TTML Profiles

G. Adams

TPAC2013

TTML1 Profiles (1)

- Expression of authorial intent about what a processor must support in order to process a document.
- If processor doesn’t satisfy profile requirements, then must abort processing unless overridden.
- May be interpreted as a definition of a **processor profile**.

TTML1 Profiles (2)

- A profile definition consists of an ordered set of profile specifications.
- A profile specification is either a *feature* specification or an *extension* specification.
- Each feature and extension is referred to using a *designation*.
- Each feature and extension is assigned a *value* (explicitly or by default): used, required, optional.

TTML1 Profiles (3)

- Feature designations and their associated semantics are defined only by the TTWG.
- Extension designations and their associated semantics are defined by either TTWG or 3rd parties.

TTML1 Profiles (4)

- Profile definition may be inline (embedded within) or external to document.
- When external, referred to as Profile Definition Document.
- Multiple profile definitions may be combined to form effective profile.
- Single UNION (most inclusive) profile combination method.

TTML1 Profiles (5)

- Two mechanisms for referring to multiple profiles:
  - Employ **use** attribute on **ttp:profile** element to include specifications from a *baseline* profile.
  - Use of multiple **ttp:profile** elements within document.
- Both mechanism may be used together.

TTML1 Profiles (6)

- When combining specifications from a referenced *baseline* profile and an *inline* profile definition, a REPLACE combination method is used.
- The REPLACE combination method says that a following specification replaces a prior specification when both specifications refer to the same feature or extension.

What’s Missing?

- How to associated a profile with a profile designator?
- How to define requirements or options on content (documents) as opposed to processors? i.e., how to define a **content** profile?
- How does instance document refer to a content profile?
- What processing requirements (mandatory or optional) are implied by content profiles?
- How to combine profiles using methods other than UNION?
- How to combine profile specifications using methods other than REPLACE?
- How to relate new features or extensions with existing features or extensions, particularly as a restriction (subset)?

Profile Designator Proposal

- Add @designator on ttp:profile:
  - Value is xsd:anyURI, where URI serves as designator for said profile.

Content Profile Proposal (1)

- Add @type on ttp:profile:
  - Value set: {*processor*, *content*}
  - Default value *processor*.
- Modify @value on ttp:{feature,extension}:
  - Add *prohibited* value.
  - Deprecate *use* value (not used in practice).
  - Refine semantics of *required* and *optional* according to whether used in processor or content profile.

Content Profile Proposal (2)

- Add @ttp:contentProfile on tt:root, to refer to single content profile.
- If author desires to declare adherence to multiple content profiles, then use multiple ttp:profile elements with @type *content*.
- Add #contentProfile feature.

Content Profile Proposal (3)

- Add @ttp:validation on tt:root:
  - Value set: {required, optional}
  - Default value *optional*.
  - If *required* and validation not supported by processor, then must abort unless overridden.
- Add @ttp:validationAction on tt:root:
  - Value set: {abort, warn, ignore}
  - Default value *abort*.
  - If *abort* and validation fails, then must abort unless overridden.
  - If *warn* and validation fails, then warn unless overridden.
- Add #validation feature.

Profile Combination Proposal

- Add @ttp:profileCombination to tt:root:
  - Value set: {replace, intersection, union}
  - Default value *union*.

Profile Specification Combination Proposal

- Add @combine to ttp:profile:
  - Value set: {replace, intersection, union}
  - Default value *replace*.

Feature Relation Proposal

- Add @restricts to ttp:{feature, extension}:
  - Value is xsd:anyURI, where URI refers to a defined feature or extension designation.
  - A ttp:feature may only refer to a feature designation.
  - A ttp:extension may refer to a feature designation or an extension designation.
- Could also define @extends using same format if use case is identified.