представления могут наложить ограничения на значения, которые может использовать элемент данных.

3 Групповое KLV-кодирование

Можно использовать групповое кодирование элементов данных, чтобы уменьшить лишний запас, обусловленный повторением избыточной информации, которая возникает в ключе каждой единицы, как это имеет место в универсальных наборах. Групповое кодирование также позволяет логическим группам единичных элементов данных или группам элементов применить совместное кодирование и предложить опции по увеличению битовой эффективности. Чтобы повысить эффективность кодирования, для поддержки универсальных наборов, глобальных наборов, локальных наборов, пакетов переменной длины и пакетов задаваемой длины, можно применить протокол кодирования KLV, как описано ниже:

- Универсальные наборы должны использоваться для формирования логических группировок элементов данных и других KLV-кодированных элементов. Универсальные наборы всегда пользуются полными конструкциями KLV-кодирования.
- Глобальные наборы определяются через универсальные наборы, но обеспечивают эффективность кодирования благодаря совместному использованию общего заголовка ключа. Выигрыш при кодировании достигается без потери информации, причем каждый ключ можно восстановить полностью, используя только данные глобального набора.
- Локальные наборы определяются через универсальные наборы, но обеспечивают эффективность кодирования благодаря использованию коротких локальных тегов, смысловое значение которых определяется в контексте локального набора. Локальные наборы сохраняют конструкцию KLV-кодирования, но необходима отдельная Рекомендация или RP, чтобы определить смысловое значение локальных тегов и иметь план перехода от значения локального тега к значению ключа.
- Пакеты переменной длины определяются как результат дальнейшего группирования элементов данных, которое устраняет использование ключей и локальных тегов во всех единичных элементах в группе. Поэтому пакеты переменной длины опираются на Рекомендацию или RP, которые определяют порядок следования элементов данных в пакете и универсальную метку каждого элемента в пакете.
- Пакеты задаваемой длины представляют собой наиболее эффективное (и наименее гибкое) группирование элементов данных, которое устраняет использование ключей и локальных тегов, а также удаляет длину во всех единичных элементах группы. Поэтому пакеты задаваемой длины опираются на Рекомендацию или RP, которые определяют порядок следования элементов данных, длину каждого элемента данных в пакете и UL каждого элемента в пакете.

Групповое кодирование должно использоваться только для кодирования наборов и пакетов, описанных в настоящей Рекомендации.

Наборы и пакеты должны содержать несколько единичных элементов данных, кодированных как группа конструкцией данных наборов или пакетов KLV. Набор или пакет должен определяться полным ключом, значение которого должно быть зарегистрировано Центром регистрации SMPTE.

Наборы и пакеты могут кодировать данные, которые сами являются наборами и пакетами, а также единичными элементами регистров. Это называется рекурсивным кодированием KLV, причем настоящая Рекомендация не накладывает ограничения на число уровней рекурсии, которые могут использоваться конкретным применением.

Наличие наборов и пакетов должно указываться значением 0x02 в поле "указатель категории регистра" (байт 5) ключа набора или пакета. Для идентификации типа набора или пакета должно использоваться поле "указатель регистра" (байт 6). Регистр набора или пакета должен идентифицироваться полем "указатель структуры" (байт 7), а версия регистра должна идентифицироваться полем "номер версии" (байт 8).

Значение набора или пакета должно состоять из ряда единичных элементов данных, кодированных согласно определению типа набора или пакета. В пакете определены порядок и наличие элементов. В наборе порядок и наличие элементов не определены по умолчанию. Конкретные рекомендованные Практики или Рекомендации могут определить ограничения на порядок и наличие элементов данных в конкретном наборе или конкретной группе наборов.

В последующих разделах определяется, как кодируются элементы данных в универсальных наборах, глобальных наборах, локальных наборах, пакетах переменной длины и пакетах определенной длины.

3.1 Универсальные наборы

Универсальный набор определяется как совокупность нескольких элементов данных, сгруппированных в интересах применения или управления. В универсальном наборе элементы данных могут быть размещены в любом порядке и могут присутствовать или отсутствовать.

Использование ключей для кодирования универсального набора должно быть определено сопутствующей Рекомендацией или RP, включая указатель структуры, и сопутствующий регистр универсального набора, включающий номер версии.

Ключ универсального набора должен иметь длину 16 байтов.

Длина универсального набора должна кодироваться согласно нотации ASN.1 с использованием по мере необходимости длинной или короткой формы.

Значение универсального набора должно быть последовательностью KLV-кодированных элементов, общая длина которых задается полем "длина". Все элементы данных в универсальном наборе должны применять протокол KLV-кодирования данных, включая полное значение ключа.

Соответствующие Рекомендации или RP для применения могут определять ограничения на значение универсального набора, например на число и размер элементов, разрешенную последовательность элементов и на то, обязательными или факультативными должны быть те или иные элементы.

Ключ для универсальных наборов описан в таблице 4. Указатель универсального набора определен в последних восьми байтах ключа универсального набора. Ключи универсального набора должны быть определены в соответствующей Рекомендации или RP, а значение ключа должно быть зарегистрировано Центром регистрации SMPTE в соответствии с положениями соответствующей Рекомендации или RP, чтобы гарантировать уникальность значения ключа.

На рисунке 3 показана структура данных для кодирования универсальных наборов.

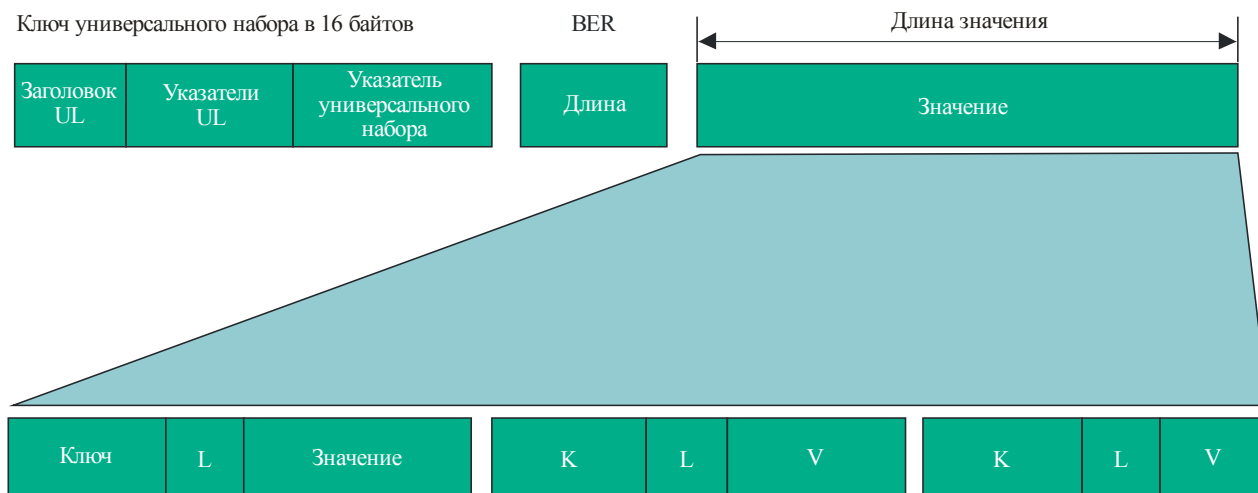
ТАБЛИЦА 4

Описание полей для ключей для KLV-кодирования универсальных наборов

Номер	Поле	Описание	Длина	Контент/Формат
	Заголовок UL			
1	OID	Идентификатор объекта	1 байт	Всегда 0x06
2	Размер UL	Размер UL 16 байтов	1 байт	Всегда 0x0E
	Указатель UL			
3	Код UL	Конкатенация субидентификаторов ISO, ORG	байт	Всегда 0x2B
4	Указатель SMPTE	Субидентификатор SMPTE	1 байт	Всегда 0x34
5	Указатель категории реестра	Наборы и пакеты	1 байт	Всегда 0x02
6	Указатель реестра	Универсальные наборы	1 байт	Всегда 0x01
7	Указатель структуры	Указатель варианта структуры в регистре универсального набора	1 байт	Определено регистром универсального набора и Рекомендацией или RP
8	Номер версии	Версия данного регистра, который первый определяет элемент, отмеченный указателем элемента	1 байт	Возрастающий номер
9–16	Указатель универсального набора	Уникальная идентификация конкретного универсального набора	8 байтов	См. регистр универсального набора

РИСУНОК 3

Структура данных KLV-кодированного универсального набора



Значение составляют отдельно KLV-кодированные элементы с ключом в 16 байтов и длиной согласно BER

3.2 Глобальные наборы

Глобальный набор определяется как совокупность нескольких элементов данных, сгруппированных, для того чтобы без потери информации уменьшить длину ключей для каждого элемента в наборе. В глобальном наборе элементы данных могут быть размещены в любом порядке и даже отсутствовать.

Использование ключей для кодирования глобальных наборов должно быть определено соответствующей Рекомендацией или RP, включая Указатель структуры, и соответствующим регистром глобального набора, включающим номер версии.

Ключ глобального набора должен иметь длину 16 байтов.

Длина глобального набора должна кодироваться по умолчанию согласно нотации ASN.1 с использованием по мере необходимости длинной или короткой формы.

Значение глобального набора должно быть последовательностью KLV-кодированных элементов, общая длина которых задается полем "длина". Все элементы данных в глобальном наборе должны применять протокол KLV-кодирования данных, но с укороченным значением глобального тега, заменяющего ключ, как описано далее.

Метка UL глобального набора должна определяться в двух частях:

Первая группа длиной 8 байтов (заголовок UL и указатель UL) должна быть зарегистрирована Центром регистрации SMPTE и использоваться для идентификации Рекомендации и RP (включая указатель структуры), касающихся глобального набора. Каждая запись в регистре глобального набора должна содержать номер версии, в которой она была впервые определена.

Вторая группа длиной 8 байтов, называемая указателем глобального набора, должна использоваться для определения общих заголовка UL и указателя UL для всех ключей глобального набора. Эта группа в 8 байтов должна содержать поля для заголовков UL, а также максимальное число указателей UL, общих для всех элементов глобального набора и учтенных указателем структуры (байт 7). Указатель глобального набора может заканчиваться нулевым байтом, свидетельствующим о достижении конца общего корня указателя UL. Длина последовательности старших байтов до завершающего нулевого байта второй группы должна составлять 2–8 байтов. Если длина второй группы равна 8 байтам, завершающий байт с нулевым значением не требуется.

Каждый глобальный тег имеет длину от 2 до 12 слов. Глобальные теги длиной менее 12 слов должны завершаться одним нулевым значением, чтобы устранить избыточные данные UL.

Полный Ключ длиной 16 байтов каждого элемента данных в глобальном наборе можно восстановить без потери информации, используя конкатенацию ненулевых байтов указателя глобального набора и глобального тега элемента. Если длина результирующей конкатенации меньше 16 байтов, остальные байты пространства в 16 байтов должны быть заполнены нулями.

В соответствующих Рекомендации или RP, касающихся применения, могут указываться ограничения на значение универсального набора, такие как число и размер элементов, разрешенная последовательность элементов и на то, обязательными или факультативными должны быть те или иные элементы.

Ключ для глобальных наборов описан в таблице 5. Указатель глобального набора определен последними 8 байтами ключа глобального набора. Ключи глобального набора должны быть определены в соответствующих Рекомендации или RP, а значение ключа должно быть зарегистрировано Центром регистрации SMPTE в соответствии с положениями соответствующих Рекомендации или RP, чтобы гарантировать уникальность значения ключа.

ТАБЛИЦА 5

Описания полей для ключа для кодирования глобального набора

Номер	Поле	Описание	Длина	Контент/Формат
	Заголовок UL			
1	OID	Идентификатор объекта	1 байт	Всегда 0x06
2	Размер UL	Размер UL 16 байтов	1 байт	Всегда 0x0E
	Указатель UL			
3	Код UL	Конкатенация субидентификаторов ISO, ORG	1 байт	Всегда 0x2B
4	Указатель SMPTE	Указатель SMPTE	1 байт	Всегда 0x34
5	Указатель категории реестра	Наборы и пакеты	1 байт	Всегда 0x02
6	Указатель реестра	Глобальные наборы	1 байт	См. таблицу 6
7	Указатель структуры	Указатель варианта структуры в регистре глобального набора	1 байт	См. примечание в конце таблицы
8	Номер версии	Версия регистра глобального набора, который первым определяет элемент, отмеченный указателем глобального набора	1 байт	Возрастающий номер
9–16	Указатель глобального набора	Общая часть ключа, совместно используемая всеми глобальными тегами	8 байтов	Активное число определяет байты, необходимые для создания общего корня для всех глобальных тегов (2–8 байтов)

ПРИМЕЧАНИЕ 1. – Значение равно 1 плюс число начальных байтов ключа, которые скопированы из таблицы до начала общей части. Разрешены значения от 1 (скопировано 0 байтов) до 9 (скопировано 8 байтов). Значение 5 (скопировано 4 байта) наиболее приемлемо.

На рисунке 4 показана структура, используемая для кодирования глобальных наборов данных.

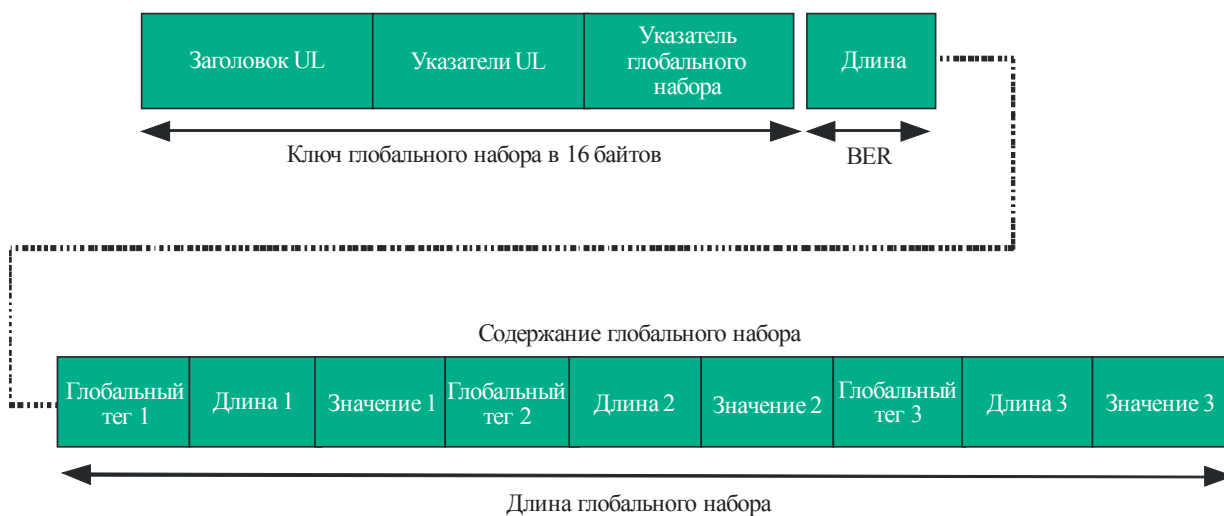
За ключом глобального набора в 16 байтов следует длина глобального набора (кодированная с использованием ASN.1 BER применительно к кодированию длины) с последующим рядом элементов данных, каждый из которых должен быть триплетом: глобальный тег, длина и значение.

По умолчанию, спецификация полей "длина" единичных элементов данных предусматривает короткую или длинную форму кодирования ASN.1 BER. Если используется кодирование, отличное от кодирования ASN.1 BER, оно должно иметь прямой порядок байтов (big-endian). В соответствии с таблицей 6 указателем реестра определен весь диапазон разрешенных длин полей "длина". Все поля "длина" глобального набора должны придерживаться одного и того же синтаксиса.

Глобальные наборы допускают рекурсию, поэтому ключ, связанный с глобальным тегом, может идентифицировать либо единичный элемент данных из регистра, либо набор или пакет данных из любых Рекомендации или RP и соответствующего регистра по наборам и пакетам.

РИСУНОК 4

Структура данных KLV-кодированного глобального набора



ВТ.1563-04

ТАБЛИЦА 6

Кодирование указателя реестра (байт 6) для синтаксиса глобального набора

Значение байта 6	Поле "длина"	Описание
0x02	ASN.1 BER, короткая или длинная форма	Любая длина (по умолчанию)
0x22	1 байт	Длина до 255
0x42	2 байта	Длина до 65 535
0x62	4 байта	Длина до $2^{32}-1$

3.3 Локальные наборы

Локальный набор определяется как совокупность нескольких элементов данных, сгруппированных с целью уменьшения длины ключей для каждого элемента в наборе. В локальном наборе элементы данных могут быть размещены в любом порядке и могут присутствовать или отсутствовать.

Использование ключей для кодирования локальных наборов должно быть определено соответствующими Рекомендацией или RP, включая указатель структуры, и соответствующим регистром локального набора, включающим номер версии.

Ключ локального набора должен иметь длину 16 байтов.

Длина локального набора по умолчанию должна кодироваться согласно нотации длины ASN.1 BER с использованием по мере необходимости длинной или короткой формы.

Значение локального набора должно быть последовательностью KLV-кодированных элементов, общая длина которых задается полем "длина".

Ключ для локальных наборов описан в таблице 7. Указатель локального набора определен в последних 8 байтах ключа локального набора. Ключи локального набора должны быть определены в соответствующей Рекомендации или RP, а значение ключа должно быть зарегистрировано Центром регистрации SMPTE в соответствии с положениями соответствующей Рекомендации или RP, чтобы гарантировать уникальность значения ключа.

ТАБЛИЦА 7

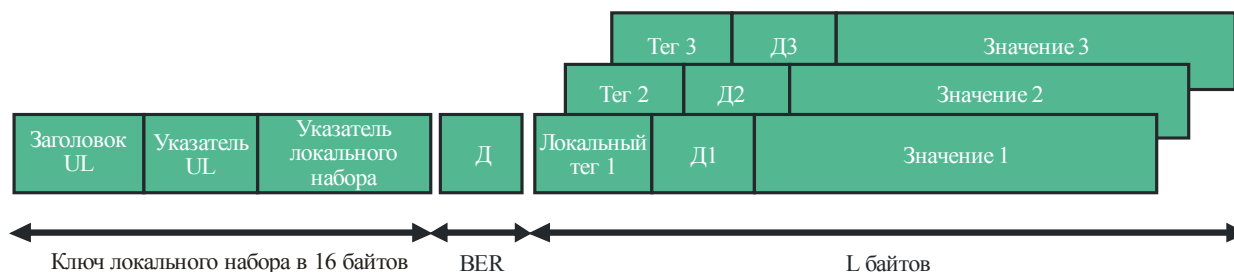
Описания полей для ключа для кодирования локального набора

Номер	Поле	Описание	Длина	Контент/Формат
	Заголовок UL			
1	OID	Идентификатор объекта	1 байт	Всегда 0x06
2	Размер UL	Размер UL 16 байтов	1 байт	Всегда 0x0E
	Указатель UL			
3	Код UL	Конкатенация субидентификаторов ISO, ORG	1 байт	Всегда 0x2B
4	Указатель SMPTE	Указатель SMPTE	1 байт	Всегда 0x34
5	Указатель категории реестра	Наборы и пакеты	1 байт	Всегда 0x02
6	Указатель реестра	Локальные наборы	1 байт	См. таблицу 8
7	Указатель структуры	Указатель варианта структуры в регистре локального набора	1 байт	Определено регистром локального набора и Рекомендацией или RP
8	Номер версии	Версия регистра локального набора, который первым определяет элемент, отмеченный указателем локального набора	1 байт	Возрастающий номер
	Указатель локального набора			
9–16	Указатель локального набора	Определяет размещение локального набора в иерархической структуре	8 байтов	Определено регистром локального набора и Рекомендацией или RP

Структура данных для кодирования локальных наборов показана на рисунке 5.

РИСУНОК 5

Структура KLV-кодированного локального набора



ВТ.1563-05

За ключом локального набора в 16 байтов следует длина набора с последующим рядом элементов данных, каждый из которых должен быть триплетом – локальный тег, длина и значение.

Предпочтительная длина полей локального тега – 1 байт. По умолчанию спецификация полей "длина" предусматривает короткую или длинную форму кодирования ASN.1 BER. Если используется кодирование, отличное от кодирования ASN.1 BER, оно должно иметь прямой порядок байтов (big-endian). В соответствии с таблицей 8 указателем реестра определяется весь диапазон разрешенных сочетаний длин локальных тегов и полей "длина".

ТАБЛИЦА 8

Кодирование указателя реестра (байт 6) для синтаксиса локального набора

Значение байта 6	Поле "длина"	Поле локального тега	Описание
0x03	Короткая или длинная форма ASN.1 BER	1 байт	Любая длина (по умолчанию)
0x0B	Короткая или длинная форма ASN.1 BER	ASN.1 OID BER	
0x13	Короткая или длинная форма ASN.1 BER	2 байта	
0x1B	Короткая или длинная форма ASN.1 BER	4 байта	
0x23	1 байт	1 байт	Длина до 255
0x2B	1 байт	ASN.1 OID BER	
0x33	1 байт	2 байта	
0x3B	1 байт	4 байта	
0x43	2 байта	1 байт	Длина до 65 535
0x4B	2 байта	ASN.1 OID BER	
0x53	2 байта	2 байта	
0x5B	2 байта	4 байта	
0x63	4 байта	1 байт	Длина до $2^{32}-1$
0x6B	4 байта	ASN.1 OID BER	
0x73	4 байта	2 байта	
0x7B	4 байта	4 байта	

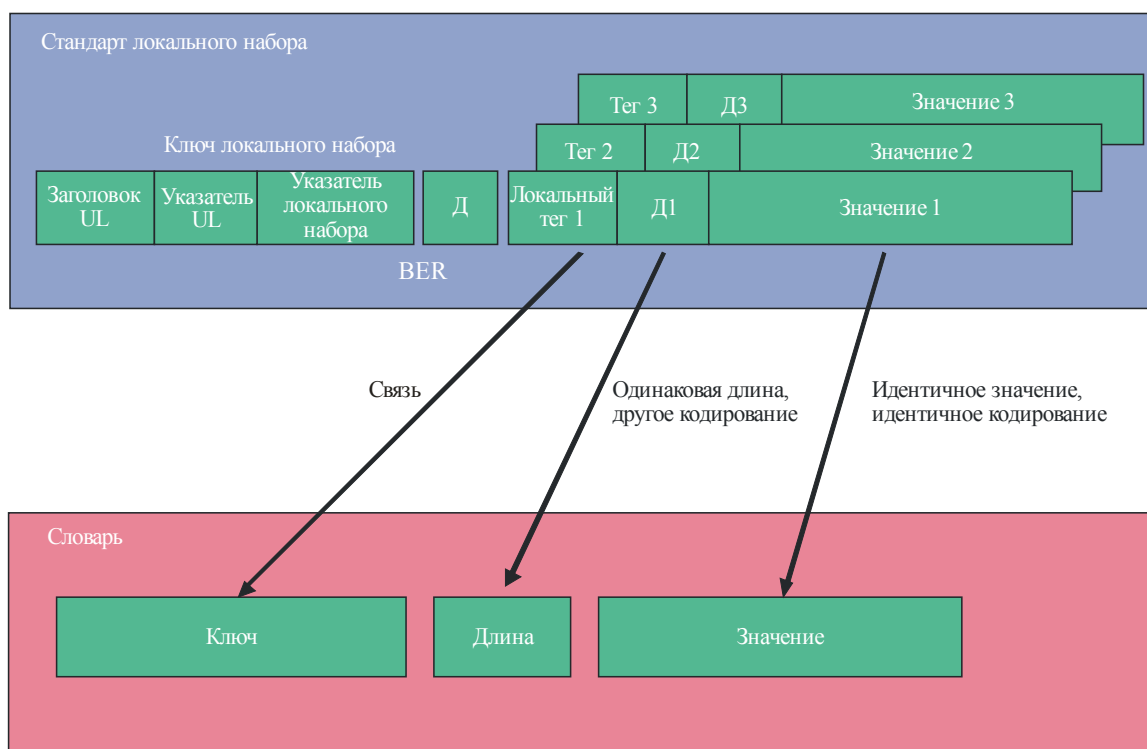
ПРИМЕЧАНИЕ 1. – Соответствующая Рекомендация или RP для локального набора должна определять связь между локальным тегом каждого элемента данных и соответствующей величиной ключа. Эта связь должна быть определена в соответствующей Рекомендации или RP для локального набора, которая предоставит каждому локальному тегу ключ определяющего элемента. Это связующее определение является механизмом, обеспечивающим пользователям настоящей Рекомендации свободу в определении их собственных псевдонимов для достижения высокоэффективного кодирования. Соответствующая Рекомендация или RP должна также определять предполагаемую сферу применимости локального тега или псевдонима в рамках спецификации. Разработчики локальных наборов должны предоставить средства преобразования между каждым тегом в локальном наборе и определяющим ключом. В отличие от универсальных наборов и глобальных наборов, в которых ключ каждого элемента данных в наборе можно воссоздать без потери информации, ключ каждого тега локального набора невозможно воссоздать без рекурсивного обращения к определяющей Рекомендации или RP и соответствующему регистру.

Локальные наборы допускают рекурсию, поэтому ключ, связанный с локальным тегом, может идентифицировать либо единственный элемент данных из регистра либо набор или пакет данных из любой Рекомендации или RP по наборам и пакетам и соответствующего регистра.

На рисунке 6 для сведения показана связь между локальным тегом и полным ключом.

РИСУНОК 6

Иллюстрация связи метки локального набора с меткой глобального набора



ВТ.1563-06

3.4 Пакеты переменной длины

Пакет переменной длины аналогичен локальному набору, но не имеет локальных тегов. Поэтому каждый элемент пакета переменной длины содержит только поле "длина" и поле "значение". В пакете переменной длины элементы должны появляться в определенном порядке.

Использование ключей в пакете переменной длины должно быть определено соответствующей Рекомендацией или RP, включающей указатель структуры и сопутствующий регистр пакета переменной длины, включающий номер версии.

Ключ пакета переменной длины должен иметь длину 16 байтов.

Длина пакета переменной длины по умолчанию должна кодироваться согласно нотации длины ASN.1 BER с использованием по мере необходимости длинной или короткой формы.

Значение пакета переменной длины должно быть последовательностью KLV-кодированных элементов, общая длина которых задается полем "длина".

Ключ для пакета переменной длины описан в таблице 9. Указатель пакета переменной длины определен в последних 8 байтах ключа локального набора. Ключи пакета переменной длины должны быть определены в соответствующей Рекомендации или RP, а значение ключа должно быть зарегистрировано Центром регистрации SMPTE в соответствии с положениями соответствующей Рекомендации или RP, чтобы гарантировать уникальность значения ключа.

ТАБЛИЦА 9

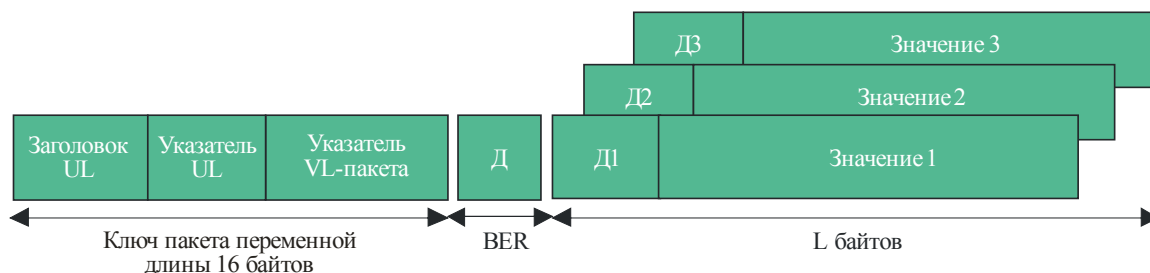
Описание полей для ключа для кодирования пакета переменной длины

Номер	Поле	Описание	Длина	Контент/Формат
	Заголовок UL			
1	OID	Идентификатор объекта	1 байт	Всегда 0x06
2	Размер UL	Размер UL 16 байтов	1 байт	Всегда 0x0E
	Указатель UL			
3	Код UL	Конкатенация субидентификаторов ISO, ORG	1 байт	Всегда 0x2B
4	Указатель SMPTE	Указатель SMPTE	1 байт	Всегда 0x34
5	Указатель категории реестра	Наборы и пакеты	1 байт	Всегда 0x02
6	Указатель реестра	Пакеты переменной длины	1 байт	См. таблицу 10
7	Указатель структуры	Указатель варианта структуры в регистре пакета переменной длины	1 байт	Определен регистром пакета переменной длины и Рекомендацией или RP
8	Номер версии	Версия регистра пакета переменной длины, который первым определяет элемент, отмеченный указателем пакета переменной длины	1 байт	Нарастающее число
	Указатель пакета переменной длины			
9–16	Указатель пакета переменной длины	Определяет размещение пакета переменной длины в иерархической структуре	8 байтов	Определен регистром пакета переменной длины и Рекомендацией или RP

Структура данных для кодирования пакетов переменной длины показана на рисунке 7.

РИСУНОК 7

Структура KLV-кодированного пакета переменной длины (VL)



ВТ.1563-07

За ключом пакета переменной длины 16 байтов следует длина пакета переменной длины (кодированная согласно нотации ASN.1 BER применительно к кодированию длины) с последующим рядом элементов, каждый из которых должен быть дуплетом: длина и значение.

Спецификация по умолчанию полей "длина" единичных элементов предусматривает короткую или длинную форму кодирования ASN.1 BER. Если используется кодирование, отличное от кодирования BER, оно должно иметь прямой порядок байтов (big-endian). В соответствии с таблицей 10 указателем реестра определен весь диапазон разрешенных длин поля "длина".

ТАБЛИЦА 10

Кодирование указателя реестра (байт 6) для синтаксиса пакета переменной длины

Значение байта 6	Поля "длина"	Описание
0x04	ASN.1 BER короткая или длинная форма	Любая длина (по умолчанию)
0x24	1 байт	Длина до 255
0x44	2 байта	Длина до 65 535
0x64	4 байта	Длина до $2^{32}-1$

Поскольку элементы пакета не имеют локального тега, порядок следования элементов должен быть определен Рекомендацией, Стандартом или RP.

Соответствующая Рекомендация или RP для пакета переменной длины должна определить связь между каждым элементом данных и соответствующим значением ключа посредством предоставления ключа каждому элементу в пакете переменной длины. Это связующее определение является средством, которое предоставляет пользователям этой Рекомендации свободу в определении их собственных псевдонимов для достижения высокоэффективного кодирования. Разработчики пакетов переменной длины должны показать возможность преобразования между каждым элементом в пакете переменной длины и определяющим ключом. В отличие от универсальных наборов и Глобальных наборов, в которых ключ каждого элемента в наборе можно воссоздать без потери информации, ключ каждого элемента в пакете переменной длины невозможно воссоздать без рекурсивного обращения к определяющей Рекомендации или RP и соответствующему регистру.

Пакеты переменной длины допускают рекурсию, поэтому ключ, связанный с элементом, может идентифицировать либо единичный элемент данных из регистра либо группу данных из любой Рекомендации или RP по наборам и пакетам и соответствующего регистра.

3.5 Пакеты определенной длины

ПРИМЕЧАНИЕ 1. – В настоящей версии Рекомендации термин "пакет фиксированной длины" был заменен термином "пакет определенной длины".

Пакет определенной длины идентичен пакету переменной длины, но не имеет полей "длина". Поэтому каждый элемент пакета определенной длины содержит только поле "значение". Элементы в пакете определенной длины должны появляться в определенном порядке.

Использование ключей для кодирования пакета определенной длины должно быть определено соответствующей Рекомендацией или RP, включающей указатель структуры и сопутствующий регистр пакета определенной длины, включающий номер версии.

Ключ пакета определенной длины должен иметь длину 16 байтов.

Длина пакета определенной длины по умолчанию должна кодироваться согласно нотации ASN.1 BER применительно к длине с использованием длинной или короткой формы, как требуется.

Значение пакета определенной длины должно быть последовательностью элементов, общая длина которых задается полем "длина".

Единичные элементы в пакете определенной длины должны появляться в определенном порядке, причем каждый элемент имеет определенную длину. Единичные элементы в пакете могут иметь значения длины, которые должны быть определены, подвергая элемент синтаксическому анализу. Результатом является пакет, имеющий определенную, но пока еще переменную общую длину. В этой Рекомендации к пакетам определенной длины не предъявляется требование иметь фиксированные, постоянные значения.

Ключ для пакета определенной длины описан в таблице 11. Указатель пакета определенной длины определен последними 8 байтами ключа локального набора. Ключи пакета определенной длины должны быть определены в сопутствующей Рекомендации или RP, а значение ключа должно быть зарегистрировано Центром регистрации SMPTE согласно положениям соответствующей Рекомендации или RP, чтобы гарантировать уникальность значения ключа.

Структуру данных для кодирования пакетов определенной длины иллюстрирует рисунке 8.

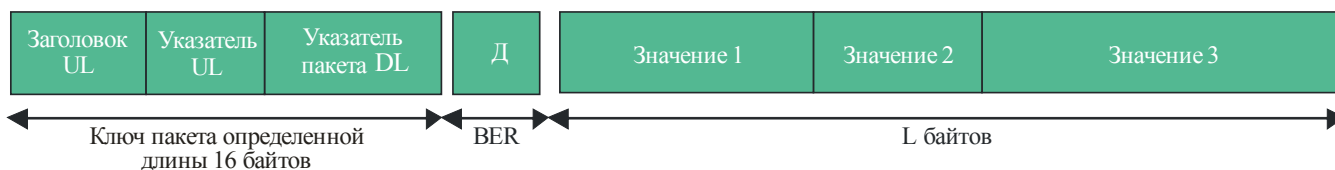
ТАБЛИЦА 11

Определение полей для ключа для кодирования пакета переменной длины

Номер	Поле	Описание	Длина	Контент/Формат
	Заголовок UL			
1	OID	Идентификатор объекта	1 байт	Всегда 0x06
2	Размер UL	Размер UL 16 байтов	1 байт	Всегда 0x0E
	Указатель UL			
3	Код UL	Конкатенация субидентификаторов ISO, ORG	1 байт	Всегда 0x2B
4	Указатель SMPTE	Указатель SMPTE	1 байт	Всегда 0x34
5	Указатель категории реестра	Наборы и пакеты	1 байт	Всегда 0x02
6	Указатель реестра	Пакеты постоянной длины	1 байт	Всегда 0x05
7	Указатель структуры	Указатель варианта структуры в регистре пакета постоянной длины	1 байт	Нарастающее число
8	Номер версии	Версия регистра пакета постоянной длины, который первым определяет элемент, отмеченный указателем пакета постоянной длины	1 байт	Нарастающее число
	Указатель пакета постоянной длины			
9–16	Указатель пакета постоянной длины	Определяет размещение пакета постоянной длины в иерархической структуре	8 байтов	Определен регистром пакета постоянной длины и Рекомендацией или RP

РИСУНОК 8

Структура KLV-кодированного пакета определенной длины (DL)



Поскольку элементы в пакете постоянной длины не имеют локального тега, порядок размещения элементов должен устанавливаться определяющей Рекомендацией или RP.

Соответствующая Рекомендация или RP для пакета постоянной длины должна определить связь между каждым элементом данных и соответствующим значением ключа посредством предоставления ключа определяющего элемента. Это связующее определение является средством, которое предоставляет пользователям настоящей Рекомендации гибкость в определении их собственных псевдонимов для достижения высокоэффективного кодирования. Разработчики пакетов постоянной длины должны показать возможность преобразования между каждым элементом в пакете постоянной длины и определяющим ключом. В отличие от универсальных наборов и глобальных наборов, в которых ключ каждого элемента в наборе можно воссоздать без потери информации, ключ каждого элемента в пакете постоянной длины невозможно воссоздать без рекурсивного обращения к определяющей Рекомендации или RP и соответствующему регистру.

Пакеты постоянной длины допускают рекурсию, поэтому ключ, связанный с элементом, может идентифицировать либо единичный элемент данных из регистра либо набор или группу данных из любой Рекомендации или RP по наборам и пакетам и соответствующего регистра.

ПРИМЕЧАНИЕ 2. – Во многих случаях группы могут кодироваться как универсальные наборы, локальные наборы, пакеты переменной длины и пакеты постоянной длины без изменения значений индивидуальных элементов метаданных в группе. В каждом случае применительно к данной группе в ключе группы изменится только байт 6, а байты 9–16 останутся прежними.

3.6 Запрещенное использование

Значение 0x06 байта 6 не должно использоваться для KLV-кодирования.

4 Оболочки и контейнеры

Оболочки и контейнеры должны быть идентифицированы байтом 5, имеющим значение 0x03.

Оболочки и контейнеры отличаются от наборов и пакетов тем, что они не обязательно пользуются общей конструкцией данных KLV для всего содержания оболочки или контейнера. Рекомендуется, чтобы в отдельных частях оболочек и контейнеров данные кодировались с использованием протокола KLV-кодирования, но эти части могут объединить другие технологии. В некоторых случаях оболочка или контейнер может пользоваться общей конструкцией KLV для некоторых применений (например, в потоковом интерфейсе), но применять другую технологию в других применениях (например, в контейнере памяти). В этих случаях оболочка или контейнер не переопределяется как набор или пакет, сохраняя определение контейнера или оболочки, чтобы обеспечить постоянство идентификации.

По определению, простые оболочки и контейнеры встраивают все данные в единственную структуру, не имеющую внешних ссылок. Простые оболочки и контейнеры должны идентифицироваться байтом 6, имеющим значение 0x01.

Сложные оболочки и контейнеры определены структурами, в которых индивидуальные элементы данных могут быть загружены в файл по ссылке, а не встроены. Сложные оболочки и контейнеры, которые могут быть эффективнее, подходят для локального окружения, где можно легко получить ссылки. Сложные оболочки и контейнеры должны идентифицироваться байтом 6, имеющим значение 0x02.

Определение отдельных спецификаций оболочек и контейнеров, которое находится вне сферы применения настоящей Рекомендации, можно будет найти в других документах.

5 Метки SMPTE

Метки SMPTE должны идентифицироваться байтом 5, имеющим значение 0x04. Метки SMPTE не должны использоваться в качестве ключа при KLV-кодировании. Метки SMPTE можно использовать как значение в триплетах KLV-кодирования или в любых других конструкциях кодирования.

Метки SMPTE должны использоваться для идентификации любого объекта, смысловое значение которого полностью передает сама UL, администрируемая SMPTE. Метки SMPTE могут использоваться для идентификации систем кодирования ядра, предоставления уникальной идентификации значений параметров, идентификации конструкций метаданных и многого другого.

В оболочках и контейнерах, а иногда даже в наборах время от времени возникает потребность идентифицировать некоторые аспекты содержания данных, которые не идентифицированы ключом набора, оболочки или контейнера. Такой аспект может быть идентифицирован, если включить метку SMPTE в набор, оболочку или контейнер в качестве элемента данных. Необходимо определить присутствие метки SMPTE на высоком уровне структуры UL SMPTE, чтобы декодеры понимали, что элемент является меткой SMPTE, а не ключом кодированного триплета KLV. В таблице 12 приводится описание полей UL SMPTE для меток SMPTE. На рисунке 9 показана структура метки SMPTE.

ТАБЛИЦА 12

Описание полей администрируемой SMPTE UL для меток SMPTE

Номер	Поле	Описание	Длина	Контент/Формат
	Заголовок UL			
1	OID	Идентификатор объекта	1 байт	Всегда 0x06
2	Размер UL	Размер UL 16 байтов	1 байт	Всегда 0x0E
	Указатель UL			
3	Код UL	Конкатенация субидентификаторов ISO, ORG	1 байт	Всегда 0x2B
4	Указатель SMPTE	Указатель SMPTE	1 байт	Всегда 0x34
5	Указатель категории реестра	Метки	1 байт	Всегда 0x04
6	Указатель реестра	Регистр специальных меток	1 байт	Нарастающее число
7	Указатель структуры	Указатель варианта структуры в регистре меток	1 байт	Нарастающее число
8	Номер версии	Версия регистра меток, который первым определяет элемент, отмеченный указателем меток	1 байт	Нарастающее число
	Указатель меток			
9–16	Указатель меток	Определяет размещение меток в иерархической структуре	8 байтов	Определен регистром меток и Рекомендацией или практикой

РИСУНОК 9

UL, администрируемая SMPTE, для меток SMPTE

Метка SMPTE



UL, администрируемая SMPTE, 16байтов

6 Зарегистрированная частная информация

Зарегистрированная частная информация должна идентифицироваться в поле "категория реестра" метки UL, администрируемой SMPTE, установленную на 0x05. Назначение этой категории – предоставить Рекомендацию, однозначные средства переноса информации, которую регистрирует внешнее агентство когда нежелательно открыто вносить эту информацию в регистр SMPTE или квалифицировать ее как метаданные или как ядро.

Эта зарегистрированная частная информация не определяет администрируемую SMPTE UL как ключ или как метку SMPTE.

Следует обратиться к SMPTE RP 225, чтобы иметь авторитетное определение остальных полей UL, если указатель категории реестра не установлен на 0x05.

Дополнение А

Глоссарий терминов и аббревиатур

В настоящей Рекомендации используются следующие термины и определения.

SMPTE UL – Аббревиатура термина **SMPTE Administered UL**.

Абстрактная синтаксическая нотация (Abstract Syntax Notation, ASN) (см. ISO/IEC 8825-1 (МСЭ-Т X.690)).

Администрируемая SMPTE UL (SMPTE Administered UL) – UL, администрируемая SMPTE согласно ANSI/SMPTE 298M. Все UL, администрируемые SMPTE, имеют длину 16 байтов.

Базовые правила кодирования (Basic Encoding Rules (BER)) – рекомендация ИСО по кодированию применительно к различным конструкциям в ASN.1. Включает кодирование идентификаторов объекта, а также полей "длина". Байты длины в пакете KLV должны соответствовать Базовым правилам кодирования (BER) в случае кодирования короткой или длинной формы согласно спецификации в ISO/IEC 8825-1, пп. 8.1.3.4 и 8.1.3.5.

Байт (byte) – широко распространенный эквивалент термина "октет".

Группа данных (Data Group) – совокупность элементов данных.

Данные управления (Control Data) – элемент данных, предоставляющий функцию управления данными ядра или метаданными.

Идентификатор объекта (Object identifier, OID) – первый байт в UL, который идентифицирует UL – сокращенно OID. Всегда имеет значение "06" в шестнадцатеричном (hex) счислении (0x06).

Канонические правила кодирования (Canonical Encoding Rules, CER) (см. ISO/IEC 8825-1 (МСЭ-Т X.690)).

Ключ (Key) – универсальная метка длительностью 16 байтов, администрируемая SMPTE, которая используется при KLV кодировании данных.

Ключ-Длина-Значение (Key-Length-Value, KLV).

Контейнер (Container) – обобщенное название объекта данных, который предоставляет структуру для размещения информации различного рода. Этот термин обычно употребляется в мультимедийной среде, где аудиоданные, видеоданные, данные ядра и метаданные сводятся в единый объект данных.

Международный уникальный идентификатор аудиовизуальной продукции (International Standard Audiovisual Number, ISAN).

Метаданные (Metadata) – обычно определяются как "данные о данных" или "данные, описывающие другие данные". Метаданные – это информация, которая считается вспомогательной или иным образом непосредственно дополняющей по отношению к ядру. Кроме того, любая информация, считающаяся полезной или ценной при связи с ядром.

Метка SMPTE (SMPTЕ Label) – самоидентифицирующаяся UL SMPTE (см. раздел 7).

Оболочка (Wrapper) – "Целевая группа по гармонизированным рекомендациям для обмена программным материалом в виде цифровых потоков" (TFHS) (SMPTE/EBU) определила оболочку как средство упаковки программной информации (видео-, аудиоданные, данные ядра и метаданные) в единую структуру. Такое определение оболочки идентично определению контейнера, однако "оболочку" можно также использовать для "обертывания" других метаданных вокруг уже определенного контейнера. В этом смысле контейнер – это многоцелевой блок, содержащий аудиовизуальную информацию, а оболочка – это обертка этого блока, включая метки и другие поддерживающие метаданные.

Октет (Octet) – слово данных, содержащее 8 двоичных цифр.

Особые правила кодирования (Distinguished Encoding Rules, DER)

(см. ISO/IEC 8825-1 (МСЭ-Т X.690)).

Простое кодирование (Primitive Encoding) – в нотации ASN.1 метод кодирования с определенной длиной, который применяется к типам простого кодирования и типам, образованным из простых типов посредством присвоения тегов. Метод требует, чтобы длина субидентификаторов была известна заранее.

Прямой порядок байтов (Big-Endian) – любой мультиоктетный (мультибайтовый) элемент данных, в котором старший октет (байт) появляется первым по времени или занимает крайнее левое положение на диаграммах.

Регистр (Register) – банк или база данных, которую поддерживает реестр.

Реестр (Registry) – информационная система для регистрации данных.

Словарь (Dictionary) – регистр, предоставляющий семантическую интерпретацию элементов данных, содержащихся в регистре.

Словарь метаданных (Metadata Dictionary) – база данных утвержденных элементов метаданных в Рекомендации, включая определения и разрешенные форматы.

Тег (Tag) – специальная форма идентификации, локальная по отношению к формату кодирования. В полностью развернутой форме тег может быть идентичен указателю элемента.

Тип данных (Data Type) – (см. определение типа ниже.)

Тип или Тип данных (Type or Data Type) – информация, определяющая способ представления данных.

Универсальная метка (Universal Label, UL) – идентификатор объекта согласно ISO/IEC 8824-1 (см. также ANSI/SMPTE 298M). В настоящей Рекомендации этот термин означает то же, что и администрируемая SMPTE UL.

Центр регистрации SMPTE (SMPTE Registration Authority) – организация по регистрации, которая протоколирует случаи использования ключей UL, соответствующих ANSI/SMPTE 298M, и другие эталонные данные.

Элемент данных (Data Item) – в настоящей Рекомендации это вводимые данные минимальной длины. Следует отметить, что термин "элемент" широко используется в других документах и может иметь другое значение. Следует отметить также, что в настоящей Рекомендации элемент данных – это не группа данных.

Элемент метаданных (Metadata Item) – общий термин для единицы метаданных.

Ядро (Essence) – абстрактный термин, описывающий любые данные или сигнал, необходимый для представления визуального, слухового или какого-либо другого сенсорного ощущения одного типа, причем независимо от метода кодирования. Также определяется "Целевой группой по гармонизированным рекомендациям для обмена программным материалом в виде цифровых потоков" (TFHS) SMPTE/EBU как информация в виде видео-, аудиоинформации и/или данных. Ядро может также включать графику, телеметрию, фотографии и другую информацию.

Дополнение В (для сведения)

Кодирование длины согласно нотации ASN.1 BER

ПРИМЕЧАНИЕ 1. – Термин "байт", используемый в настоящей Рекомендации, является синонимом термина "октет", используемого в ISO/IEC 8825-1.

Следующий раздел полностью заимствован из ISO/IEC 8825-1:

"8.1.3.3 В случае определенной формы октеты, описывающие длину, должны содержать один или несколько октетов и должны представлять число октетов в совокупности октетов, описывающей контент, используя либо короткую форму (см. п. 8.1.3.4) либо длинную форму (см. п. 8.1.3.5) как вариант отправителя.

ПРИМЕЧАНИЕ 2. – Короткая форма может использоваться только тогда, когда число октетов в совокупности октетов, описывающей контент, меньше или равно 127".

"8.1.3.4 В случае короткой формы совокупность октетов, описывающая длину, должна состоять из одного октета, в котором бит 8 равен нулю, а биты 7–1 кодируют число октетов в совокупности октетов (которые могут быть нулевыми), описывающей контент (значение) как двоичное целое число без знака, в котором бит 7 является самым старшим битом.

Пример

$L = 38$ можно закодировать как 00100110_2 ".

"8.1.3.5 В случае длинной формы совокупность октетов, описывающая длину, должна содержать начальный октет и один или несколько последующих октетов. Начальный октет должен кодироваться следующим образом:

- a) бит 8 должен быть 1;
- b) биты 7–1 должны кодировать число последующих октетов в последовательности октетов, описывающей длину, как двоичное целое число без знака, в котором бит 7 является самым старшим битом;
- c) значение 11111111_2 использоваться не должно.

ПРИМЕЧАНИЕ 3. – Это ограничение введено с учетом возможных расширений в будущем.

Биты 8–1 первого последующего байта, за которыми следуют биты 8–1 второго последующего байта, за которыми, в свою очередь, следуют биты 8–1 каждого последующего байта, пока не появится последний последующий байт, должны кодироваться как двоичное целое число без знака, равное числу байтов в поле "значение", в котором бит 8 первого последующего байта является самым старшим битом.

ПРИМЕЧАНИЕ 4. – Это иногда называют "прямым порядком байтов".

Пример

$L = 201$ можно закодировать как:

Байт 1 = 10000001_2 , байт 2 = 11001001_2 [бит 7 бит 0].

Дополнение С (для сведения)

Кодирование значения идентификатора объекта согласно нотации ASN.1 BER

ПРИМЕЧАНИЕ 1. – Термин "байт", используемый в настоящей Рекомендации, является синонимом термина "октет", используемого в ISO/IEC 8825-1.

Следующий раздел полностью заимствован из ISO/IEC 8825-1:

8.19 Кодирование значения идентификатора объекта

8.19.1 Кодирование значения идентификатора объекта должно быть простым.

8.19.2 Оклеты, описывающие контент, должны представлять собой (упорядоченный) список операций кодирования субидентификаторов (см. 8.19.3 и 8.19.4), подвергнутых конкатенации.

Каждый субидентификатор представлен рядом октетов (одним или несколькими). Бит 8 в каждом октете показывает, является ли он последним в ряду: бит 8 последнего октета равен 0; бит 8 в каждом предыдущем октете равен 1. Биты 7–1 октетов в ряду совместно кодируют субидентификатор. Концептуально эти группы битов подвергнуты конкатенации для формирования двоичного числа без знака, у которого бит 7 первого октета является самым старшим, а бит 1 последнего октета является самым младшим. Субидентификатор должен кодироваться с привлечением как можно меньшего числа октетов, то есть ведущий октет субидентификатора не должен иметь значение 80_{16} .

8.19.3 Число субидентификаторов (N) должно быть на единицу меньше числа компонентов идентификатора объекта в значении идентификатора объекта, подвергаемого кодированию.

8.19.4 Числовое значение первого субидентификатора получают на основе значений первых **двух** компонентов идентификаторов объектов в значении идентификатора объекта, подвергаемого кодированию, используя формулу:

$$(X*40) + Y,$$

где X – значение первого компонента идентификатора объекта, а Y – значение второго компонента идентификатора объекта.

ПРИМЕЧАНИЕ. – Эта упаковка первых двух компонентов идентификаторов объектов является признанием того, что корневым узлом назначены только три значения, а 39 (максимум) последующих значений от узлов доступны при $X = 0$ и $X = 1$.

8.19.5 Числовое значение i -го субидентификатора, ($2 \leq i \leq N$) равно $(i + 1)$ -му компоненту идентификатора объекта.

Пример

Значение ИДЕНТИФИКАТОРА ОБЪЕКТА:

{joint-iso-itu-t 100 3},

что не отличается от:

{2 100 3}

имеет первый субидентификатор 180 и второй субидентификатор 3. Результирующее кодирование следующее:

ИДЕНТИФИКАТОР

ОБЪЕКТА	Длина	Контент
06 ₁₆	03 ₁₆	813403 ₁₆
