الاتحـاد الدولـي للاتصـالات
قطاع الاتصالات الراديوية

منشورات ITU
التوصيات

التوصية ITU-R BS.2088-2

(2025/11)

السلسلة BS: الخدمة الإذاعية (الصوتية)

نسق الملفات الطويلة للتبادل الدولي لمواد البرامج السمعية مع بيانات شرحية

تمهيد

يضطلع قطاع الاتصالات الراديوية بدور يتمثل في تأمين الترشيد والإنصاف والفعالية والاقتصاد في استعمال طيف الترددات الراديوية في جميع خدمات الاتصالات الراديوية، بما فيها الخدمات الساتلية، وإجراء دراسات دون تحديد لمدى الترددات، تكون أساساً لإعداد التوصيات واعتمادها.

ويؤدي قطاع الاتصالات الراديوية وظائفه التنظيمية والسياساتية من خلال المؤتمرات العالمية والإقليمية للاتصالات الراديوية وجمعيات الاتصالات الراديوية بمساعدة لجان الدراسات.

حقوق الملكية الفكرية

يسترعي الاتحاد الانتباه إلى أن تطبيق هذه التوصية أو تنفيذها قد يستلزم استعمال حق من حقوق الملكية الفكرية. ولا يتخذ الاتحاد أي موقف من القرائن المتعلقة بحقوق الملكية الفكرية أو صلاحيتها أو نطاق تطبيقها سواء طالب بها عضو من أعضاء الاتحاد أو طرف آخر لا تشمله عملية إعداد التوصيات.

وعند الموافقة على هذه التوصية، كان الاتحاد قد تلقى إخطاراً بملكية فكرية تحميها براءات الاختراع يمكن المطالبة بها لتنفيذ هذه التوصية. ومع ذلك، ونظراً إلى أن هذه المعلومات قد لا تكون هي الأحدث، يوصى المسؤولون عن تنفيذ هذه التوصية بالاطلاع على معلومات براءات الاختراع ذات الصلة لقطاع الاتصالات الراديوية المتاحة في الموقع الإلكتروني: https://www.itu.int/en/ITU-R/study-groups/Pages/itu-r-patent-information.aspx.

سلاسل توصيات قطاع الاتصالات الراديوية

(يمكن الاطلاع عليها أيضاً في الموقع الإلكتروني https://www.itu.int/publ/R-REC/ar)

السلسلة	العنوان
BO	البث الساتلي
BR	التسجيل من أجل الإنتاج والأرشفة والعرض؛ الأفلام التلفزيونية
BS	**الخدمة الإذاعية (الصوتية)**
BT	الخدمة الإذاعية (التلفزيونية)
F	الخدمة الثابتة
M	الخدمة المتنقلة وخدمة التحديد الراديوي للموقع وخدمة الهواة والخدمات الساتلية ذات الصلة
P	انتشار الموجات الراديوية
RA	علم الفلك الراديوي
RS	أنظمة الاستشعار عن بُعد
S	الخدمة الثابتة الساتلية
SA	التطبيقات الفضائية والأرصاد الجوية
SF	تقاسم الترددات والتنسيق بين أنظمة الخدمة الثابتة الساتلية والخدمة الثابتة
SM	إدارة الطيف
SNG	التجميع الساتلي للأخبار
TF	إرسالات الترددات المعيارية وإشارات التوقيت
V	المفردات والمواضيع ذات الصلة

ملاحظة: *تمت الموافقة على النسخة الإنكليزية لهذه التوصية الصادرة عن قطاع الاتصالات الراديوية بموجب الإجراء الموضح في القرار ITU-R 1.*

النشر الإلكتروني
جنيف، 2026

© ITU 2026
جميع حقوق النشر محفوظة. لا يمكن استنساخ أي جزء من هذه المنشورة بأي شكل كان ولا بأي وسيلة إلا بإذن خطي من الاتحاد الدولي للاتصالات (ITU).

التوصية ITU-R BS.2088-2*

نسق الملفات الطويلة للتبادل الدولي لمواد البرامج السمعية مع بيانات شرحية

(2025-2019-2015)

مجال التطبيق

تتضمن هذه الوثيقة مواصفات نسق الملف السمعي للموجات الإذاعية 64 بتة (BW64) بما في ذلك المقاطع الجديدة <ds64> و<axml> و<bxml> و<sxml> و<chna> التي تمكّن الملف من حمل ملفات كبيرة متعددة القنوات وبيانات شرحية بما في ذلك نموذج التعريف السمعي (ADM) الذي يرد وصفه في التوصية ITU-R BS.2076.

مصطلحات أساسية

ملف، نسق الملف، بيانات شرحية، موجة، BW64، تبادل، برنامج سمعي، WAV، BWF، RIFF، RF64، ملف موجي، الإشارة السمعية الغامرة، نموذج التعريف السمعي (ADM)، نموذج التعريف السمعي التسلسلي (S-ADM)

إن جمعية الاتصالات الراديوية للاتحاد الدولي للاتصالات،

إذ تضع في اعتبارها

أ) أن وسائط التخزين المعتمدة على تكنولوجيا المعلومات، بما فيها أقراص البيانات وأشرطتها اجتاحت كافة مجالات الإنتاج السمعي للإذاعة الراديوية وخصوصاً منها مجال التحرير غير الخطي والعرض على الهواء والأرشفة؛

ب) أن هذه التكنولوجيا توفر مزايا هامة من ناحية المرونة في التشغيل، وتدفق الإنتاج وأتمتة المحطات، وهي بالتالي على قدر كبير من الأهمية بالنسبة إلى تحديث الاستوديوهات الموجودة وتصميم منشآت جديدة للاستوديوهات؛

ج) أن اعتماد نسق ملف وحيد من أجل تبادل الإشارات البيني سيبسّط قابلية التشغيل البيني للتجهيزات الفردية والاستوديوهات البعيدة إلى حد كبير، وأنه سيمكن من تسهيل التكامل المرغوب لعمليات التحرير والعرض على الهواء والأرشفة؛

د) أن الملف يجب أن يحتوي على مجموعة دنيا من المعلومات المتصلة بالإذاعة من أجل توثيق البيانات الشرحية المتعلقة بالإشارات السمعية؛

هـ) أنه ضماناً للتوافق بين التطبيقات مختلفة التعقيدات، لا بد من الاتفاق على مجموعة دنيا من الوظائف المشتركة بين جميع التطبيقات تكون قادرة على معالجة نسق الملف الموصى به؛

و) أن التوصية ITU-R BS.646 تعرّف النسق السمعي الرقمي المستخدم في الإنتاج السمعي من أجل الإذاعة الصوتية والتلفزيونية؛

ز) أن التوافق مع أنساق الملفات التجارية المتيسرة حالياً قد يُقلص إلى أدنى حد المجهودات الصناعية المطلوبة من أجل تنفيذ ذلك النسق في التجهيزات؛

ح) أن وجود نسق قياسي لتشفير المعلومات التاريخية والبيانات الشرحية الأخرى ذات الصلة سيسهل استعمال هذه المعلومات بعد تبادل البرامج؛

* أدخلت لجنة الدراسات 6 للاتصالات الراديوية في عام 2026 تعديلات صياغية على هذه التوصية وفقاً للقرار ITU-R 1.

ط) أن نوعية الإشارة السمعية تتأثر بالمعالجة التي تجرى على الإشارة، خاصة من جراء استعمال تشفير وفك تشفير غير خطيّين خلال عمليات خفض معدل البتات؛

ي) أن الأنظمة الراديوية المتقدمة تتطلب بيانات شرحية مصاحبة للعناصر السمعية المقرر نقلها في الملف؛

ك) أن الأنظمة الراديوية المتقدمة تستخدم مجموعة من التشكيلات متعددة القنوات بما في ذلك العناصر السمعية القائمة على الأشياء والقنوات والمشاهد كما هو محدد في التوصية ITU-R BS.2051؛

ل) أن البيانات الشرحية المتعلقة بالبيانات السمعية المستعملة في الأنظمة الصوتية المتقدمة موصَّفة في التوصية ITU-R BS.2076 مع تعاريفها المشتركة الموصَّفة في التوصية ITU-R BS.2094 والتمثيل التسلسلي للبيانات الشرحية (S-ADM) الموصَّف في التوصية ITU-R BS.2125؛

م) أن التوصية ITU-R BS.1352 تنطوي على قيود فيما يتعلق بمقاس الملف وقدرته على نقل بيانات شرحية إضافية؛

ن) أن الملفات السمعية متعددة القنوات يمكن أن يكون مقاسها أكبر من 4 جيغابايت،

توصي

1 بأن يتم وضع معلمات الإشارات السمعية وتردد الاعتيان (الجزء 1) وعمق البت (الجزآن 4 و5) والتشديد المسبق (الجزء 6) من أجل تبادل البرامج السمعية وفقاً للأجزاء المعنية من التوصية ITU-R BS.646؛

2 بأن يتم استخدام نسق الملف المحدد في الملحق 1 من أجل التبادل البيني للبرامج السمعية في حالات الاستعمال التالية:

- في البيئات القائمة على ملف WAVE، حيث ترغب التطبيقات الإذاعية القائمة على ملف WAVE في رفع مرتبتها للتعامل مع محتوى غامر، مع الحفاظ في الوقت نفسه على التوافق المباشر؛
- في مسارات العمل القائمة على الملف، حيث توجد مكتبة مختلطة من المحتوى التقليدي القائم على ملف WAVE ومحتوى غامر؛
- في مسارات العمل القائمة على الملف، حيث يفضل غلاف للبيانات أحادية الرزمة مع البيانات الشرحية.

ملاحظة – يبين الملحق 4 التغييرات التي أدخلت على المواصفات الواردة في الملحق 1 من النسخة السابقة من هذه التوصية وهي للعلم فقط.

الملحق 1
(معياري)

مواصفات نسق الملف BW64

1 مقدمة

يعتمد نسق الموجات الإذاعية (BW64) على نسق الملف السمعي WAVE (الذي يرد وصفه في الملحق 2)، وهو نمط من الملفات محدد في نسق ملف تبادل الموارد (RIFF). وتحتوي ملفات WAVE أساساً على بيانات سمعية. وتحتوي كتلة البناء الأساسية لنسق الملف RIFF، المسماة بالمقطع، على زمرة من قطع المعلومات وثيقة الصلة ببعضها البعض. وتتكون هذه الكتلة من معرف المقطع وعدد صحيح يمثل طول البايتات والمعلومات المحمولة. ويتكون الملف RIFF من مجموعة من المقاطع. ويستخدم نسق BW64 هذا العناصر الأساسية للنسق كما هو محدد في المعيار التقني لاتحاد الإذاعة الأوروبي EBU Tech 3306.

ويتسم نسق الملف WAVE، الوارد في التوصية ITU-R BS.1352، بعدد من القيود، وأبرزها:

- المقاس الأقصى للملف أقل من 4 جيغابايت.
- لا يوجد دعم للعناصر السمعية المتقدمة متعددة القنوات بسبب محدودية البيانات الشرحية المتعلقة بالإشارة السمعية.
- دعم غير كافٍ للبيانات الشرحية التقنية.

ويرمي النسق BW64 الوارد في هذه التوصية إلى تجاوز هذه القيود والحفاظ على أكبر قدر ممكن من التوافق مع نسق التوصية ITU-R BS.1352 مع تقاسم العديد من العناصر الأساسية.

وهناك طلب متزايد على نقل البيانات الشرحية، وخصوصاً نقل البيانات الشرحية لنموذج التعريف السمعي (ADM) وفقاً للتوصية ITU-R BS.2076. وتتضمن هذه التوصية تعريفاً للمقاطع <axml> و<bxml> و<sxml> الخاصة بتخزين ونقل البيانات الشرحية على شكل ملف XML مكتوب بتشفير UTF-8، بنسق مضغوط وتسلسلي، على التوالي.

والغرض الأولي من المقطع <chna> الوارد في هذه التوصية هو توفير المراجع من كل مسار في ملف BW64 إلى معرفات الهوية في البيانات الشرحية لنموذج التعريف السمعي (ADM) المحددة في التوصية ITU-R BS.2076.

وباستثناء الهدف الأولي من ربط كل مسار في الملف بالبيانات الشرحية لنموذج التعريف السمعي المرتبطة به، يسمح المقطع <chna> أيضاً بالنفاذ على نحو أسرع إلى معرفات هوية نموذج التعريف السمعي (ADM ID) دون الحاجة إلى النفاذ إلى البيانات الشرحية XML (إذا كانت معرفات الهوية ضمن مجموعة من القيم المحددة مسبقاً لتشكيلات نموذج التعريف السمعي المعيارية). وبما أنه يمكن تثبيت مقاس المقطع <chna>، وأنه يوضع قبل h لمقاطع <data> و<axml> و<bxml> و<sxml> فمن السهل النفاذ إلى محتوياته بسرعة، أو توليدها أو تعديلها.

وتستخدم أنماط البيانات في هذه الوثيقة بأكملها وفقاً للملحق 3.

2 وصف النسق BW64

1.2 محتويات ملف النسق BW64

يتعين أن يبدأ ملف النسق BW64 بالرأسية "WAVE" الإلزامية وبالمقاطع التالية على الأقل:

```
<WAVE-form> ->
    BW64('WAVE'
        <ds64-ck>       // ds64 chunk for 64-bit addressing
        <fmt-ck>        // Format of the audio signal: PCM/non-PCM
        <chna-ck>       // chna chunk for ADM look-up
        <axml-ck>       // axml chunk for carrying XML metadata
        <bxml-ck>       // bxml chunk for carrying compressed XML metadata
        <sxml-ck>       // sxml chunk for carrying XML metadata associated with
                        // sub-chunk or sound data
        <wave-data>)    // sound data
```

الملاحظة 1 – يمكن وجود مقاطع إضافية في الملف. وقد يقع بعض من هذه المقاطع الإضافية خارج نطاق هذه التوصية. ويمكن للتطبيقات أن تقوم أو لا تقوم بترجمة هذه المقاطع أو الاستفادة منها، حيث إنه لا يمكن ضمان سلامة البيانات المتضمنة في هذه المقاطع المجهولة. ومع ذلك، يجب على التطبيقات الممتثلة أن تمرر المقاطع المجهولة بشفافية. وقد تحتوي ملفات BW64 المحددة في هذه التوصية على مقاطع أخرى، بما في ذلك المقاطع <bext> و<ubxt> المنشأة من BWF. ومع ذلك، يمكن للتطبيقات المتوافقة مع التوصية ITU-R BS.2088 (BW64) قراءة بيانات XML الوصفية التي قد تحتوي أيضاً على بيانات وصفية مُحولة من مقاطع قديمة، بما في ذلك مقاطع BWF <bext> و<ubxt>.

الملاحظة 2 – قد يسمح بوضع المقطع <axml> أو <bxml> أو <sxml> بعد المقطع <data>، لأن طول البيانات الشرحية XML يحتمل أن يكون مجهولاً وقد يكون الموقع المعروف لبدء العينات السمعية في الملف ذا طابع عملي أكبر.

2.2 المقاطع الموجودة المعرفة كجزء من المعيار RIFF/WAVE

يَستخدم المعيار RIFF/WAVE عدداً من المقاطع التي جرى تعريفها بالفعل. وهي كما يلي:

- <RIFF>
- <fmt>
- <data>

ويرد وصف هذه المقاطع في الفقرات من 1.6.2 إلى 3.6.2.

والمعيار RIFF/WAVE هو مجموعة فرعية من النسق موضوع التوصية ITU-R BS.1352. وتتضمن التوصية ITU-R BS.1352 هذين المقطعين التاليين:

- <bext>
- <ubxt>

يمكن إدراج هذه المقاطع في النسق BW64. ومع ذلك، ينبغي تحويل المحتوى الموجود في هذه المقاطع إلى بيانات وصفية XML موجودة في مقطع XML ذي الصلة (انظر الفقرتين 10 و11). وينبغي للتطبيقات استخدام البيانات الوصفية XML. وتتاح معلومات حول استخدام النسق BW64 في التقرير ITU-R BS.2388.

3.2 المقاطع والبنى الجديدة في النسق BW64

المقاطع الجديدة التي أدخلت على النسق BW64 هي:

- <BW64>
- <ds64>
- <JUNK>
- <axml> أو <bxml> أو <sxml>
- <chna>

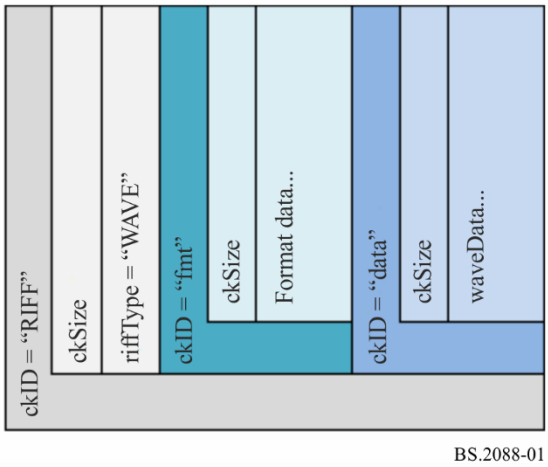
ويرد وصف لهذه المقاطع في الفقرات من 3 إلى 8.

4.2 استخدام المقطع <ds64> للتمكن من استعمال ملفات بمقاس أكبر من 4 جيغابايت

يرجع السبب في وجود حاجز بقيمة 4 جيغابايت إلى حيز العناوين المكون من 32 بتّة في النسق RIFF/WAVE ونسق الموجات الإذاعية BWF. فبواسطة 32 بتة يمكن تحقيق مقاس أقصى من العناوين بقيمة 4294967296 بايت = 4 جيغابايت. ولحل هذه المسألة، يلزم وجود حيز عناوين مكون من 64 بتة. ويبين الشكل 1 هيكل ملف تقليدي أساسي RIFF/WAVE، حيث الحقول kSize هي أعداد مكونة من 32 بتة تمثل أحجام المقاطع الخاصة بها.

الشكل 1

الهيكل الأساسي لملف RIFF/WAVE



ومن شأن مجرد تغيير مقاس كل حقل في ملف الموجات الإذاعية (BWF) إلى 64 بتة أن يعطي ملفاً لا يكون مطابقاً لنسق المعيار RIFF/WAVE – وهذه ملاحظة بديهية ولكن هامة.

وينص النهج المتبع على تعريف نسق جديد على أساس 64 بتة يدعى النسق BW64 ويكون مماثلاً للنسق RIFF/WAVE الأصلي، باستثناء التغيرات التالية:

- يستخدم معرف الهوية 'BW64' بدلاً من 'RIFF' في أول أربع بايتات من الملف
- يضاف مقطع إلزامي <ds64> (مقاس البيانات 64 بتة)، ويجب أن يكون المقطع الأول بعد "المقطع BW64".

يتميز المقطع 'ds64' بقيمتين إلزاميتين مكونتين من عدد صحيح من 64 بتة، يستعاض بهما عن حقلين مكونين من 32 بتة في النسق RIFF/WAVE:

- bw64Size (يستعاض به عن حقل المقاس في المقطع <RIFF>)
- dataSize (يستعاض به عن حقل المقاس في المقطع <data>)

ولجميع الحقول المكونة من 32 بتة في النسق RIFF/WAVE تطبق القاعدتان التاليتان:

إذا كانت القيمة المكونة من 32 بتة في الحقل لا تساوي 0xFFFFFFFF عندئذ تستعمل هذه القيمة المكونة من 32 بتة.

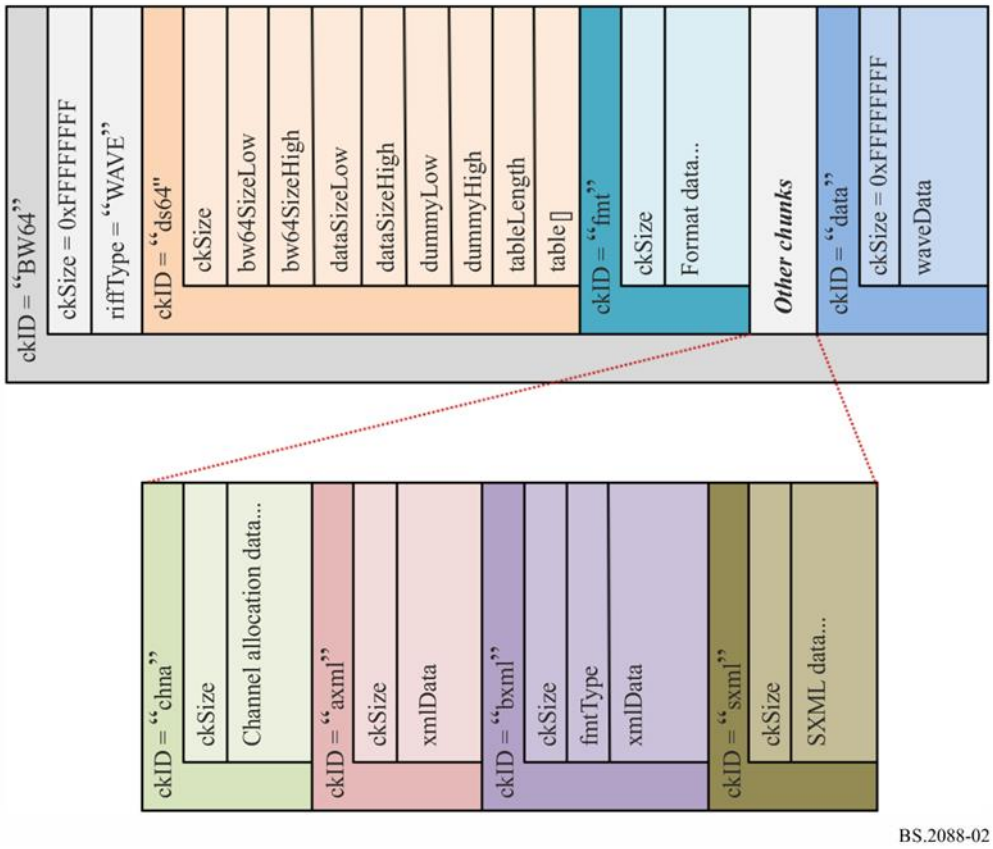
وإذا كانت القيمة المكونة من 32 بتة تساوي 0xFFFFFFFF تستعمل بدلاً منها القيمة المكونة من 64 بتة في المقطع 'ds64'.

- يمكن استعمال صفيف اختياري واحد من البنى (انظر الملحق 1) بأحجام مقاطع إضافية مكونة من 64 بتة.

ويوضح الشكل 2 الهيكل الكامل لنسق الملف BW64، حيث تضبط قيم ckSize للمقطعين <BW64> و<data> على 0xFFFFFFFF للسماح لها باستخدام قيم المقاس المكونة من 64 بتة المأخوذة من المقطع <ds64>.

الشكل 2

هيكل الملف BW64



ملاحظة – يمكن أن تكون مقاسات بيانات المقاطع متغيرة. وتتحاذى بداية كل مقطع بالكلمات فيما يتعلق ببداية ملف BW64 للحفاظ على التوافق مع نسق BWF الموصَّف في التوصية ITU-R BS.1352. وإذا كان مقاس المقطع عدداً مفرداً من البايتات، تُكتب بايتة الحشو بقيمة صفر بعد المقطع. ولكن قيمة ckSize لا تتضمن بايتة الحشو.

5.2 تحقيق التوافق بين RIFF/WAVE وBW64

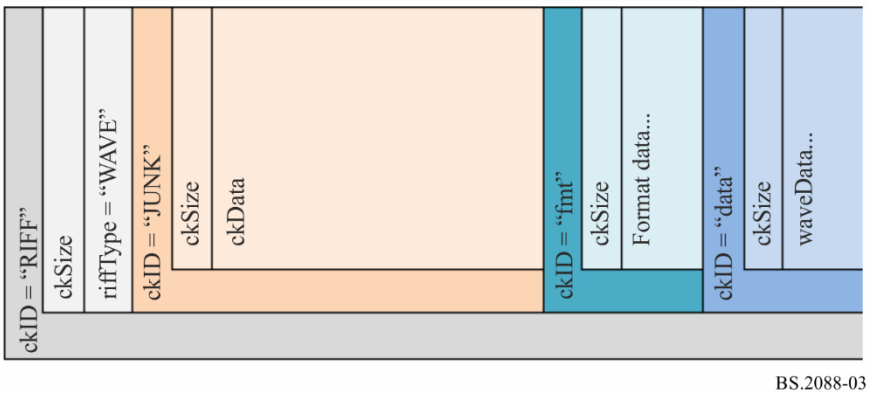
على الرغم من ترددات الاعتيان العالية والإشارة السمعية متعددة القنوات، تكون بعض ملفات الإنتاج السمعي حتماً أصغر من 4 جيغابايت ولذلك ينبغي أن تبقى في نسق RIFF/WAVE المختصر (كما هو وارد في الملحق 2). وتنشأ المشكلة المتمثلة بأن أحد تطبيقات التسجيل لا يتمكن بشكل مسبق من معرفة ما إذا كانت الإشارة السمعية المسجلة التي يقوم بتجميعها ستتجاوز 4 جيغابايت أم لا في نهاية التسجيل (أي ما إذا كان بحاجة إلى استخدام النسق BW64 أم لا).

ويكون الحل بتمكين تطبيق التسجيل من التحول على الفور من النسق RIFF/WAVE إلى النسق BW64 بحد للمقاس قدره 4 جيغابايت، فيما لا يزال التسجيل جارياً.

ويتحقق ذلك من خلال عكس الحيز الإضافي في النسق RIFF/WAVE عن طريق إدراج مقطع <JUNK> يكون بنفس مقاس المقطع <ds64>. وليس لهذا الحيز المعكوس أي معنى بالنسبة للنسق WAVE المختصر، ولكنه يتحول إلى المقطع <ds64> إذا كان الانتقال إلى النسق BW64 ضرورياً. ويبين المخطط في الشكل 3 مقطع الخانة المحجوزة <JUNK> الذي يوضع قبل المقطع <fmt>.

الشكل 3

هيكل الملف مع المقطع JUNK



عند بداية التسجيل، ينشئ تطبيق متنبه للنسق BW64 نسقاً معيارياً RIFF/WAVE مع المقطع ’JUNK‘ كمقطع أول. ويتحقق أثناء التسجيل من أحجام النسق RIFF والبيانات. فإذا تجاوزت 4 جيغابايت، يقوم التطبيق بما يلي:

- الاستعاضة عن معرّف المقطع <JUNK> بمقطع <ds64>. (يحوّل ذلك المقطع <JUNK> السابق إلى مقطع <ds64>).
- إدراج مقاس الملف RIFF ومقاس المقطع ’data‘ في المقطع <ds64>
- ضبط مقاس الملف RIFF ومقاس المقطع ’data‘ في الحقول المكونة من 32 بتة على القيمة 0xFFFFFFFF
- الاستعاضة عن معرف ’RIFF‘ بالنسق ’BW64‘ في البايتات الأربع الأولى من الملف
- مواصلة التسجيل.

6.2 المقاطع والبنى الموجودة في النسق RIFF/WAVE

تظهر أدناه المقاطع الموجودة في النسق RIFF/WAVE:

```
struct RiffChunk            // declare RiffChunk structure
{
    CHAR   ckID[4];         // ‘RIFF’
    DWORD  ckSize;          // 4 byte size of the traditional RIFF/WAVE file
    CHAR   riffType[4];     // ‘WAVE’
};

struct FormatChunk          // declare FormatChunk structure
{
    CHAR   ckID[4];         // ‘fmt ’
    DWORD  ckSize;          // 4 byte size of the ‘fmt ’ chunk
    WORD   formatTag;       // WAVE_FORMAT_PCM = 0x0001, etc.
    WORD   channelCount;    // 1 = mono, 2 = stereo, etc.
    DWORD  sampleRate;      // 32000, 44100, 48000, etc.
    DWORD  bytesPerSecond;  // only important for compressed formats
    WORD   blockAlignment;  // container size (in bytes) of one set of samples
    WORD   bitsPerSample;   // valid bits per sample 16, 20 or 24
    WORD   cbSize;          // should be excluded as extraData is not used, but if
                            // present shall be set to zero
    CHAR   extraData[22];   // extra data of WAVE_FORMAT_EXTENSIBLE when necessary,
```

```
                        // shall not be used as cbSize will be zero or not present.
};

struct DataChunk            // declare DataChunk structure
{
    CHAR   ckID[4];         // 'data'
    DWORD  ckSize;          // 4 byte size of the 'data' chunk
    CHAR   waveData[ ];     // audio samples
};
```

وتشير أقواس الصفيف الفارغة إلى إمكانية استخدام عدد متغير من العناصر (بما في ذلك الصفر).

1.6.2 عناصر المقطع <RIFF>

المقطع <RIFF> هو المستوى الأعلى للملف.

الحقل	الوصف
ckID	صفيف من أربع سمات {'R', 'I', 'F', 'F'} يستخدم لتحديد هوية المقطع.
ckSize	قيمة من 4 بايت لمقاس الملف.
riffType	صفيف من أربع سمات {'W', 'A', 'V', 'E'} يدل على أن الملف هو ملف سمعي من النمط WAVE.

2.6.2 عناصر المقطع <fmt>

يحتوي المقطع <fmt> على معلومات عن أنساق العينات السمعية المخزنة في المقطع <data>.

الحقل	الوصف
ckID	صفيف من أربع سمات {'f', 'm', 't', ' '} يستخدم لتحديد هوية المقطع.
ckSize	قيمة من 4 بايتات لمقاس الملف.
formatTag	قيمة من 2 بايت تمثل نسق العينات السمعية. وتعني القيمة 0x0001 أن النسق هو تشكيل شفري نبضي (PCM)، وتخصص القيمة 0x0000 للأنساق المجهولة.
channelCount	قيمة من 2 بايت تدل على عدد المسارات السمعية في الملف.
sampleRate	قيمة من 4 بايت تدل على معدل العينات السمعية بالهرتز (Hz).
bytesPerSecond	متوسط عدد البايتات التي يتعين نقل بيانات الموجة بها في الثانية الواحدة. وباستخدام تلك القيمة يمكن لبرمجيات استعادة التسجيل تقدير مقاس الدارئ.
blockAlignment	تراصف كتل بيانات الموجة (بالبايتات). وتحتاج برمجيات استعادة التسجيل إلى معالجة بايتات متعددة من **blockAlignment** من البيانات في المرة الواحدة، بحيث يمكن استخدام القيمة **blockAlignment** لتراصف الدارئ.
bitsPerSample	عدد البتات في العينة الواحدة وفي القناة الواحدة. ويفترض أن يكون لكل القنوات نفس استبانة العينات. وإذا لم يكن هذا الحقل ضرورياً، يتعين ضبطه على الصفر.
cbSize	مقاس هيكل المعلومات الإضافية extraData بالبايتات.
extraData	معلومات إضافية تستخدم لتخزين معلومات النسق WAVE_FORMAT_EXTENSIBLE. ويجب عدم استعمالها في النسق BW64.

والمقطع FormatChunk هو بالفعل مقطع نسق متخصص للبيانات السمعية PCM.

يستخدم الصفيف extraData في المقطع FormatChunk عندما يضبط formatTag على القيمة 0xFFFE (WAVE_FORMAT_EXTENSIBLE). وبما أنه ينبغي وصف الإشارات السمعية متعددة القنوات باستخدام البيانات الشرحية لنموذج التعريف السمعي (ADM)، فإنه ينبغي تفادي استخدام formatTag هذا. ومع ذلك، ينبغي أن يكون التنفيذ قادراً على التعامل مع قراءة ملف يتضمن formatTag هذا ومعالجته بطريقة صحيحة.

ولضمان عدم تناقض FormatChunk مع المعلومات المتعلقة بالمقاطع <chna> و<axml> و<bxml> و<sxml>، يوصى بضبط formatTag على القيمة 0x0001 للإشارة السمعية PCM، وعلى القيمة 0x0000 (formatTag = غير معروف) لجميع الإشارات السمعية الأخرى خلاف إشارات PCM.

3.6.2 عناصر المقطع <data>

يخصص المقطع <data> لتخزين العينات السمعية.

الحقل	الوصف
ckID	صفيف من أربع سمات {'d', 'a', 't', 'a'} يستخدم لتحديد هوية المقطع.
ckSize	قيمة من 4 بايتات لمقاس المقطع.
waveData	المكان الذي تخزن فيه العينات السمعية. تخزن العينات بترتيب البايتات الأقل أهمية. وتخزن المسارات المتعددة بالتشذير على أساس كل عينة على حدة. فعلى سبيل المثال، في حالة إشارة سمعية من 16 بتة ومسارين:

البايت	العينة	المسار
0	0 – LSB	1
1	0 – MSB	1
2	0 – LSB	2
3	0 – MSB	2
4	1 – LSB	1
5	1 – MSB	1
6	1 – LSB	2
7	1 – MSB	2

3 مقطع المستوى الأعلى للنسق BW64

1.3 تعريف

يستخدم مقطع المستوى الأعلى للنسق <BW64> بدلاً من المقطع <RIFF> المستخدم في ملفات بمقاس 32 بتة. وتعني قراءه هذا المقطع أنه يتعين وجود المقطع <ds64> لقراءة الأحجام المكونة من 64 بتة. ويظهر أدناه المقطع <BW64>:

```
struct BW64Chunk            // declare BW64Chunk structure
{
    CHAR ckID[4];           // 'BW64'
    DWORD ckSize;           // 0xFFFFFFFF means don't use this data, use
                            // bw64SizeHigh and bw64SizeLow in 'ds64' chunk instead
    CHAR BW64Type[4];       // 'WAVE'
};
```

2.3 عناصر المقطع <BW64>

الحقل	الوصف
ckID	صفيف من أربع سمات {‘4’, ‘6’, ‘w’, ‘b’} يستخدم لتحديد هوية المقطع.
ckSize	قيمة من 4 بايتات يتعين ضبطها على القيمة 0xFFFFFFFF للدلالة على أن قيمة المقاس هذه ليست مستخدمة وأنه يتعين استخدام المقطع <ds64> لتحديد الأحجام.
BW64Type	صفيف من أربع سمات {‘E’, ‘V’, ‘A’, ‘W’} يدل على أن الملف هو ملف سمعي من النمط WAVE.

4 المقطعان DS64 وJUNK

1.4 تعاريف

ينقل المقطع <ds64> قيماً مكونة من 64 بتة لمقاس الملف وللمقطع <data> ولصفيف من قيم بمقاس 64 بتة للمقاطع الأخرى القابلة للتعريف. ويظهر أدناه هيكل المقطع <ds64>، يليه هيكل الجدول **ChunkSize64** الذي ينقل أحجام المقاطع القابلة للتعريف (خلاف المقطع <data>). وتشير أقواس الصفيف الفارغة إلى إمكانية استخدام عدد متغير من العناصر (بما في ذلك الصفر).

```
struct DataSize64Chunk      // declare DataSize64Chunk structure
{
    CHAR ckID[4];           // ‘ds64’, FOURCC chunk identifier
    DWORD ckSize;           // 4 byte size of the <ds64> chunk
    DWORD bw64SizeLow;      // low 4 byte size of <BW64> chunk
    DWORD bw64SizeHigh;     // high 4 byte size of <BW64> chunk
    DWORD dataSizeLow;      // low 4 byte size of <data> chunk
    DWORD dataSizeHigh;     // high 4 byte size of <data> chunk
    DWORD dummyLow;         // dummy value for cross compatibility
    DWORD dummyHigh;        // dummy value for cross compatibility
    DWORD tableLength;      // number of valid entries in array “table”
    ChunkSize64 table[ ];   // array of chunk sizes for chunks exceeding 4 Gbytes
};
struct ChunkSize64          // declare ChunkSize64 structure
{
    CHAR ckID[4];           // chunk ID of chunk which needs 64bit addressing;
                            // e.g. ‘axml’ is used when <axml> chunk exceeds 4 Gbytes
    DWORD ckSizeLow;        // low 4 byte chunk size
    DWORD ckSizeHigh;       // high 4 byte chunk size
};
```

والمقطع <JUNK> هو عبارة عن خانة محجوزة للمقطع <ds64> يستعمل إذا تم توليد ملف سمعي مكون من 32 بتة قد يحتاج إلى تحويل فوري إلى ملف مكون من 64 بتة لاحقاً. ويجب أن يكون مقاس المقطع <JUNK> مطابقاً لمقاس المقطع <ds64> المحتمل الذي يحل محله. ويظهر هنا هيكل المقطع:

```
struct JunkChunk            // declare JunkChunk structure
{
    CHAR   ckID[4];         // 'JUNK'
    DWORD  ckSize;          // 4 byte size of the 'JUNK' chunk. This must be at
                            // least 28 if the chunk is intended as a place-holder
                            // for a 'ds64' chunk.
    CHAR   ckData[];        // dummy bytes
};
```

2.4 عناصر المقطع <ds64>

الحقل	الوصف
ckID	صفيف من أربع سمات {'4', '6', 's', 'd'} يستخدم لتحديد هوية المقطع.
ckSize	مقاس المقطع <ds64> من 4 بايت.
bw64SizeLow	أدنى مقاس للمقطع <BW64> مكون من 4 بايت. ويعبر عن مقاس البيانات المكون من 64 بتة بقيمة 0xHHHHLLLL إذا كان <bw64SizeLow> و<bw64SizeHigh> يساويان 0xLLLL و0xHHHH، على التوالي. وتكون الكمية التي ليس لها علامة والمكونة من 32 بتة بنسق الترتيب الأقل أهمية.
bw64SizeHigh	أعلى مقاس للمقطع <BW64> مكون من 4 بايت. وتكون الكمية التي ليس لها علامة والمكونة من 32 بتة بنسق الترتيب الأقل أهمية.
dataSizeLow	أدنى مقاس للمقطع <data> مكون من 4 بايت. ويعبر عن مقاس البيانات المكون من 64 بتة بقيمة 0xHHHHLLLL إذا كان <dataSizeLow> و<dataSizeHigh> يساويان 0xLLLL و0xHHHH، على التوالي. وتكون الكمية التي ليس لها علامة والمكونة من 32 بتة بنسق الترتيب الأقل أهمية.
dataSizeHigh	أعلى مقاس للمقطع <data> مكون من 4 بايت. وتكون الكمية التي ليس لها علامة والمكونة من 32 بتة بنسق الترتيب الأقل أهمية.
dummyLow	قيمة وهمية من 4 بايت يتعين إهمالها عند قراءتها، وضبطها على الصفر عند كتابتها. وهي توجد لضمان التوافق مع مواصفة المعيار EBU Tech 3306 RF64، الذي يستخدم هذه القيمة لنقل معلومات متعلقة بمقاس المقطع <fact> لا توجد في النسق BW64.
dummyHigh	قيمة وهمية من 4 بايت يتعين تجاهلها عند قراءتها، وضبطها على الصفر عند كتابتها. والغرض منها هو نفسه الذي للقيمة <dummyLow>.
tableLength	عدد البنود الصالحة في الصفيف "ChunkSize64 table".
ChunkSize64 table	صفيف أحجام المقاطع التي تتجاوز 4 جيغابايت.

يحدد الجدول **ChunkSize64** على النحو التالي. يستخدم صفيف من بنى **ChunkSize64** لتخزين طول أي مقطع خلاف المقطع <data> في الجزء الاختياري من المقطع <ds64>. وحالياً، يرجح أن يكون المقطع <axml> هو النمط الوحيد من المقاطع خلاف المقطع <data> الذي يتجاوز مقاسه 4 جيغابايت (ربما في ملفات سمعية كبيرة جداً قائمة على الأشياء).

الحقل	الوصف
ckID	صفيف من 4 سمات يستخدم للإشارة إلى معرف المقطع <ckID> الذي يحتاج إلى حيز عناوين مكون من 64 بتة. وعلى سبيل المثال، يستخدم الصفيف {‘a’, ‘x’, ‘m’, ‘l’} للمقطع <axml>.
ckSizeLow	أدنى مقاس مكون من 4 بايت للمقطع الذي يشير إلى <ckID>. وتكون الكمية التي ليس لها علامة والمكونة من 32 بتة بنسق الترتيب الأقل أهمية.
ckSizeHigh	أعلى مقاس مكون من 4 بايت للمقطع الذي يشير إلى <ckID>. وتكون الكمية التي ليس لها علامة والمكونة من 32 بتة بنسق الترتيب الأقل أهمية.

3.4 عناصر المقطع <JUNK>

الحقل	الوصف
ckID	صفيف من أربع سمات {‘J’, ‘U’, ‘N’, ‘K’} يستخدم لتحديد هوية المقطع.
ckSize	مقاس للمقطع <JUNK> بقيمة 4 بايت. ويجب أن يكون مكوناً من 28 بتة على الأقل ليكون خانة محجوزة للمقطع <ds64>.
ckData	بيانات وهمية يتعين إهمالها.

5 المقطع AXML

1.5 تعريف

قد يتضمن المقطع <axml> أي بيانات متوافقة مع النسق XML 1.0 أو أي نسق بعده، وهو نسق واسع الانتشار لتبادل البيانات [1]. ويلاحظ أن المقطع <axml> قد يتضمن أجزاء من ملف XML مأخوذة من أكثر من مخطط. وقد ترد بأي ترتيب مع المقاطع RIFF الأخرى ضمن الملف نفسه.

ويتكون المقطع <axml> من رأسية تليها بيانات متوافقة مع نسق XML. ولا يكون الطول الإجمالي للمقطع ثابتاً.

انظر الفقرة 11 للاطلاع على مثال على كيفية استخدام المقطع <axml> في النسق BW64 لنقل البيانات الشرحية الإذاعية، بما في ذلك المعلمات الواردة في المقطعين <bext> و<ubxt> السابقين.

```
struct axml_chunk
{
    CHAR    ckID[4];        // {'a','x','m','l'}
    DWORD   ckSize;         // size of the <axml> chunk in bytes
    CHAR    xmlData[];     // text data in XML
};
```

وبما أن ملف XML قد يستوعب أكثر من 4 جيغابايت، فقد يكون من الضروري استخدام المقطع <ds64> لإتاحة حقل بمقاس 64 بتة للمقطع <axml>. ويرد أدناه شبه شفرة معينة لتوضيح كيفية تحقيق ذلك باستخدام صفيف الجدول في المقطع <ds64>.

```
DataSize64Chunk.tableLength = 1;   // number of valid entries in array "table"
DataSize64Chunk.table[0] = {
    ChunkSize64.ckID = {`a`, `x`, `m`, `l`};   // chunk ID for the <axml> chunk
     ckSizeLow = xxxx     // low 4 byte chunk size
     ckSizeHigh = xxxx    // high 4 byte chunk size
}
```

2.5 عناصر المقطع <axml>

ckID صفيف من أربع سمات {‘a’, ‘x’, ‘m’, ‘l’} يستخدم لتحديد هوية المقطع.

ckSize مقاس قسم البيانات في المقطع بالبايت. (لا يشمل قيمة 8 بايت التي يستخدمها كل من ckID وckSize.)

xmlData يتضمن هذا الحقل معلومات عن النص بلغة XML.

يكون هيكل بيانات XML بتراتيب تدرجي وتخزن البيانات في سلاسل النص وفقاً للنسق XML 1.0 أو أي نسق بعده.

وإذا لم يتمكن جهاز الاستقبال من تفسير محتوى المقطع <axml> وفقاً للمواصفة الواردة في ملف XML، يتعين إهمال المقطع بكامله.

6 المقطع BXML

1.6 تعريف

يجوز أن يحتوي المقطع <bxml> على بيانات XML المضغوطة بدلاً من المقطع <axml>.

ويتكون المقطع <bxml> من رأسية تليها بيانات XML مضغوطة بأسلوب الضغط المحدد في النمط **fmtType**. والطول الكلي للمقطع غير ثابت.

```
struct bxml_chunk
{
    CHAR    ckID[4];        // {'b','x','m','l'}
    DWORD   ckSize;         // size of the <bxml> chunk in bytes
    WORD    fmtType;        // type of compression method, 0x0001=”gzip”, etc.
    CHAR    xmlData[];     // XML text data compressed by the compression method
};
```

وبما أن بيانات XML المضغوطة يمكن أن تستوعب أكثر من 4 جيغابايتة، فقد تقتضي الضرورة استخدام المقطع <ds64> لإتاحة حقل مقاسه 64 بتة للمقطع <axml>. ويرد أدناه شبه شفرة معينة لتوضيح كيفية تحقيق ذلك باستخدام صفيف الجدول في المقطع <ds64>.

```
DataSize64Chunk.tableLength = 1;   // number of valid entries in array “table”
DataSize64Chunk.table[0] = {
    ChunkSize64.ckID = {`b`, `x`, `m`, `l`};   // chunk ID for the <bxml> chunk
     ckSizeLow = xxxx     // low-4-byte chunk size
     ckSizeHigh = xxxx    // high-4-byte chunk size
}
```

2.6 عناصر المقطع <bxml>

ckID صفيف من أربع سمات {‘b’, ‘x’, ‘m’, ‘l’} يستخدم لتحديد هوية المقطع.

ckSize مقاس قسم البيانات في المقطع بالبايت. (لا يشمل قيمة 8 بايت التي يستخدمها كل من ckID وckSize.)

fmtType قيمة بايتتين (2) تمثل أسلوب ضغط نص XML. وقيمة 0x0001 تعني أن أسلوب الضغط هو gzip (IETF RFC 1952). وتُستخدم قيمة 0x0000 لنص XML غير مضغوط.

xmlData يحتوي هذا الحقل على شفرة XML مضغوطة بأسلوب الضغط المشار إليها في نمط fmtType.

7 المقطع SXML

1.7 تعريف

يمكن أن يحتوي المقطع <sxml> على أي بيانات XML مضغوطة أو غير مضغوطة تتوافق مع نسق XML 1.0 أو ما بعده وترتبط بشرائح البيانات السمعية. ويمكن أن يحدث بأي ترتيب مع مقاطع RIFF الأخرى داخل نفس الملف.

ويتكون المقطع <sxml> من رأسية تليها مقاطع فرعية **(SubXMLChunk)** مع بيانات XML المضغوطة أو غير المضغوطة على النحو الموصَّف بواسطة **fmtType**. ويتوافق كل مقطع **SubXMLChunk** مع عدد فريد من العينات السمعية المتاخمة لمقاطع **SubXMLChunks** المجاورة. ويكتمل المقطع <sxml> بجدول نقاط محاذاة اختياري، مما يسمح بالنفاذ القائم على ختم زمني إلى **SubXMLChunk** المختار. والطول الكلي للمقطع <sxml> غير ثابت.

ويمكن استخدام المقطع <sxml> لنقل البيانات الشرحية المتغيرة بمرور الوقت، من قبيل، تمثيل تسلسلي لنموذج التعريف السمعي (ADM) الموصَّف في التوصية ITU-R BS.2125.

```
struct sxml_chunk
{
    CHAR    ckID[4];              // {'s','x','m','l'}
    DWORD   ckSize;               // size of the <sxml> chunk in bytes
    WORD    fmtType;              // type of compression method, 0x0001=”gzip”, etc.
    DWORD   subXMLCkTbSizeLow;    //  low  4  byte  of  size  of  nSubXMLChunks  +
                                  // SubXMLChunk table[]
    DWORD   subXMLCkTbSizeHigh;   //  high  4  byte  of  size  of  nSubXMLChunks  +
                                  // SubXMLChunk table[]
    DWORD   nSubXMLChunks;        // number of sub-chunks with XML data
    SubXMLChunk table[];          // array of sub-chunks with XML data
    DWORD   nAlignmentPoints;     // number of alignment points
    AlignmentPoint table[];       // array of alignment points
};

struct SubXMLChunk
{
    DWORD   subXMLChunkSize;      // size of SubXMLChunk in bytes
    DWORD   nSamplesSubDataChunk;// number of audio samples associated with SubXMLChunk
    CHAR    xmlData[];           // compressed or uncompressed XML data
};

struct AlignmentPoint
{
    DWORD  subXMLChunkByteOffsetLow;  // low 4 byte of SubXMLChunk byte offset
    DWORD  subXMLChunkByteOffsetHigh; // high 4 byte of SubXMLChunk byte offset
    DWORD  nSamplesAlignPointLow;     // low 4 byte of alignment point sample count
    DWORD  nSamplesAlignPointHigh;    // high 4 byte of alignment point sample count
};
```

وبما أن بيانات XML المضغوطة أو غير المضغوطة يمكن أن تستوعب أكثر من 4 جيغابايتة، فقد تقتضي الضرورة استخدام المقطع <ds64> لإتاحة حقل مقاسه 64 بتة للمقطع <sxml>. ويرد أدناه شبه شفرة معينة لتوضيح كيفية تحقيق ذلك باستخدام صفيف الجدول في المقطع <ds64>.

```
DataSize64Chunk.tableLength = 1;   // number of valid entries in array “table”
DataSize64Chunk.table[0] = {
```

```
    ChunkSize64.ckID = {`s`, `x`, `m`, `l`};   // chunk ID for the <sxml> chunk
     ckSizeLow = xxxx     // low-4-byte chunk size
     ckSizeHigh = xxxx    // high-4-byte chunk size
}
```

2.7 عناصر المقطع <sxml>

الحقل	الوصف
ckID	صفيف من أربع سمات {‘s’, ‘x‘, ‘m‘, ‘l‘} يستخدم لتحديد هوية المقطع.
ckSize	مقاس قسم البيانات في المقطع بالبايت. لا يشمل قيمة 8 بايت التي يستخدمها كل من ckID وckSize.
fmtType	قيمة بايتتين (2) تمثل أسلوب ضغط نص XML. وقيمة 0x0001 تعني أن أسلوب الضغط هو gzip (IETF RFC 1952). وتُستخدم قيمة 0x0000 لبيانات XML غير مضغوطة.
subXMLCkTbSizeLow	مقاس 4 بايتات منخفض لجدول SubXMLChunk [] بما في ذلك 4 بايتات لحقل nSubXMLChunks. ويرد التعبير عن مقاس البيانات 64 بتة بقيمة 0xHHHHLLLL إذا كان <subXMLCkTbSizeLow> و<subXMLChTbSizeHigh> هما 0xLLLL و0xHHHH، على التوالي. وتكون كمية 32 بتة المجردة من إشارة بنسق نهاية صغيرة.
subXMLCkTbSizeHigh	مقاس 4 بايتات عال لجدول SubXMLChunk [] بما في ذلك 4 بايتات لحقل nSubXMLChunks.
nSubXMLChunks	عدد القيود الصالحة في صفيف "جدول SubXMLChunk".
SubXMLChunk table	صفيف المقاطع الفرعية ببيانات XML.
nAlignmentPoints	عدد القيود الصالحة في صفيف "جدول AlignmentPoint".
AlignmentPoint table	صفيف نقاط المحاذاة.

ويوصَّف جدول **SubXMLChunk** على النحو التالي.

الحقل	الوصف
subXMLChunkSize	مقاس مقطع البيانات في مقطع SubXMLChunk بالبايتات باستثناء 8 بايتات المستخدمة في subXMLChunkSize وnSamplesSubDataChunk.
nSamplesSubDataChunk	عدد العينات السمعية لكل قناة مرتبطة بمقطع SubXMLChunk.
xmlData	يحتوي هذا الحقل على بيانات XML أو بيانات XML المضغوطة بأسلوب الضغط المشار إليه بالنمط fmtType.

ويوصَّف جدول **AlignmentPoint** على النحو التالي.

الحقل	الوصف
subXMLChunkByteOffsetLow	تَخالف بايتة البداية في مقطع SubXMLChunk بنقطة محاذاة معبر عنها بالبايتات من بداية المقطع <sxml> باستثناء 8 بايتات المستخدمة في ckID وckSize. وهو تَخالف بايتة البداية في مقطع SubXMLChunk بمقاس 4 بايتات منخفضة. ويرد التعبير عن مقاس البيانات 64 بتة بقيمة 0xHHHHLLLL إذا كان <subXMLChunkByteOffsetLow> و<subXMLChunkByteOffsetHigh> هما 0xLLLL و0xHHHH، على التوالي. وتكون كمية 32 بتة المجردة من إشارة بنسق نهاية صغيرة.
subXMLChunkByteOffsetHigh	تَخالف بايتة البداية في مقطع SubXMLChunk بمقاس 4 بايتات عالية وبنقطة محاذاة. وتكون كمية 32 بتة المجردة من إشارة بنسق نهاية صغيرة.
nSamplesAlignPointLow	الختم الزمني لنقطة المحاذاة المعبر عنها في العينات السمعية لكل قناة من بداية المقطع <data>. وهو تعداد عينات الختم الزمني بمقاس 4 بايتات منخفضة. ويرد التعبير عن تعداد مقاس البيانات 64 بتة بقيمة 0xHHHHLLLL إذا كان <nSamplesAlignPointLow> و<nSamplesAlignPointHigh> هما 0xLLLL و0xHHHH، على التوالي. وتكون كمية 32 بتة المجردة من إشارة بنسق نهاية صغيرة.
nSamplesAlignPointHigh	تعداد عينات الختم الزمني بمقاس 4 بايتات عالية. وتكون كمية 32 بتة المجردة من إشارة بنسق نهاية صغيرة.

8 المقطع CHNA

1.8 تعريف

المقطع <chna> هو مقطع موصَّف لكي يستخدم مع نموذج التعريف السمعي (ADM) في التوصية ITU-R BS.2076. ويتكون المقطع <chna> من رأسية يليها عدد من المسارات وعدد من المعرفات الفريدة (UID) للمسارات. يلي ذلك صفيف من هياكل المعرفات التي يتضمن كل منها معرفات مقابلة لمعرفات عناصر النموذج ADM.

يعتمد مقاس المقطع على عدد المعرفات الفريدة (UID) للمسارات المقرر تعريفها. ويجب أن يكون عدد هياكل المعرفات مساوياً لعدد المعرفات الفريدة المستخدمة أو أكبر. ويمكن من خلال السماح لعدد هياكل المعرفات بتجاوز عدد المعرفات الفريدة تسهيل تحديث المقطع وإضافة معرفات جديدة إليه دون الاضطرار إلى تغيير مقاس المقطع. فعلى سبيل المثال، قد لا يكون من الواضح كيف يمكن توليد العديد من المعرفات الفريدة في البداية، ولذلك يضبط عدد هياكل المعرفات في المقطع على 64 (لأن المنفذ يعتبر ذلك أكثر من كاف لهذه المهمة)؛ ثم تولد البرمجيات 55 معرفاً فريداً (عدد نموذجي للمعرفات الفريدة الأولية) تملأ 55 من هياكل المعرفات الأولى، بحيث تضبط قيم هياكل المعرفات التسعة الباقية 9 على الصفر.

يمكن لمعرفات النموذج ADM ضمن المقطع أن تشير إما إلى البيانات الشرحية للنموذج ADM التي تنقل في المقاطع <axml> و<bxml> أو <sxml>، أو في ملف تعريف مشترك خارجي. فإذا كانت الأرقام الأربعة الأخيرة بالنظام الستة عشري للمعرفات بقيمة x0FFF أو أقل فإنها تعرف بوصفها تعاريف مشتركة في التوصية ITU-R BS.2094 – *تعاريف مشتركة لنموذج تعريف الصوت* (مثلاً تعاريف القناة لـ 'FrontLeft' و'FrontRight'). وتعرف أي معرفات لا تقل قيمتها عن 0x1000 بوصفها تعاريف مخصصة، وبالتالي تكون متضمنة في المقاطع <axml> و<bxml> أو <sxml> ضمن الملف.

يتضمن هيكل المعرف audioID مؤشراً للمسار يستخدم في الملف <data> (الذي يتضمن العينات السمعية)، بدءاً من القيمة 1 للمسار الأول. وهو يتضمن معرفاً فريداً للمسار تتضمنه البيانات الشرحية للنموذج ADM. وقد تكون العناصر السمعية لمسار معين مختلفة خلال أحد الملفات، وفي هذه الحالة يكون لكل تعريف معرف فريد مختلف. وبالتالي يمكن أن يكون لكل مسار عدة معرفات فريدة. وتشير القيمتان الأخريان في الهيكل إلى معرفات عناصر النسق audioTrackFormat والنسق audioPackFormat في النموذج ADM. ويسمح النموذج ADM بحذف نسقي audioTrackFormat وaudioStreamFormat إذا كان نمط نسق الجوهر السمعي هو تشكيل PCM الخطي. وعندئذ يشار إلى نسق ويسمح النموذج ADM بحذف نسقي audioTrackFormat وaudioStreamFormat إذا كان نمط نسق الجوهر السمعي هو تشكيل PCM الخطي. بدلاً من النسق audioTrackFormat.

```
struct chna_chunk
{
    CHAR    ckID[4];        // {'c','h','n','a'}
    DWORD   ckSize;         // size of the <chna> chunk
    WORD    numTracks;      // number of tracks used
    WORD    numUIDs;        // number of track UIDs used
    audioID ID[N];          // IDs for each track  (where N >= numUIDs)
};

struct audioID
{
    WORD    trackIndex;     // index of track in file
    CHAR    UID[12];        // audioTrackUID value
    CHAR    trackRef[14];   // audioTrackFormatID or audioChannelFormatID reference
    CHAR    packRef[11];    // audioPackFormatID reference
    CHAR    pad;            // padding byte to ensure even number of bytes
}
```

2.8 عناصر المقطع <chna>

ckID صفيف من أربع سمات {'c','h','n','a'}[1] يستخدم لتحديد هوية المقطع.

ckSize مقاس قسم البيانات في المقطع (لا يتضمن البايتات الثماني المستخدمة في كل من المعرف ckID ومقاس المقطع ckSize).

numTracks عدد المسارات المستخدمة في الملف. وحتى إذا تضمن المسار أكثر من مجموعة واحدة من المعرفات، فإنه يظل يشكل مساراً واحداً.

numUIDs عدد المعرفات الفريدة (UID) المستخدمة في ملف. وبما أنه يمكن إعطاء المسار الواحد عدة معرفات فريدة (تغطي فترات زمنية مختلفة)، فقد يأخذ قيمة أكبر من **numTracks**. ويتعين أن تكون هذه القيمة مطابقة لعدد المعرفات المحددة في معرف الهوية **ID**.

ID هيكل يتضمن مجموعة المعرفات المرجعية السمعية للمسار. ويتضمن هذا الصفيف عدد N من المعرفات، حيث N >= numUID، وعندما يكون numUID أقل من N تضبط محتويات معرفات المسار غير المستعملة على الصفر. وعند قراءة المقطع يمكن اشتقاق قيمة N من ckSize، حيث إن ckSize = 4 + (N * 40)، وبالتالي N = (ckSize – 4) / 40.

trackIndex مؤشر المسار في الملف، ويبدأ بالرقم 1. ويقابل ذلك مباشرة ترتيب المسارات المشذرة في المقطع <data>.

[1] **ملاحظة:** لا يكون التعريف "chna" = DWORD ckID فريداً. فالمعماريات المختلفة تولد ترتيبات مختلفة للسمات. وبالتالي يعرف بدلاً من ذلك char ckID[4] = {'c','h','n','a'}.

UID قيمة audioTrackUID الخاص بالمسار. ويتميز صفيف السمات بالنسق ATU_xxxxxxxx حيث x رقم ستة عشري.

trackRef مرجع معرف المسار audioTrackFormatID. ويتميز صفيف السمات بالنسق AT_xxxxxxxx_xx حيث x رقم ستة عشري. ويُستخدم أيضاً النسق AC_xxxxxxxx_00، (تحشو اللاحقة "00" السلسلة لمطابقة نسق سلسلة audioTrackFormatID وهي لا تحمل أي معنى) حيث تكون x خانة رقمية ست عشرية، عند حذف كل من audioTrackFormat وaudioStreamFormat للجوهر السمعي في تشكيل PCM الخطي والإحالة إلى نسق audioChannelFormat مباشرة في شفرة ADM XML.

packRef مرجع معرف المسار audioTrackFormatID. ويتميز صفيف السمات بالنسق ATU_xxxxxxxx حيث x رقم ستة عشري. وإذا لم يكن audioPackFormatID مطلوباً (أي عندما يشير audioPackFormatID إلى النسق audioPackFormat بدلاً من النسق audioChannelFormat) يتعين أن يملأ هذا الحقل بقيم صفرية.

pad بايت واحد لضمان أن يكون لهيكل audioID عدد وزجي من البايتات.

ويتعين عند عدم استخدام أحد المعرفات **ID** إعطاء المؤشر **trackIndex** القيمة صفر وإعطاء الحقول الأخرى سلاسل صفرية يكون لها نفس طول السلسلة العادية للمعرف المستخدم. لذلك تتكون السلسلة الصفرية للمرجع packRef من 11 سمة صفرية (القيمة صفر بتشفير ASCII) ويتكون trackRef من 14 سمة صفرية.

3.8 أمثلة إعلامية

للمساعدة في توضيح عمل المقطع <chna> فقد أوردنا هنا بعض الأمثلة البسيطة. وتَستخدم شبه الشفرة في كل مثال ترميزاً يشبه السلسلة للمعرفات (مثلاً "AT_00010001_01")، حيث يتعين عملياً استعمال صفيف من السمات لتجتب إدراج سمة إنهاء خاوية في النهاية (وبالتالي يمكن القيام بذلك عملياً بهذه الطريقة: {'A','T','_','0','0','0','0','1','0','0','0','0','1','_','0','1'}).

1.3.8 الملف المجسم البسيط

لا تزال غالبية الملفات السمعية القائمة عبارة عن ملفات مجسمة من قناتين، يتضمن المسار الأول القناة اليسرى ويتضمن المسار الثاني القناة اليمنى. ويتم تعريف القناة اليسرى في النموذج ADM بواسطة المعرف AT_00010001_01، والقناة اليمنى بالمعرف AT_00010002_01. أما تعريف كدسة الستريو فيتم بواسطة المعرف AP_00010002.

وتظهر أدناه شبه الشفرة:

```
ckID = {'c','h','n','a'};
ckSize = 84;
numTracks = 2;
numUIDs = 2;
ID[0]={ trackIndex=1; UID="ATU_00000001"; trackRef="AT_00010001_01"; packRef="AP_00010002"; pad='\0`; };
ID[1]={ trackIndex=2; UID="ATU_00000002"; trackRef="AT_00010002_01"; packRef="AP_00010002"; pad='\0`; };
```

ويكون عدد هياكل المعرف 2، وبالتالي لا يتضمن هذا المثال هياكل ID غير مستخدمة.

وعندما يرد في نموذج ADM حذف كل من audioTrackFomat وaudioStreamFormat وإحالات إلى audioChannelFormat، تُستخدم الشفرة التالية.

```
ckID = {'c','h','n','a'};
ckSize = 84;
numTracks = 2;
numUIDs = 2;
ID[0]={ trackIndex=1; UID="ATU_00000001"; trackRef="AC_00010001_00"; packRef="AP_00010002"; pad='\0`; };
ID[1]={ trackIndex=2; UID="ATU_00000002"; trackRef="AC_00010002_00"; packRef="AP_00010002"; pad='\0`; };
```

2.3.8 مثال بسيط قائم على الأشياء

لا تغطي الأشياء السمعية إلا قسماً فرعياً من الوقت في الملف السمعي. ولتوفير حيز من المكان، قد تتقاسم أشياء غير متراكبة المسار نفسه. ويحدث ذلك عند حدوث عدة معرفات فريدة (UID) في المسار نفسه. ويستخدم هذا المثال أيضاً عدداً من هياكل المعرف ID (32 في هذه الحالة) أكبر من عدد numUID لإظهار كيفية ضبط هياكل ID على الصفر.

```
ckID = {'c','h','n','a'};
ckSize = 1284;
numTracks = 2;
numUIDs = 4;
ID[0]={ trackIndex=1; UID="ATU_00000001"; trackRef="AT_00031001_01"; packRef="AP_00031001"; pad='\0`; };
ID[1]={ trackIndex=1; UID="ATU_00000002"; trackRef="AT_00031003_01"; packRef="AP_00031002"; pad='\0`; };
ID[2]={ trackIndex=1; UID="ATU_00000003"; trackRef="AT_00031004_01"; packRef="AP_00031003"; pad='\0`; };
ID[3]={ trackIndex=2; UID="ATU_00000004"; trackRef="AT_00031002_01"; packRef="AP_00031001"; pad='\0`; };
ID[4]={ trackIndex=0; UID=['\0']*12;      trackRef=['\0']*14;        packRef=['\0']*11;     pad='\0`; };
   :
ID[31]={ trackIndex=0; UID=['\0']*12;      trackRef=['\0']*14;        packRef=['\0']*11;     pad='\0`; };
```

يتضمن المسار الأول 3 معرفات فريدة UID، وبالتالي فهو يتضمن 3 أشياء مختلفة (لها معرفات المسار AT_00031001_01 و AT_00031003_01 و AT_00031004_01) في مواقع زمنية مختلفة ضمن الملف. ويتضمن المسار الثاني معرفاً فريداً واحداً، وبالتالي فهو يتضمن شيئاً واحداً. ويكون لهذا الشيء معرف الكدسة ID ذاته (AP_00031001) الذي للشيء الأول في المسار 1. ويشير ذلك إلى أن الشيء الأول يتضمن قناتين محمولتين في المسارين 1 و2. وقد تستخدم البيانات الشرحية للنموذج ADM المنقولة في المقطع <axml> أو <bxml> أو <sxml> لتوضيح توزيع القنوات والمسارات.

3.3.8 مثال على تعدد المحتوى

قد يتضمن الملف BW64 محتوى متعدداً في ملف واحد، مثل مزيج 5.1 رئيسي على أول 6 مسارات، مع مزيج مجسم بلغة أجنبية على المسارين التاليين. وتتضمن التوصية ITU-R BS.1738 عدة تشكيلات، ويظهر المثال كيف يتمكن سيناريو الإنتاج 5 الوارد في هذه التوصية من التعامل مع المقطع <chna>. ويتضمن هذا السيناريو 8 مسارات، تحتوي المسارات الستة الأولى على مزيج 5.1 كامل، ويحتوي المساران التاليان على مزيج دولي مجسم. ويظهر أدناه المقطع <chna> الناتج:

```
ckID = {'c','h','n','a'};
ckSize = 324;
numTracks = 8;
numUIDs = 8;
ID[0]={ trackIndex=1; UID="ATU_00000001"; trackRef="AT_00010001_01"; packRef="AP_00010003"; pad='\0`; };
ID[1]={ trackIndex=2; UID="ATU_00000002"; trackRef="AT_00010002_01"; packRef="AP_00010003"; pad='\0`; };
ID[2]={ trackIndex=3; UID="ATU_00000003"; trackRef="AT_00010003_01"; packRef="AP_00010003"; pad='\0`; };
ID[3]={ trackIndex=4; UID="ATU_00000004"; trackRef="AT_00010004_01"; packRef="AP_00010003"; pad='\0`; };
ID[4]={ trackIndex=5; UID="ATU_00000005"; trackRef="AT_00010005_01"; packRef="AP_00010003"; pad='\0`; };
ID[5]={ trackIndex=6; UID="ATU_00000006"; trackRef="AT_00010006_01"; packRef="AP_00010003"; pad='\0`; };
ID[6]={ trackIndex=7; UID="ATU_00000007"; trackRef="AT_00010001_01"; packRef="AP_00010002"; pad='\0`; };
ID[7]={ trackIndex=8; UID="ATU_00000008"; trackRef="AT_00010002_01"; packRef="AP_00010002"; pad='\0`; };
```

وتتضمن البيانات الشرحية للنموذج ADM في المقطع <axml> أو <bxml> أو <sxml> معلومات بشأن كيفية فصل هذين المزيجين.

9 قواعد مقاطع XML

هناك ثلاث مقاطع مختلفة يمكنها حمل بيانات XML الشرحية: <axml> و<bxml> و<sxml>. وفي حين أن الغرض الأساسي لهذه المقاطع هو حمل بيانات ADM XML الشرحية (على النحو الموصَّف في التوصية ITU-R BS.2076) أو بيانات S-ADM الشرحية (على النحو الموصَّف في التوصية ITU-R BS.2125)، يمكن لها أيضاً أن تحمل بيانات XML الشرحية الأخرى، مثل البيانات الشرحية الإذاعية الموصوفة في الفقرة 11. وبقدرة أجزاء متعددة على حمل بيانات XML الشرحية، يخاطَر بأن تتعارض البيانات الشرحية في مقطع ما مع البيانات الشرحية في مقطع آخر. لذلك تطبق القواعد التالية:

1 يجب ألا يكون هناك أكثر من مثيل واحد لأي مقطع XML معين.

2 إذا حُملت بيانات ADM الشرحية:

أ) يجب ألا تظهر إلا في المقطع <axml> أو <bxml>، وليس في كليهما؛

ب) يتعين وجود مقطع <chna> يشير مرجعياً إلى بيانات ADM الشرحية.

3 إذا حُملت بيانات S-ADM الشرحية، يجب ألا تظهر إلا في المقطع <sxml>.

4 في حال حمل بيانات ADM الشرحية وبيانات S-ADM الشرحية معاً، يجب أن تستقل عن بعضها البعض (أي لا تتقاطع مرجعياً مع بعضها البعض).

5 إذا حُملت بيانات شرحية أخرى (أي مغايرة لبيانات ADM أو S-ADM):

أ) يمكن حملها إلى جانب البيانات الشرحية لنموذجي ADM وS-ADM في نفس المقطع إذا رُغب في ذلك؛

ب) ينبغي ألا تمثل محتويات "البيانات الشرحية الأخرى" أي شيء يجري وصفه بالفعل في بيانات ADM أو بيانات S-ADM الشرحية القائمة؛

ج) في حال إشارة "البيانات الشرحية الأخرى" مرجعياً إلى بيانات ADM أو بيانات S-ADM الشرحية، يجب أن تكون بيانات ADM أو S-ADM الشرحية المشار إليها موجودة في الملف.

10 التوافق مع التوصية ITU-R BS.1352

بما أن النسق BWF (التوصية ITU-R BS.1352) هو نسق مختصر للملف RIFF/WAVE (كما هو محدد في الملحق 2) يحتوي على مقاطع زائدة، وأبرزها المقطع <bext> و<ubxt>، ثمة حاجة إلى فهم التوافق بين BWF وBW64.

مقاطع BWF التوصية ITU-R BS.1352-4	مقاطع BW64 التوصية ITU-R BS.2088-2	كيفية التعامل
<fmt>	<fmt>	يستخدم بشكل تقليدي
<data>	<data>	يستخدم بشكل تقليدي
<fact>	<fact>	يستخدم بشكل تقليدي
—	<ds64>	انظر الفقرتين 4.2 و4
—	<JUNK>	انظر الفقرتين 4.2 و4
—	<chna>	انظر الفقرة 8 بشأن توزيعات القنوات. ملاحظة: لا تدعم التوصية ITU-R BS.2088-0 الإشارة المرجعية إلى نسق audioChannelFormat.
—	<axml> أو <bxml> أو <sxml>	انظر الفقرات من 5 إلى 7. يستخدم لإذاعة البيانات الشرحية التي قد توجد في المقطعين <bext> أو <ubxt>
<bext> أو <ubxt>	(<bext> أو <ubxt>)	عند قراءة المقطعين <bext> أو <ubxt>، يحول إلى بيانات المقاطع <axml> و<bxml> أو <sxml> المقابلة لنقل النموذج ADM وأي بيانات شرحية XML إذاعية ذات صلة. انظر الفقرة 11 لمزيد من التفاصيل.

11 توليد البيانات الشرحية الإذاعية XML

تحتوي التوصية ITU-R BS.1352 على بيانات شرحية إذاعية في المقطعين <bext> و<ubxt>. ولهذين المقطعين حقول ثابتة الطول ويقتصران على حقول محددة، ما يحول بالتالي من نقل أي بيانات شرحية إذاعية ذات صلة أخرى. وبإمكان المقاطع <axml> و<bxml> و<sxml> في الملف BW64 أن تنقل أي بيانات شرحية XML، وبالتالي يمكن استخدامها في نقل بيانات شرحية إذاعية، بما في ذلك المعلمات الواردة في المقطعين <bext> و<ubxt>.

ومن أجل نقل معلمات <bext> أو <ubxt> في المقاطع <axml> و<bxml> و<sxml> ينبغي استخدام هيكل XML التالي، حيث تشير الملاحظات التي يلها السابقة 'BEXT' (يستخدم "UBXT" عندما يحمل معلمات <ubxt>) إلى معلمات المقطع <bext>.

```
<?xml version="1.0" encoding="UTF-8"?>
<ebuCoreMain xmlns="urn:ebu:metadata-schema:ebuCore_2015"
xmlns:dc="http://purl.org/dc/elements/1.1/"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <coreMetadata>
    <creator>
      <contactDetails>
        <name>
          <!--BEXT: Originator/UBXT: uOriginator-->
        </name>
      </contactDetails>
      <organisationDetails>
        <organisationName>
          <!--BEXT: OriginatorReference/UBXT: uOriginatorReference-->
        </organisationName>
      </organisationDetails>
    </creator>

    <description typeDefinition="bextDescription" or "ubxtDescription">
      <dc:description>
        <!--BEXT: Description/UBXT: uDescription-->
      </dc:description>
    </description>
    <date>
      <!--BEXT: OriginationDate/UBXT: OriginationDate and
          BEXT: OriginationTime/UBXT: OriginationTime below-->
      <created startDate="2000-10-10" startTime="12:00:00"/>
    </date>

    <format>
      <audioFormatExtended>
        <!--BEXT: TimeReference/UBXT: TimeReference below-->
        <audioProgramme audioProgrammeID="..." start="00:00:00:00">
          <!--Other audioProgramme metadata here -->
        </audioProgramme>
        <!--Other ITU-R BS.2076 ADM metadata here -->
      </audioFormatExtended>
      <technicalAttributeString typeDefinition="CodingHistory">
        <!--BEXT: CodingHistory/UBXT: uCodingHistory-->
      </technicalAttributeString>
    </format>

    <identifier formatLabel="UMID"
formatLink="http://www.ebu.ch/metadata/cs/ebu_IdentifierTypeCodeCS.xml#1.1">
      <dc:identifier>
        <!--BEXT: UMID/UBXT: UMID-->
      </dc:identifier>
    </identifier>
  </coreMetadata>
</ebuCoreMain>
```

تقوم XML أعلاه على مخططي البيانات الشرحية EBUCore [2] وAESCore [3]، اللذين يعتبران متوافقين مع التوصية ITU-R BS.2076.

وعند قراءة ملف BWF موضوع التوصية ITU-R BS.1352 بهدف تحويله إلى ملف BW64، ينبغي تحويل المقطعين <bext> أو <ubxt> إلى لغة XML الواردة هنا لإدراجهما ضمن المقاطع <axml> أو <bxml> أو <sxml>.

12 تمديد اسم الملف لملف النسق BW64

يعرّف تمديد الملف الخاص بملفات متوافقة مع النسق BW64 على شكل ".wav". ويمكّن ذلك البرمجيات التقليدية من قراءة المقاطع في الملف التي تفهمها (بالدرجة الأولى <fmt> و<data>)، بحيث يمكن النفاذ إلى العينات السمعية على الأقل.

ومع أنه لا يوصى باستخدام أي تمديدات بديلة لأسماء الملفات لدى توليد ملفات BW64، يمكن التوقع بأن يكون التمديد ".bw64" قد استخدم بصورة غير مناسبة. لذلك ينبغي للبرمجيات التي تقرأ الملف BW64 أن تجيز هذا التمديد البديل لاسم الملف.

13 بيبليوغرافيا

[1] Extensible Markup Language (XML) 1.0 W3C Recommendation 26-November-2008 http://www.w3.org/TR/2008/REC-xml-20081126

[2] EBU Tech 3293, "EBU Core Metadata Set v.1.6".

[3] AES 60-2011, "AES standard for audio metadata – Core audio metadata".

[4] IETF: RFC 1952, "GZIP file format specification version 4.3," Internet Engineering Task Force, Reston, VA, May, 1996. http://tools.ietf.org/html/rfc1952

الملحق 2
(إعلامي)

نسق الملف RIFF WAVE (.WAV)

أخذت المعلومات المدرجة في هذا الملحق من وثائق المواصفات الخاصة بنسق الملف RIFF. وهي واردة للعلم فقط. وقد أدرجت نتيجة لغياب مصدر خارجي موثوق للأغراض المرجعية.

1 نسق الملفات السمعية من النمط الموجي (WAVE)

فيما يلي تعريف النسق WAVE. وينبغي للبرامج أن تنتظر (وتتجاهل) أي مقاطع مجهولة تواجهها كما يحدث مع كافة الأشكال RIFF. ولكن يتعين على المقطع <fmt-ck> أن يحدث دائماً قبل المقطع <wave-data>، ويكون كلا هذين المقطعين إجبارياً في الملف WAVE.

```
<WAVE-form> ->
    RIFF('WAVE'
        <fmt-ck>    // Format chunk
        [<fact-ck>]   // Fact chunk
        [<other-ck>]  // Other optional chunks
        <wave-data>)  // Sound data
```

تتولى الفقرات التالية وصف المقاطع WAVE:

1.1 مقطع النسق WAVE

يحدد مقطع النسق WAVE <fmt-ck> نسق البيانات <wave-data> ويعرَّف المقطع <fmt-ck> كما يلي:

```
<fmt-ck> ->fmt(<common-fields>
                <format-specific-fields>)
<common-fields> ->
    Struct {
        WORD  wFormatTag;           // Format category
        WORD  nChannels;            // Number of channels
        DWORD nSamplesPerSec;           // Sampling rate
        DWORD nAvgBytesPerSec;    // For buffer estimation
        WORD  nBlockAlign;          // Data block size
    }
```

وتكون الحقول التي يحتوي عليها الجزء <common-fields> من المقطع على النحو التالي:

الحقل	الوصف
wFormatTag	رقم يدل على فئة النسق WAVE للملف. ويعتمد على هذه القيمة محتوى الجزء <format-specific-fields> من المقطع <fmt-ck> وتفسير بيانات الموجة.
nchannels	عدد القنوات الممثلة في بيانات الموجة، مثل 1 بالنسبة لغير المجسم أو 2 بالنسبة للمجسّم.
nSamplesPerSec	معدل الاعتيان (عدد العينات في الثانية الواحدة) الذي تتعين إعادة إنتاج كل قناة به.
nAvgBytesPerSec	متوسط عدد البايتات التي يتعين نقل بيانات الموجة بها في الثانية الواحدة. وباستخدام تلك القيمة يمكن لبرمجيات استعادة التسجيل تقدير مقاس الدارئ.
nBlockAlign	تراصف الكتلة (بالبايتات) لبيانات الموجة. ويتعين على برمجيات استعادة التسجيل معالجة بايتات متعددة من <nBlockAlign> من البيانات في المرة الواحدة، بحيث يمكن استخدام القيمة <nBlockAlign> لتراصف الدارئ.

ويتكون الحقل <format-specific-fields> من صفر أو أكثر من بايتات المعلمات. وتتوقف المعلمات المدرجة على فئة النسق WAVE – راجع الفقرات التالية لمزيد من التفاصيل بهذا الخصوص. وتكتب برمجيات استعادة التسجيل بشكل يسمح (ويتجاهل) أي معلمات مجهولة تظهر في آخر الحقل <format-specific-fields>.

2.1 فئات النسق WAVE

تحدد فئة النسق WAVE للملف بواسطة قيمة الحقل <wFormatTag> من المقطع <fmt>. ويعتمد تمثيل البيانات في الحقل <wave-data> ومحتوى الحقل <format-specific-fields> من المقطع <fmt> على فئة النسق.

وفيما يلي الفئات المفتوحة غير المحمية بحقوق الملكية المعرفة حالياً في النسق WAVE:

wFormatTag	القيمة	النسق
WAVE_FORMAT_UNKNOWN	0x0000	غير معروف
WAVE_FORMAT_PCM	0x0001	نسق PCM (تشكيل شفري نبضي)
WAVE_FORMAT_IEEE_FLOAT	0x0003	IEEE float
WAVE_FORMAT_EXTENSIBLE	0xFFFE	نسق موجي موسّع – يحدده SubFormat

ملاحظة – لا يتم استخدام إلا النسقان WAVE_FORMAT_PCM وWAVE_FORMAT_UNKNOWN في الوقت الحالي مع الملفات BW64. وتعطي الفقرة 2 التالية تفاصيل النسق PCM WAVE. أما الفقرة 3 فتعطي معلومات عامة متعلقة بأنساق WAVE الأخرى. وقد يتم تعريف أنساق WAVE أخرى غير تلك في المستقبل.

في الماضي كان النسق WAVE_FORMAT_EXTENSIBLE يستعمل للملفات متعددة القنوات، ولكن ينبغي تجنب ذلك في المستقبل.

3.1 المقطع Fact

يخزّن المقطع <fact-ck> معلومات تتعلق بمحتويات الملفات non-PCM WAVE. ولذلك لا يستعمل هذا المقطع في هذه الصيغة للنسق BW64. ويعرف هذا المقطع على النحو التالي:

```
<fact-ck> ->  fact( <dwSampleLength:DWORD> )
```

ويمثل المقطع <dwSampleLength> طول البيانات في العينات. ويستعمل الحقل <nSamplesPerSec> من رأسية النسق الموجي بالاقتران مع الحقل <dwSampleLength> لتحديد طول البيانات بالثواني.

ويلزم المقطع "fact" لجميع الأنساق non-PCM WAVE. ولا يطلب المقطع لملفات WAVE_FORMAT_PCM المعيارية.

وسيتم تمديد المقطع "fact" ليشمل أي معلومات أخرى تستلزمها الأنساق WAVE في المستقبل. وتظهر الحقول المضافة بعد الحقل <dwFileSize>. ويمكن للتطبيقات أن تستخدم حقل مقاس المقطع لتحديد الحقول الموجودة في الوقت الحاضر.

4.1 المقاطع الاختيارية الأخرى

ويحدَّد عدد من المقاطع الأخرى لاستخدامها في النسق WAVE. وتضم مواصفات النسق WAVE وأي تحديثات تدخل عليها لاحقاً تفاصيل هذه المقاطع.

ملاحظة – يمكن للنسق WAVE أن يدعم المقاطع الاختيارية الأخرى التي قد تتضمنها الملفات WAVE لحمل معلومات محددة. وتعتبر هذه المقاطع مقاطع خاصة وتتجاهلها التطبيقات التي لا تستطيع تفسيرها.

2 نسق التشكيل الشفري النبضي (PCM)

إذا ضبط الحقل <wFormatTag> من المقطع <fmt-ck> على القيمة WAVE_FORMAT_PCM، فإن بيانات الموجات تتكون بالتالي من عينات تمثل في شكل نسق التشكيل الشفري النبضي (PCM). أما بالنسبة إلى بيانات موجات التشكيل الشفري النبضي، فإن الحقل <format-specific-fields> يعرّف على النحو التالي:

```
<PCM-format-specific> ->
    struct {
        WORD  nBitsPerSample;     // Sample size
    }
```

ويحدد الحقل <nBitsPerSample> عدد بتات البيانات المستخدمة في تمثيل كل عينة من كل قناة. وفي حالة وجود قنوات متعددة يكون مقاس العينة هو ذاته لكل قناة.

يتعين على الحقل <nBlockAlign> أن يكون مساوياً للصيغة التالية مقرباً إلى أقرب عدد صحيح تالٍ:

nChannels × BytesPerSample

وتحسب قيمة BytesPerSample بتقريب قيمة nBitsPerSample لأقرب بايتة صحيحة تالية. وعندما تكون كلمة العينة السمعية أقل من عدد صحيح من البايتات، توضع البتات الأكثر دلالة للعينة السمعية في البتات الأكثر دلالة لكلمة البيانات، وتضبط بتات البيانات غير المستخدمة المجاورة للبتة الأقل دلالة على الصفر.

وبالنسبة لبيانات PCM، يجب أن يكون الحقل <nAvgBytesPerSec> من المقطع <fmt> مساوياً للصيغة التالية:

nSamplesPerSec × nBblockAlign

الملاحظة 1 – تسمح المواصفة الأصلية للنسق WAVE، على سبيل المثال لعينات من 20 بتة من قناتين بالترزيم إلى 5 بايتات تتقاسم بايتة واحدة للبتتين الأقل دلالة في القناتين. وتحدد هذه التوصية عدداً صحيحاً من البايتات لكل عينة سمعية للحد من الالتباس في التطبيقات ولتحقيق الحد الأقصى من التوافق في التبادل البيني.

1.2 ترزيم البيانات للملفات PCM WAVE

تخزن العينات في الملف WAVE وحيد القناة على نحو متتال. وتمثل القناة 0 بالنسبة إلى الملفات WAVE المجسمة القناة اليسرى، في حين تمثل القناة 1 القناة اليمنى. وتكون عينات الملفات WAVE متعددة القنوات مشذرة.

وتبين المخططات التالية ترزيم البيانات للملفات WAVE المجسمة وغير المجسمة ذات البتات الثماني (8):

ترزيم البيانات من أجل التشكيل PCM غير المجسم ذي البتات الثماني (8)

العينة 1	العينة 2	العينة 3	العينة 4
القناة 0	القناة 0	القناة 0	القناة 0

ترزيم البيانات من أجل التشكيل PCM غير المجسم ذي البتات الثماني (8)

العينة 1		العينة 2	
القناة 0 (يسرى)	القناة 1 (يمنى)	القناة 0 (يسرى)	القناة 1 (يمنى)

وتبين المخططات التالية ترزيم البيانات للملفات WAVE المجسمة وغير المجسمة ذات البتات الست عشرة (16):

ترزيم البيانات من أجل التشكيل PCM غير المجسم ذي البتات الست عشرة (16)

العينة 1		العينة 2	
القناة 0 بايتة من الرتبة الدنيا	القناة 0 بايتة من الرتبة العليا	القناة 0 بايتة من الرتبة الدنيا	القناة 0 بايتة من الرتبة العليا

ترزيم البيانات من أجل التشكيل PCM المجسم ذي البتات الست عشرة (16)

العينة 1			
القناة 0 (يسرى)	القناة 0 (يسرى)	القناة 1 (يمنى)	القناة 1 (يمنى)
بايتة من الرتبة الدنيا	بايتة من الرتبة العليا	بايتة من الرتبة الدنيا	بايتة من الرتبة العليا

2.2 نسق البيانات في العينات

تدرج كل عينة في عدد صحيح i. ويكون مقاس العينة i هو أقل عدد لازم من البايتات من أجل احتواء مقاس العينة المحدد. وتخزن أقل البايتات دلالة أولاً. وتخزن البتات التي تمثل اتساع العينة في أكثر البتات دلالة في العينة i، وتضبط البتات الباقية على الصفر.

وإذا كان مقاس العينة (المسجل في الحقل <nBitsPerSample>) 12 بتة مثلاً، يتم تخزين كل عينة في عدد صحيح مؤلف من بايتتين. وتضبط أقل أربع بتات دلالة من البايتة الأولى (الأقل دلالة) على الصفر. ويكون نسق البيانات والقيم القصوى والدنيا من عينات الموجات PCM ذات الأحجام المتباينة على النحو التالي:

مقاس العينة	نسق البيانات	القيمة القصوى	القيمة الدنيا
من بتة إلى ثماني بتات	عدد صحيح غير جبري	255 (0xFF)	0
تسع بتات أو أكثر	عدد صحيح جبري i	أكبر قيمة موجبة من i	أكبر قيمة سالبة من i

وتكون القيم القصوى والدنيا والوسطى مثلاً من أجل بيانات الموجات ذات البتات الثماني (8) والست عشرة (16) على النحو التالي:

النسق	القيمة القصوى	القيمة الدنيا	القيمة الوسطى
PCM ذو 8 بتات	255 (0xFF)	0	128 (0x80)
PCM ذو 16 بتة	32767(0x7FFF)	–32768(–0x8000)	0

3.2 أمثلة على الملفات PCM WAVE

فيما يلي **مثال** على الملف PCM WAVE بمعدل اعتيان يبلغ 11,025 kHz، غير مجسم ذي 8 بتات في العينة الواحدة:

```
RIFF('WAVE' fmt(1, 1, 11025, 11025, 1, 8)
          data(<wave-data>) )
```

وفيما يلي **مثال** على الملف PCM WAVE بمعدل اعتيان يبلغ 22,05 kHz، مجسم ذي 8 بتات في العينة الواحدة:

```
RIFF('WAVE' fmt(1, 2, 22050, 44100, 2, 8)
          data(<wave-data>) )
```

4.2 تخزين البيانات من فئة WAVE

يحتوي الحقل <**wave-data**> على بيانات الموجات ويعرف على النحو التالي:

```
<wave-data> ->  { <data-ck> }
    <data-ck> -> data( <wave-data> )
```

5.2 المقطع Fact

يخزن مقطع الحقل <fact-ck> معلومات هامة تتعلق بمحتويات الملف WAVE. ويعرف هذا المقطع على النحو التالي:

```
<fact-ck> ->   fact(<dwFileSize:DWORD>) // Number of samples
```

ولا يكون هذا المقطع مطلوباً بالنسبة إلى الملفات من فئة PCM.

وسيتم تمديد المقطع "fact" ليشمل أي معلومات أخرى تستلزمها الأنساق WAVE في المستقبل. وتظهر الحقول المضافة بعد الحقل <dwFileSize>. وللتطبيقات أن تستخدم حقل مقاس المقطع لتحديد الحقول الموجودة في الوقت الحاضر.

6.2 المقاطع الاختيارية الأخرى

يحدَّد عدد من المقاطع الأخرى لاستخدامها في النسق WAVE. وتضم مواصفة النسق WAVE وأي تحديث يدخل عليها لاحقاً تفاصيل هذه المقاطع.

الملاحظة 1 – يمكن للنسق WAVE أن يدعم المقاطع الاختيارية الأخرى التي قد تتضمنها الملفات WAVE لحمل معلومات محددة. ويتم اعتبار هذه المقاطع في نسق الموجات الإذاعية مقاطع خاصة وتتجاهلها التطبيقات التي لا تستطيع تفسيرها.

3 أنماط أخرى من الملفات WAVE

يستخدم هيكل نسق الموجة الممدد والمضاف إلى المقطع <fmt-ck> من أجل تحديد جميع بيانات الموجات التي لها نسق يختلف عن النسق PCM، وفيما يلي وصف لها. ويستخدم هيكل نسق الموجة الممدد العام من أجل كافة الأنساق غير الأنساق PCM.

```
typedef struct waveformat_extended_tag {
    WORD  wFormatTag;      // format type
    WORD  nChannels;         // number of channels (i.e. mono, stereo...)
    DWORD nSamplesPerSec;  // sample rate
    DWORD nAvgBytesPerSec; // for buffer estimation
    WORD  nBlockAlign;     // block size of data
    WORD  wBitsPerSample;  // number of bits per sample of mono data
    WORD  cbSize;            // the count in bytes of the extra size
} WAVEFORMATEX;
```

الحقل	الوصف
wFormatTag	يعرف نمط الملف WAVE.
nChannels	عدد القنوات في الموجة السمعية، قناة واحدة لغير المجسم وقناتان للمجسم.
nSamplesPerSec	تردد معدل الاعتيان للملف WAVE. ويساوي هذا التردد 48000 أو 44100، إلخ. كما يستخدم هذا المعدل دخل مقاس العينة في المقطع fact لتحديد مدة البيانات.
nAvgBytesPerSec	متوسط معدل البيانات. ويمكن لبرمجيات استعادة التسجيل أن تقدر مقاس الدارئ باستخدام القيمة <nAvgBytesPerSec>.
nBlockAlign	تراصف الفدرة (بالبايتات) للبيانات في المقطع <data-ck>. وتحتاج برمجيات استعادة التسجيل إلى معالجة مضاعفات بايتات <nBlockAlign> من البيانات في المرة الواحدة، مما يجعل قيمة الحقل <nBlockAlign> قابلة للاستخدام من أجل تراصف الدارئ.
wBitsPerSample	عدد البتات في العينة الواحدة وفي القناة الواحدة. ويفترض أن يكون لكل القنوات نفس استبانة العينات. وإذا لم يكن هذا الحقل ضرورياً، يتعين ضبطه على الصفر.
cbSize	مقاس المعلومات الإضافية (بالبايتات) في رأسية النسق WAVE ما عدا مقاس بنية التمديد WAVEFORMATEX.

ملاحظة – تحتوي الحقول التي تأتي بعد الحقل <cbSize> على معلومات محددة يحتاجها النسق WAVE المحدد في الحقل <wFormatTag>.

الملحق 3
(معياري)

تعاريف أنماط البيانات الأولية

الوسوم التالية هي وسوم ذرية، وهي وسوم تشير إلى أنماط بيانات أولية. ونمط البيانات المكافئ C مبين أيضاً.

الوسم	المعنى	النمط C
<CHAR>	عدد صحيح جبري ذو 8 بتات	سمة جبرية
<BYTE>	عدد صحيح غير جبري ذو 8 بتات	سمة غير جبرية
<INT>	عدد صحيح جبري ذو 16 بتة بنسق الترتيب الأقل أهمية	عدد صحيح جبري
<WORD>	عدد صحيح غير جبري ذو 16 بتة بنسق الترتيب الأقل أهمية	عدد صحيح غير جبري
<LONG>	عدد صحيح جبري ذو 32 بتة بنسق الترتيب الأقل أهمية	عدد صحيح طويل جبري
<DWORD>	عدد صحيح غير جبري ذو 32 بتة بنسق الترتيب الأقل أهمية	عدد صحيح طويل غير جبري
<FLOAT>	عدد بفاصلة عائمة ذو 32 بتة وفقاً للمعيار IEEE	عائم
<DOUBLE>	عدد بفاصلة عائمة ذو 64 بتة وفقاً للمعيار IEEE	مزدوج
<STR>	سلسلة (تسلسل من السمات)	
<ZSTR>	سلسلة تنتهي بصفر	
<BSTR>	سلسلة ذات سابقة بمقاس كلمة (8 بتات)	
<WSTR>	سلسلة ذات سابقة بمقاس كلمة (16 بتة)	
<BZSTR>	سلسلة تنتهي بصفر ذات سابقة بمقاس بايت	

الملحق 4
(إعلامي)

تغييرات للمواصفات الواردة في الملحق 1

1 التغييرات من التوصية ITU-R BS.2088-0 إلى التوصية ITU-R BS.2088-1

تقدم المراجعة 1 لهذه التوصية التغييرات التالية للمواصفات الواردة في الملحق 1:

- إضافة المقطع BXML في الفقرة 6.
- إضافة المقطع SXML في الفقرة 7.
- إضافة وظيفة جديدة لحذف النسقين audioTrackFormat وaudioStreamFormat في الفقرة 8.

2 التغييرات من التوصية ITU-R BS.2088-1 إلى التوصية ITU-R BS.2088-2

يقدم التعديل الثاني لهذه التوصية تفاصيل إضافية حول معالجة المقاطع المستخدمة في أنساق ملفات wave الأخرى الواردة في الملحق 1:

- توضيح كيفية معالجة المقاطع الواردة في الفقرات 1.2 و2.2 و10.
- إضافة طريقة لإنشاء ملف XML من المقطع <ubxt> الوارد في الفقرة 11.

_____