



---

## Changes made in ETSI Drafting Rules (EDRs)

---

### Paragraphs (clause 2.12.2)

#### The change

Latest version of ETSI Drafting rules strengthened the rule on clauses and subclauses in ETSI deliverables such that clauses or subclauses that have text below the title cannot contain further subdivisions.

#### Why the change

In the previous version of the ETSI Drafting Rules (May 2014) text paragraphs followed by clause subdivision were called "Hanging Paragraphs". Whilst they were discouraged, they were not expressly disallowed.

In the new version of the [ETSI Drafting Rules](#) updates were made to disallow the use of "Hanging paragraphs" and thus ensure precise referencing of specific text in ETSI deliverables. The rules now state that, when a clause or subclause has text below the title, there can be no further subdivisions.

Note that this rule is aligned with internationally recognized practice, for example [ISO/IEC Directives Part 2](#) : <http://isotc.iso.org/livelink/livelink?func=ll&objId=4230456&objAction=browse&sort=subtype>.

Note also that the current EDRs no longer use the term "hanging paragraphs". The reason is that when you look it up on Google it refers to indented paragraphs, which is not what EDRs are talking about.

---

### Application clarification

When editing a first draft (not a revision) it is easier to handle the changes as the document itself is not yet referenced in another document. For this reason, the clauses can be easily updated as well as the cross referencing within the document

It is slightly more difficult to make changes in clause numbering in a revision of an existing deliverable. To avoid renumbering of the entire document, we recommend that a new subclause is added as shown in the example on page 3. It will then be necessary to check the cross referencing. If clause <X> is referenced in the document, ETSI Officer or Technical Body Rapporteur need to check (or inform [editHelp!](mailto:edithelp@etsi.org) at [edithelp@etsi.org](mailto:edithelp@etsi.org)) whether it is the entire clause <X> or clause <X>.0 that is being referenced.

On page 2 is the revised text in the current ETSI Drafting Rules (December 2014).





## Example

This is an example of the incorrect subdivision of a clause followed by the same text correctly subdivided.

### 4 → Encoding¶

The present document provides a method to encode raw SPI XML data, generated as per the hybrid radio SPI XML specification ETSI TS 102 818 [1], into a compact binary format to be broadcast using the MOT protocol [2] using the DAB [3] or DRM [4] systems.

The hybrid SPI XML contains elements and attributes that are not relevant for broadcast use and consequently these are not binary encoded. Similarly, it also represents the data in a slightly modified form to the broadcast only version [1.2] and in order to generate backwards compatible binary that permits devices equipped with an EPG decoder based on the broadcast only version of this specification [1.1] to continue to decode correctly, some additional steps are required to generate the binary.

In the present document the term "delivery system" is used to indicate whether the SPI service is delivered over DAB or DRM. This parameter is used in the encoding of several elements and attributes, but is not itself encoded. Therefore, decoders need to know which system delivered the binary data in order to correctly decode it.

The binary encoding described here uses a tag-length-value encoding. Each element or attribute is encoded using a unique tag value, a length value (indicating the length of the data contained within this element or attribute) and the actual data value(s). This enables devices to easily skip elements that are not wanted or were undefined.

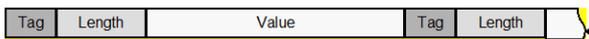


Figure 1: Tag-length-value encoding scheme¶

XML elements are all encoded in these binary structures as described in clause 4.2. Attributes are coded in a similar way (see clause 4.4). The hierarchical nature of the SPI XML is generally preserved in these binary structures, but the structure is not necessarily identical. Various common data types have been assigned efficient binary encodings as described in clause 4.7. For an example of a binary encoded XML object, see annex C.

Note that although the length of certain data types can be worked out from their encoding, there shall still be a length field in the attribute encoding (see clause 4.4).

### 4.1 → Syntax specification¶

The specifications of syntax that appear in the present document are written using a form of pseudo-code that is similar to the procedural language "C"; this provides for easy specification of loops and conditional data structures. Within these specifications, the type of individual data fields is expressed using the mnemonics given in table 1.

### 4 → Encoding¶

#### 4.0 → Encoding method¶

The present document provides a method to encode raw SPI XML data, generated as per the hybrid radio SPI XML specification ETSI TS 102 818 [1], into a compact binary format to be broadcast using the MOT protocol [2] using the DAB [3] or DRM [4] systems.

The hybrid SPI XML contains elements and attributes that are not relevant for broadcast use and consequently these are not binary encoded. Similarly, it also represents the data in a slightly modified form to the broadcast only version [1.2] and in order to generate backwards compatible binary that permits devices equipped with an EPG decoder based on the broadcast only version of this specification [1.1] to continue to decode correctly, some additional steps are required to generate the binary.

In the present document the term "delivery system" is used to indicate whether the SPI service is delivered over DAB or DRM. This parameter is used in the encoding of several elements and attributes, but is not itself encoded. Therefore, decoders need to know which system delivered the binary data in order to correctly decode it.

The binary encoding described here uses a tag-length-value encoding. Each element or attribute is encoded using a unique tag value, a length value (indicating the length of the data contained within this element or attribute) and the actual data value(s). This enables devices to easily skip elements that are not wanted or were undefined.



Figure 1: Tag-length-value encoding scheme¶

XML elements are all encoded in these binary structures as described in clause 4.2. Attributes are coded in a similar way (see clause 4.4). The hierarchical nature of the SPI XML is generally preserved in these binary structures, but the structure is not necessarily identical. Various common data types have been assigned efficient binary encodings as described in clause 4.7. For an example of a binary encoded XML object, see annex C.

Note that although the length of certain data types can be worked out from their encoding, there shall still be a length field in the attribute encoding (see clause 4.4).

#### 4.1 → Syntax specification¶

The specifications of syntax that appear in the present document are written using a form of pseudo-code that is similar to the procedural language "C"; this provides for easy specification of loops and conditional data structures. Within these specifications, the type of individual data fields is expressed using the mnemonics given in table 1.



Checking the ***editHelp!*** website regularly  
will ensure you are up-to-date with ETSI skeletons, guides, tools, toolbar, etc.

The ***editHelp!*** website has been created to facilitate access to ETSI tools as well as getting useful information and to make the production of standards a simple task.

In case of any difficulties, please do not hesitate to contact ***editHelp!*** at [editHelp@etsi.org](mailto:editHelp@etsi.org), we are here to help you produce your standard.

Working closer together can ensure that your future ETSI deliverable is published without delay.