DITA 2.0 proposed feature #257: Add Hardware domain

Provide a new domain to support documenting content associated with documenting hardware. (Formerly “Provide a new element to semantically tag things you press on keyboards or other input devices.”)

Date and version information

> **Comment by Zoe**
> Draft 3

Include the following information:

Date that this feature proposal was completed
March 15, 2020

Champion of the proposal
Zoë Lawson

Links to any previous versions of the proposal
Original email: https://lists.oasis-open.org/archives/dita/201906/msg00033.html

Links to minutes where this proposal was discussed at stage 1 and moved to stage 2
https://lists.oasis-open.org/archives/dita/201906/msg00068.html

Reviewers for Stage 2 proposal

- Bill Burns

  > **Comment by Zoe**
  > Do I need a 3rd reviewer?

- Kris Eberlein
- Kieth Schengilli-Roberts

Links to e-mail discussion that resulted in new versions of the proposal
https://www.oasis-open.org/apps/org/workgroup/dita/download.php/66383/minutes20191203.txt

Link to the GitHub issue
https://github.com/oasis-tcs/dita/issues/257

Original requirement or use case

Writers often want to add special formatting for things a user presses or types, such as `CTRL+Z` or `ALT+0235` or “Press the `Start` button”. The element `<userinput>` exists, but that generally seems to be intended for strings a user enters into fields, not button or key names. Many style guides recommend different formatting for things a user presses on a keyboard vs text a user needs to enter.

Currently, you can work around this limitation using specializations or `@outputclass` such as `<userinput outputclass="keyboard">CTRL+S</userinput>` in processing. It would be convenient to trim this down to something like `<kbd>CTRL+S</kbd>`.

Potentially this would be a small addition to the software elements domain. There is also an argument that this could be a new domain for hardware elements.

After further discussion with the Technical Council, it was decided to instead create a new domain for elements associated with documenting hardware.

Use cases

There is a need for a semantic tag to indicate "things you press" on a keyboard or other physical devices.

The original intention is for keys on a keyboard such as "ENTER", "PRINT SCREEN", or "CTRL+Z". These are different than the intention of a shortcut. There may not be a software UI equivalent, such as using CTRL+C to stop a process in a command window.

This could also be used in the hardware world for any physical button, switch, knob, or other control. For example, "Amt Tend" on a register, "Start" on a copier, or any "On/Off" switch.

The types of physical buttons, switches, knobs, sliders, and other physical controls are many and varied. Instead of trying to make numerous elements for all the various options, this domain will include elements that intended for specialization.

New terminology

The `<hwcontrol>` element would inherit from topic/ph and be defined as follows:

The `<hwcontrol>` element represents the name of a key, button, switch, or other physical control on a device. This element is part of the DITA hardware domain, a special set of DITA elements designed to document hardware tasks, concepts, and reference information.

> **Comment by Zoe**
> Do we want the @type? Part of me says yes, but how do we come up with a list? My intention is that it's here for people to specialize. Can we leave this @type blank so people can constrain it however they'd like? (e.g. outputclass?)

The `<hwcontrol>` element contains a `@type` attribute to identify the type of hardware control.

The `<partno>` element would inherit from topic/data and be defined as follows:

The `<partno>` element represents the part number for the parent `<hwcontrol>` object. This element is part of the DITA hardware domain, a special set of DITA elements designed to document hardware tasks, concepts, and reference information.

> **Comment by Zoe**
> Keith Schengili-Roberts proposed the following two elements in addition via email.

The `<continuous-control>` element would inherit from topic/ph and be defined as follows:

The `<continuous-control>` element represents hardware controls that do not have discrete states, such as sliders, knobs, or pedals. This element is part of the DITA hardware domain, a special set of DITA elements designed to document hardware tasks, concepts, and reference information.

The `<discrete-control>` element would inherit from topic/ph and be defined as follows:

The `<discrete-control>` element represents hardware controls that have discrete states such as a button or a switch. This element is part of the DITA hardware domain, a special set of DITA elements designed to document hardware tasks, concepts, and reference information.

Proposed solution

Create a new hardware domain (hw-d) to include elements germain to documenting hardware tasks, concepts, and reference information.

Add the following new elements:

- `<hwcontrol>`
- `<partno>`

Benefits

Address the following questions:

Who will benefit from this feature?

Writers can semantically mark the difference between text users should enter and buttons users need to press without configuring outputclass or otherwise extending DITA. These writers will generally be either software or hardware writers.

What is the expected benefit?

The ability to distinguish between special formatting for buttons to press without the overhead of working with outputclass or extending DITA.

How many people probably will make use of this feature?

Many - Software writers often need to refer to keys in ways that are not shortcuts. Many people want to have special formatting for keyboard key names that is different from text a user needs to type into a text field, for example. Hardware authors often have special button names they need to use.

How much of a positive impact is expected for the users who will make use of the feature?

Minor - It's relatively easy to work around this using outputclass, but it would be very nice to have.

Additional positive impacts could include:

- generating lists of buttons
- assisting terminology management for translation
- special processing or formatting

Technical requirements

Provide a detailed description of how the solution will work. Be sure to include the following details:

Adding new elements or attributes

Adding a topic or map specialization

N/A

Adding a domain

N/A

Adding an element

<hwdomain>

- Attributes: Universal attribute group, outputclass, `@keyref` (Same as `<uicontrol>`) and the `@type` attribute.
- The `<hwcontrol>` element represents the name of a key, button, switch, or other physical control on a device. This element is part of the DITA hardware domain, a special set of DITA elements designed to document hardware tasks, concepts, and reference information.
- Use the same content model as `<uicontrol>` with the addition of `<partno>`

> **Comment by Zoe**
> I'm not sure this is the correct list of content. We want `<image>` and `<data>` definitely

- Yes, this is translatable. It is a phrase element.
- Inheritance would be + topic/ph hw-d/hwcontrol

<partno>

- Attributes: Data element attributes group , Link relationship attribute group , Universal attribute group , and outputclass. (Same as `<bookpartno>`)

- The <partno> element represents the part number for the parent <hwcontrol> object. This element is part of the DITA hardware domain, a special set of DITA elements designed to document hardware tasks, concepts, and reference information.
- Use the same content model as <bookpartno> for contains.

  Comment by Zoe
  I'm not sure this is the correct list of content. Shouldn't this just be text? maybe ph so you can conref something in? Maybe data?

  The <partno> can be contained by the same as <uicontrol>, with the addition of <hwcontrol>.
- Yes, this is translatable. The part number itself may be different per country.
- Inheritance would be + topic/data hw-d/partno

DTDs:

Comment by Zoe
These are a guess. I know that large portions of these are wrong because we haven't finalized the content models yet.

hwDomain.ent

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- ============================================================= -->
<!--                    HEADER                                     -->
<!-- ============================================================= -->
<!--  MODULE:    DITA Hardware Domain                              -->
<!--  VERSION:   2.0                                               -->
<!--  DATE:      March 2020                                        -->
<!--                                                               -->
<!-- ============================================================= -->
<!-- ============================================================= -->
<!--                    PUBLIC DOCUMENT TYPE DEFINITION            -->
<!--                    TYPICAL INVOCATION                         -->
<!--                                                               -->
<!--  Refer to this file by the following public identifier or an  -->
<!--       appropriate system identifier                           -->
<!-- PUBLIC "-//OASIS//ELEMENTS DITA Hardware Domain//EN"          -->
<!--       Delivered as file "hwDomain.mod"                        -->
<!-- ============================================================= -->
<!-- SYSTEM:     Darwin Information Typing Architecture (DITA)     -->
<!--                                                               -->
<!-- PURPOSE:    Declaring the elements and specialization         -->
<!--             attributes for the Hardware Domain                -->
<!--                                                               -->
<!-- ORIGINAL CREATION DATE:                                       -->
<!--             March 2020                                        -->
<!--                                                               -->
<!--             (C) Copyright OASIS Open 2020.                    -->
<!--             All Rights Reserved.                              -->
<!--                                                               -->
<!--  UPDATES:                                                     -->
<!-- ============================================================= -->

<!-- ============================================================= -->
<!--                    ELEMENT EXTENSION ENTITY DECLARATIONS      -->
<!-- ============================================================= -->

<!ENTITY % hw-d-ph
  "hwcontrol"
>
<!ENTITY % hw-d-data
  "partno"
>
<!-- ============================================================= -->
<!--                    DOMAIN ENTITY DECLARATION                  -->
```

```
<!-- ============================================================= -->

<!ENTITY hw-d-att
  "(topic hw-d)"
>

<!-- ================ End DITA Hardware Domain ================== -->
```

hwDomain.mod

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- ============================================================= -->
<!--                    HEADER                                     -->
<!-- ============================================================= -->
<!--  MODULE:    DITA Hardware Domain                              -->
<!--  VERSION:   2.0                                               -->
<!--  DATE:      March 2020                                        -->
<!--                                                               -->
<!-- ============================================================= -->
<!-- ============================================================= -->
<!--                    PUBLIC DOCUMENT TYPE DEFINITION            -->
<!--                    TYPICAL INVOCATION                         -->
<!--                                                               -->
<!--  Refer to this file by the following public identifier or an  -->
<!--       appropriate system identifier                           -->
<!-- PUBLIC "-//OASIS//ELEMENTS DITA Hardware Domain//EN"          -->
<!--       Delivered as file "hwDomain.mod"                        -->
<!-- ============================================================= -->
<!-- SYSTEM:     Darwin Information Typing Architecture (DITA)     -->
<!--                                                               -->
<!-- PURPOSE:    Declaring the elements and specialization         -->
<!--             attributes for the Hardware Domain                -->
<!--                                                               -->
<!-- ORIGINAL CREATION DATE:                                       -->
<!--             March 2020                                        -->
<!--                                                               -->
<!--             (C) Copyright OASIS Open 2020.                    -->
<!--             All Rights Reserved.                              -->
<!--                                                               -->
<!--  UPDATES:                                                     -->
<!-- ============================================================= -->

<!-- ============================================================= -->
<!--                   ELEMENT NAME ENTITIES                       -->
<!-- ============================================================= -->

<!ENTITY % hwcontrol   "hwcontrol"                                   >
<!ENTITY % partno    "partno"                                        >

<!-- ============================================================= -->
<!--                    ELEMENT DECLARATIONS                       -->
<!-- ============================================================= -->

<!--                    LONG NAME: Hardware Control          -->
<!ENTITY % hwcontrol.content
                       "(%words.cnt; |
                         %image; |
                         %shortcut;)*"
>
<!ENTITY % hwcontrol.attributes
              "keyref
                          CDATA
                                    #IMPLIED
               %univ-atts;"
>
<!ELEMENT  hwcontrol %hwcontrol.content;>
<!ATTLIST  hwcontrol %hwcontrol.attributes;>

<!--                    LONG NAME: Part Number          -->
<!ENTITY % partno.content
                       "(%words.cnt; |
                         %image; |
                         %shortcut;)*"
```

```
>
<!ENTITY % partno.attributes
              "keyref
                          CDATA
                                    #IMPLIED
               %univ-atts;"
>
<!ELEMENT  partno %partno.content;>
<!ATTLIST  partno %partno.attributes;>

<!-- ============================================================= -->
<!--             SPECIALIZATION ATTRIBUTE DECLARATIONS             -->
<!-- ============================================================= -->

<!ATTLIST  hwcontrol    class CDATA "+ topic/ph hw-d/hwcontrol ">
<!ATTLIST  data         class CDATA "+ topic/data hw-d/partno ">

<!-- ================== End of DITA Hardware Domain ==================== -->
```

</codeblock>

RNG:

> **Comment by Zoe**
> TBD

Adding an attribute

N/A

Renaming or refactoring elements and attributes

N/A

Removing elements or attributes

N/A

Processing impact

Expected to be minimal.

Overall usability

The new `<hwcontrol>` element provides a richer semantic option for writers and makes it easier for people setting up transforms to identify a different formatting for a different concept. DITA users may choose to replace existing semantic tags with this new element.

Backwards compatibility

DITA 2.0 is the first DITA release that is open to changes affecting backwards compatibility. To help highlight any impact, does this proposal involve any of the following?

Was this change previously announced in an earlier version of DITA?

No

Removing a document type that was shipped in DITA 1.3?

No

Removing a domain that was shipped in DITA 1.3?

No

Removing a domain from a document type shell was shipped in DITA 1.3?

No

Removing or renaming an element that was shipped in DITA 1.3?

No

Removing or renaming an attribute that was shipped in DITA 1.3?

No

Changing the meaning of an element or attribute in a way that would disallow existing usage?

No. However, some users may choose to replace existing semantic tagging with this new element. It would not "disallow", but could make current semantic tagging not as valid as it was.

Changing a content model by removing something that was previously allowed, or by requiring something that was not?

No

Changing specialization ancestry?

No

Removing or replacing a processing feature that was defined in DITA 1.3?

No

Are element or attribute groups being renamed or shuffled?

No

Migration plan

If the answer to any question in the previous section is "yes":

Might any existing documents need to be migrated?

If you are already using some sort of outputclass, you may want to replace your existing extension to use this new element. Users will need to search and replace as we would have no idea how different people implemented the extension.

Might any existing processors or implementations need to change their expectations?

People may want to provide some default formatting for `<hwcontrol>`. It could be based on `<userinput>` or `<cmdname>`.

Might any existing specialization or constraint modules need to be migrated?

If there are constraints on the contexts in which `<userinput>` is permitted, there may be modifications needed for the new element.

If no migration need is anticipated, why not?

N/A

Costs

Outline the impact (time and effort) of the feature on the following groups.

Maintainers of the grammar files

- Minor cost if this is just a new element added to the software domain.
- Moderate cost if this creates a new hardware domain.

Editors of the DITA specification

- How many new topics will be required? 3 - one describing the hardware domain, and one topic for the new elements `<hwcontrol>` and `<partno>`.
- How many existing topics will need to be edited? All topics that list or describe domains, any topics describing the content models, the @type topic.
- Will the feature require substantial changes to the information architecture of the DITA specification? If so, what? - No
- If there is new terminology, is it likely to conflict with any usage of those terms in the existing specification? - N/A

Vendors of tools

This addition should require minimal effort to support.

DITA community-at-large

- Will this feature add to the perception that DITA is becoming too complex? - no
- Will it be simple for end users to understand? - yes.
- If the feature breaks backwards compatibility, how many documents are likely to be affected, and what is the cost of migration? - Migration will be optional, and it should be a relatively simple search and replace for old extension to new extension.

Producing migration instructions or tools

- How extensive will migration instructions be, if it is integrated into an overall 1.3 # 2.0 migration publication or white paper? At most a topic.
- Will this require an independent white paper or other publication to provide migration details? - No
- Do migration tools need to be created before this change can be made? If so, how complex will those tools be to create and to use? - No

Examples

Comment by Zoe
I will need help coming up with a part number example. Not entirely sure how you would use it.

```
<step><cmd>If the command is already running,
select the command window and press <<hwcontrol>>CTRL+C</<hwcontrol>>
to end processing.</cmd></step>
```

```
<step><cmd>After entering the amount you received, press <<hwcontrol>>Amt
 Tend</<hwcontrol>>.</cmd>
<stepresult>This opens the cash drawer. The display shows the amount of change to give the
 customer.</stepresult></step>
```