Stage 3 proposal #18: Make @audience, @platform, @product, @otherprops into specializations

DITA 2.0 should refactor the existing profiling attributes – @audience, @platform, @product, and @otherprops – so that they're defined in domains and specialized from @props, as we did for @deliveryTarget in 1.3.

Champion

Championed by Robert D. Anderson, originally proposed by Chris Nitchie

Tracking information

Event	Date	Links
Stage 1 proposal accepted	2 May 2017	https://lists.oasis-open.org/archives/dita/201705/msg00015.html
Stage 2 proposal submitted	28 Feb 2018	HTML, DITA
Stage 2 proposal discussed	20 March 2018	https://lists.oasis-open.org/archives/dita/201803/msg00061.html
Stage 2 proposal approved	27 March 2018	https://www.oasis-open.org/apps/org/workgroup/dita/download.php/62798/minutes20180327.txt?referring_url=%2Fkws
Stage 3 proposal submitted to reviewers	25 Sept 2018	Deb Bissantz, Dawn Stevens
Stage 3 proposal (this document) submitted to TC	1 Oct 2018	DITA source in SVN

Approved technical requirements

1. Create four new base domain modules, one for each of @audience, @platform, @product, and @otherprops.
2. Remove the existing definitions of these attributes. In OASIS grammars, these are defined as base attributes in the %filter-atts; group along with @props and a placeholder for extensions of @props. This is the only place the attributes are named in OASIS grammars (DTD: commonElements.mod; RNG: commonElementsMod.rng).
3. Add the domain attributes into the shells of all OASIS-provided grammar files, and add them to the props-extension group in each configured shell. This will restore the attributes to any element that previously defined them.

Dependencies or interrelated proposals

N/A

Modified grammar files

commonElements.mod (before)

```
<!ENTITY % filter-atts
              "props
                        CDATA

 #IMPLIED
              platform
                        CDATA

 #IMPLIED
              product
                        CDATA

 #IMPLIED
              audience
                        CDATA

 #IMPLIED
              otherprops
                        CDATA

 #IMPLIED
              %props-attribute-
extensions;"
>
```

commonElements.mod (after)

```
<!ENTITY % filter-atts
              "props
                        CDATA

 #IMPLIED
              ~~platform~~
~~                        CDATA~~

~~ #IMPLIED~~
~~              product~~
~~                        CDATA~~

~~ #IMPLIED~~
~~              audience~~
~~                        CDATA~~

~~ #IMPLIED~~
~~              otherprops~~
~~                        CDATA~~

~~ #IMPLIED~~
              %props-attribute-
extensions;"
>
```

All provided DTD shells (before)

```
<!ENTITY % deliveryTargetAtt-d-dec
  PUBLIC "-//OASIS//ENTITIES DITA
 2.0 Delivery Target Attribute
 Domain//EN"

 "deliveryTargetAttDomain.ent"
>%deliveryTargetAtt-d-dec;
```

All provided DTD shells (after)

```
<!ENTITY % audienceAtt-d-dec
  PUBLIC "-//OASIS//ENTITIES DITA
 2.0 Audience Attribute Domain//EN"
         "audienceAttDomain.ent"
>%audienceAtt-d-dec;
<!ENTITY % deliveryTargetAtt-d-dec
  PUBLIC "-//OASIS//ENTITIES DITA
 2.0 Delivery Target Attribute
 Domain//EN"

 "deliveryTargetAttDomain.ent"
>%deliveryTargetAtt-d-dec;
<!ENTITY % platformAtt-d-dec
  PUBLIC "-//OASIS//ENTITIES DITA
 2.0 Platform Attribute Domain//EN"
         "platformAttDomain.ent"
>%platformAtt-d-dec;
<!ENTITY % productAtt-d-dec
  PUBLIC "-//OASIS//ENTITIES DITA
 2.0 Product Attribute Domain//EN"
         "productAttDomain.ent"
>%productAtt-d-dec;
<!ENTITY % otherpropsAtt-d-dec
  PUBLIC "-//OASIS//ENTITIES DITA
 2.0 Otherprops Attribute Domain//
EN"
         "otherpropsAttDomain.ent"
```

All provided DTD shells (before)

```
<!ENTITY % props-attribute-
extensions
  "%deliveryTargetAtt-d-attribute;"
>
```

All provided DTD shells (after)

```
>%productAtt-d-dec;
```

```
<!ENTITY % props-attribute-
extensions
  "%audienceAtt-d-attribute;
  %deliveryTargetAtt-d-attribute;
  %platformAtt-d-attribute;
  %productAtt-d-attribute;
  %otherpropsAtt-d-attribute;"
>
```

commonElements.rng (before)

```
<define name="filter-atts">
      <optional>
        <attribute name="props"/>
      </optional>
      <optional>
        <attribute name="platform"/>
      </optional>
      <optional>
        <attribute name="product"/>
      </optional>
      <optional>
        <attribute name="audience"/>
      </optional>
      <optional>
        <attribute
 name="otherprops"/>
      </optional>
      <ref name="props-attribute-
extensions"/>
    </define>
```

commonElements.rng (after)

```
<define name="filter-atts">
      <optional>
        <attribute name="props"/>
      </optional>
      ~~<optional>~~
~~        <attribute name="platform"/>~~
~~      </optional>~~
~~      <optional>~~
~~        <attribute name="product"/>~~
~~      </optional>~~
~~      <optional>~~
~~        <attribute name="audience"/>~~
~~      </optional>~~
~~      <optional>~~
~~        <attribute~~
~~ name="otherprops"/>~~
~~      </optional>~~
      <ref name="props-attribute-
extensions"/>
    </define>
```

All provided RNG shells (before)

In the default domain declaration section:

```
<define name="domains-att">
         <optional>
            <attribute
 name="domains"

 a:defaultValue="
                        ...content
 domains...
                        a(props
 deliveryTarget)"/>
         </optional>
      </define>
```

All provided RNG shells (after)

In the default domain declaration section:

```
<define name="domains-att">
         <optional>
            <attribute
 name="domains"

 a:defaultValue="
                        ...content
 domains...
                        a(props
 audience)
                        a(props
 deliveryTarget)
                        a(props
 platform)
                        a(props
 product)
                        a(props
 otherprops)"/>
         </optional>
```

All provided RNG shells (before)

In the module inclusion section:

```
<include
 href="deliveryTargetAttDomain.rng"
 dita:since="1.3"/>
```

All provided RNG shells (after)

```
       </define>
```

In the module inclusion section:

```
<include
 href="audienceAttDomain.rng"
 dita:since="2.0"/>
<include
 href="deliveryTargetAttDomain.rng"
 dita:since="1.3"/>
<include
 href="platformAttDomain.rng"
 dita:since="2.0"/>
<include href="productAttDomain.rng"
 dita:since="2.0"/>
<include
 href="otherpropsAttDomain.rng"
 dita:since="2.0"/>
```

Rather than make this proposal appear much longer than it should due to long headers + a few lines of meaning, I've uploaded the 4 new DTD domains and 4 new RNG domains to github; they can be reviewed at:

- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/dtd/base/dtd/audienceAttDomain.ent
- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/dtd/base/dtd/platformAttDomain.ent
- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/dtd/base/dtd/productAttDomain.ent
- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/dtd/base/dtd/otherpropsAttDomain.ent
- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/rng/base/rng/audienceAttDomain.rng
- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/rng/base/rng/platformAttDomain.rng
- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/rng/base/rng/productAttDomain.rng
- https://github.com/robander/dita/blob/issue18/propertyatts/doctypes/rng/base/rng/otherpropsAttDomain.rng

Modified terminology

N/A

Modified specification documentation

The following topic needs to be updated: http://docs.oasis-open.org/dita/dita/v1.3/errata02/os/complete/part1-base/langRef/attributes/metadataAttributes.html#select-atts

Before	After
@platform Indicates operating system and hardware. If no value is specified, but the attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor.	**Specializations of @props** The attributes @audience, @deliveryTarget, @platform, @product, and @otherprops are specialized from the @props attribute. They are defined in independent attribute extension domains, and integrated by default into all OASIS-provided document-type shells. If any of these domains is not integrated into a given document-type shell, the relevant attribute(s) will not be available.
@product Contains the name of the product to which the element applies. If no value is specified, but the	**@audience** Indicates the intended audience for the element. If no value is specified, but

Before	After
attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor. **`@audience`** Indicates the intended audience for the element. If no value is specified, but the attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor. **`@otherprops`** This attribute can be used for any other properties that might be needed to describe an audience, or to provide selection criteria for the element. Alternatively, the `@props` attribute can be specialized to provide a new metadata attribute instead of using the general `@otherprops` attribute. If no value is specified, but the attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor. The `@audience`, `@platform`, `@product`, and `@otherprops` attributes are property attributes which support conditional processing for filtering or flagging. Each takes a space-delimited set of values, with optional groups of space-delimited properties. Although these attributes are not specialized and not specializeable, the group syntax matches that for generalized attributes in *Attribute generalization*. **`@deliveryTarget`** The intended delivery target of the content, for example "html", "pdf", or "epub".	the attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor. **`@deliveryTarget`** Indicates the intended delivery target for the element; for example, "html", "pdf", or "epub". If no value is specified, but the attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor. **`@platform`** Indicates operating system and hardware. If no value is specified, but the attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor. **`@product`** Indicates the name of the product to which the element applies. If no value is specified, but the attribute is specified on an ancestor within a map or within the related-links section, the value will cascade from the closest ancestor. **`@otherprops`** This attribute can be used for any other properties that might be needed to describe an audience, or to provide selection criteria for the element. Alternatively, the `@props` attribute can be specialized to provide a new metadata attribute instead of using the general `@otherprops` attribute. If no value is specified, but the attribute is specified on an ancestor within a map or within the

Before	After
	related-links section, the value will cascade from the closest ancestor.

In the topic on cascading map attributes: http://docs.oasis-open.org/dita/dita/v1.3/errata02/os/complete/part1-base/archSpec/base/cascading-of-attributes-from-map-to-map.html#cascading-of-attributes-from-map-to-map

Before	After
The following attributes cascade from map to map: • `@audience`, `@platform`, `@product`, `@otherprops`, `@rev` • `@props` and any attribute specialized from `@props` • ...	The following attributes cascade from map to map: • **`@rev`** • **`@props` and any attribute specialized from `@props` (including those integrated by default in OASIS shells: `@audience`, `@deliveryTarget`, `@platform`, `@product`, `@otherprops`)** • ...

From the topic on conditional processing, the `@product` attribute is missing from the current list: http://docs.oasis-open.org/dita/dita/v1.3/errata02/os/complete/part1-base/archSpec/base/condproc.html#condproc

Before	After
DITA defines attributes that can be used to enable filtering and flagging individual elements. The `@audience`, `@deliveryTarget`, `@otherprops`, `@platform`, and `@props` attributes (along with specializations of `@props`) allow conditions to be assigned to elements so that the elements can be included, excluded, or flagged during processing.	DITA defines attributes that can be used to enable filtering and flagging individual elements. The `@props` attribute and any attribute specialized from `@props` (including those integrated by default in OASIS shells: `@audience`, `@deliveryTarget`, `@platform`, `@product`, `@otherprops`) allow conditions to be assigned to elements so that the elements can be included, excluded, or flagged during processing.

Update to the list of common map attributes in http://docs.oasis-open.org/dita/dita/v1.3/errata02/os/complete/part1-base/archSpec/base/ditamap-attributes.html#ditamap-attributes

Before	After
The following metadata and reuse attributes are used by both DITA maps and DITA topics: • `@product`, `@platform`, `@audience`, `@otherprops`, `@rev`, `@status`, `@importance` • *...other atts...* • `@props` and any attribute specialized from `@props` (such as `@deliveryTarget`)	The following metadata and reuse attributes are used by both DITA maps and DITA topics: • `@rev`, `@status`, `@importance` • *...other atts...* • `@props` and any attribute specialized from `@props` (including those integrated by default in OASIS shells: `@audience`, `@deliveryTarget`, `@platform`, `@product`, `@otherprops`)

Remove a note in attribute generalization http://docs.oasis-open.org/dita/dita/v1.3/errata02/os/complete/part1-base/archSpec/base/generalization-attributes.html#attributegeneralize

Before	After
Note: The `@audience`, `@platform`, `@product`, and `@otherprops` attributes allow grouped values that	*...removed...*

Before	After
reuse the generalized syntax described here; however, these attributes are not specialized or specializeable. For these attributes, it can be typical to author or edit the grouped values directly.	

Updates needed in the topic on property attribute groups: http://docs.oasis-open.org/dita/dita/v1.3/errata02/os/complete/part1-base/archSpec/base/usage-of-conditional-processing-attributes.html#usage-of-conditional-processing-attributes

Before	After
Groups organize classification metadata into subcategories. This is intended to support situations where a predefined metadata attribute applies to multiple specialized subcategories. For example, the `@product` attribute can be used to classify an element based on both related databases and related application servers. Using groups for these subcategories allows each category to be processed independently; when filter conditions exclude all applicable databases within a group, the element can be safely excluded, regardless of any other `@product` conditions. Groups can be used within `@audience`, `@product`, `@platform`, or `@otherprops`. The following rules apply:	Groups can be used to organize classification metadata into subcategories. This is intended to support situations where a predefined metadata attribute applies to multiple specialized subcategories. The grouping syntax exactly matches the syntax used for generalized attributes, making it valid inside `@props` and any attribute specialized from `@props` (including those integrated by default in OASIS shells: `@audience`, `@deliveryTarget`, `@platform`, `@product`, `@otherprops`). For example, the `@product` attribute can be used to classify an element based on both related databases and related application servers. Using groups for these subcategories allows each category to be processed independently; when filter conditions exclude all applicable databases within a group, the element can be safely excluded, regardless of any other `@product` conditions. Groups can be used within `@props` and any attribute specialized from `@props` (including those integrated by default in OASIS shells: `@audience`, `@deliveryTarget`, `@platform`, `@product`, `@otherprops`). The following rules apply:
Note: While the grouped values reuse the generalized attribute syntax found in *Attribute generalization*, the `@audience`, `@product`, `@platform`, and `@otherprops` attributes cannot be specialized or generalized.	*...removed...*

In the definition of the DITAVAL `<prop>` element http://docs.oasis-open.org/dita/dita/v1.3/errata02/os/complete/part1-base/langRef/ditaval/ditaval-prop.html#ditaval-prop:

Before	After
The `<prop>` element identifies an attribute, and usually values in the attribute, to take an action on. The attribute either must be a conditional-processing attribute (`@platform`, `@product`, `@audience`, `@deliveryTarget`, `@props`, and `@otherprops`) or a specialization of the `@props` attribute.	The `<prop>` element identifies an attribute, and usually values in the attribute, to take an action on. The identified attribute is a conditional-processing attribute (either `@props` or a specialization of `@props`, such as `@audience`, `@deliveryTarget`, `@platform`, `@product`, or `@otherprops`).

Before		After	
`@att`	The attribute to be acted upon. If using a literal attribute name, it must be one of `@props`, `@audience`, `@platform`, `@product`, `@otherprops`, `@deliveryTarget`, or a specialization of `@props`. Otherwise, the value should be the name of a group used within the `@audience`, `@platform`, `@product`, or `@otherprops` attributes. If the `@att` attribute is absent, then the `<prop>` element declares a default behavior for any conditional processing attribute.	**`@att`**	The attribute to be acted upon. If using a literal attribute name, it is `@props` or a specialization of `@props` (such as `@audience`, `@deliveryTarget`, `@platform`, `@product`, or `@otherprops`). Otherwise, the value is the name of a group within one of those attributes, with the group name specified using the generalized attribute syntax. If the `@att` attribute is absent, then the `<prop>` element declares a default behavior for any conditional processing attribute.

Migration plans for backwards incompatibilities

- No documents will need to be migrated.
- Processors may optionally remove exceptions in any filter/flagging code that explicitly look for these attributes. As long as the processors correctly support `@props` and specializations, they will continue to work as designed even without migration.
- Document types that wish to retain all four attributes will need to add the four new domains into shells. When moving shells into a DITA 2.0 environment, documents may be updated manually using the tables above; the same changes made for OASIS provided DTD / RNG shells will work for local shells that are already using DITA 1.3 and use the `@deliveryTarget` module. (For a large number of document-type shells, a search/replace operation or script may be faster.) Shells that do not include `@deliveryTarget` will likely need to be updated manually (search and replace could also work well for those, but no other strings can be assumed to exist at the right spot in all shells).
- Constraints that *remove* any of these attributes today will need to be updated; it may be possible to discard the constraint module and simply not include the relevant domain(s). This will need to be handled manually, because of the many ways it could potentially be done today, but searching over any modules for the four attributes should make them easy to find.
- Constraints that explicitly declare or restrict values on these attributes will need to be updated. This will need to be handled manually, because of the many ways it could potentially be done today, but searching over any modules for the four attributes should make them easy to find.