TC39 TG1 E4X Meeting Notes 2003-06-20

Date	June 20, 2003 10:00 AM – 12:00 PM
Location	Conference call 866-500-6738 access code 795181
Convener	Rok Yu (Microsoft)
Editor	John Schneider (BEA/AgileDelta)
Participants	Jeff Dyer (Macromedia) John Schneider (BEA/AgileDelta) Michael Shenfield (RIM) Rok Yu (Microsoft) Waldemar Horwat (Netscape)

Agenda

Agenda was adopted as written.

Additional Comments on 06/06/03 Specification Draft

John asked for any additional comments the group might have on the 06/06/03 draft of the specification. Not everyone had a chance to review it. John asked that group members review the remaining sections of the specification so comments can be discussed at the next face-to-face meeting.

Additional Comments on 2003-06-06 Meeting Notes

John notes that meeting notes were incomplete and that we had discussed the section on Equality. Rok missed transcribing the last part of notes. He will send out another draft of meeting notes and issues spreadsheet.

Namespace Design

John walked the working group through the approaches taken on namespaces in the XML community.

- DOM – physical model of XML document
- XML Query – logical model of XML document

The XML Infoset is more a description of the types of data that might pulled out of an XML document and doesn't really focus on operators on the data.

DOM

DOM level 2 has each element and attribute carrying a prefix, URI, and local name. All three pieces of data move around when nodes are moved around and the operations do not attempt to maintain an association between the nodes and XML declarations.

DOM level 3 continues with this model but adds convenience methods to look up prefixes, lookup namespace URIs and normalize the namespace declarations in a document.

How are qualified names represented?
Qualified names contain a prefix, an URI, and a local name.

How are namespace declarations treated?

Declarations are modeled in the DOM as attributes.

How are namespace declarations associated with the qualified names used in elements and attributes?
There are no direct associations. Association occurs as a side effect of the qualified name for an element or attribute being in scope for a given namespace declaration. DOM trees which are inconsistent with respect to namespaces are possible.

How are default namespaces handled?
A qualified name with no prefix but a URI is considered in the default namespace. The actual default namespace declaration is modeled as an attribute.

XQuery

The XQuery takes an abstract logical view of the data model. It does not require implementations to preserve prefixes, the locations of namespace declarations or the parentage of namespace declarations.

Each element is associated with a set of the inscope namespaces. An namespace record contains a prefix and a URI.

The set of inscope namespaces move with the element and operations define how they are combined.

If it is necessary to determine the set of namespace declarations associated with a given element (e.g., for serialization), this may be computed by an algorithm which takes the set difference of the inscope namespaces on the element with those on the parent element.

The namespace prefix of a namespace qualified attribute which have a null parent pointer (and thus isn't associated with an element) may be computed arbitrarily (e.g., when serialized).

Clients of XQuery data model such as the XQuery language and XSLT may enforce additional constraints on the data. As an example, XQuery doesn't have the concept of updating.

Waldemar asks how XQuery handles XML Namespace 1.1 feature of undeclaring namespaces.

John is not sure and takes an action item (57) to look at XML Namespaces 1.1 specification to determine how undeclaring of namespaces works.

The question is raised as to what exactly is a Namespace object and whether it stores a prefix. This is an open design issue recorded in (58) Does a namespace object store only a URI, or a URI and a prefix?

How are qualified names represented?
Qualified names (QNAMES) are represented as expanded QNames containing a URI, and a local name.

How are namespace declarations treated?
Declarations are not explicitly represented in the data model. Required declarations are tracked by a set of inscope namespaces which is associated with each element. The set of required namespace declarations can be calculated by taking the set difference between the inscope namespaces of an element and it's parent.

How are namespace declarations associated with the qualified names use in elements and attributes?

There are no direct associations as declarations don’t exist. Each expanded QName contains a URI associated with one or more in-scope namespaces of the element. If the URI matches more than one in-scope namespace, one of the matching namespaces is selected arbitrarily.

How are default namespaces handled?
Default namespaces may be declared using XML syntax in a construct clause. They may also be declared directly in the prolog of the XQuery expression. The prefix for default namespaces is empty. The set of namespaces matched by the URI of an expanded QName may include the default namespace. The in-scope namespaces for any given element may contain at most one default namespace.

Discuss Items on Issues List

(50) Is namespace prefix significant in XQuery
The namespace prefix is not significant in the qualified names of the XQuery model. They are stored in the namespace declarations found in the inscope namespaces associated with each element and may be used in seralization.

(51) When should prefixes be relevant?
We discussed this and agreed to close it base on our tentative agreement to follow the XQuery model. I.e., prefixes should not be significant in qualified names or name lookup, but should be stored with the namespace bindings in the in-scope namespaces of an element.

We also took an initial poll of the group to see which namespace model they preferred. Everyone stated a preference for the XQuery model over the DOM model.