greenCDA – Synopsis of a White Paper

Introduction

The full specification of CDA XML addresses universal requirements for exchange and management of structured clinical documents. The greenCDA concept explores one method of working with an implementation-specific XML while maintaining the full utility of CDA, asserting as a primary principle that any simplification must also deliver valid, normative CDA (plain vanilla CDA).

We present this to initiate discussion and further development. We call it "green CDA" because it's good for the environment!

What does it mean to be green? And who is the target audience?

Changing a premise changes the shape of the best-fit solution: good discussion requires explicit premises. To begin with, we assert that a greenCDA schema will:

1. be easier to work with for instance generation
2. have a shorter learning curve and yet can deliver valid CDA.

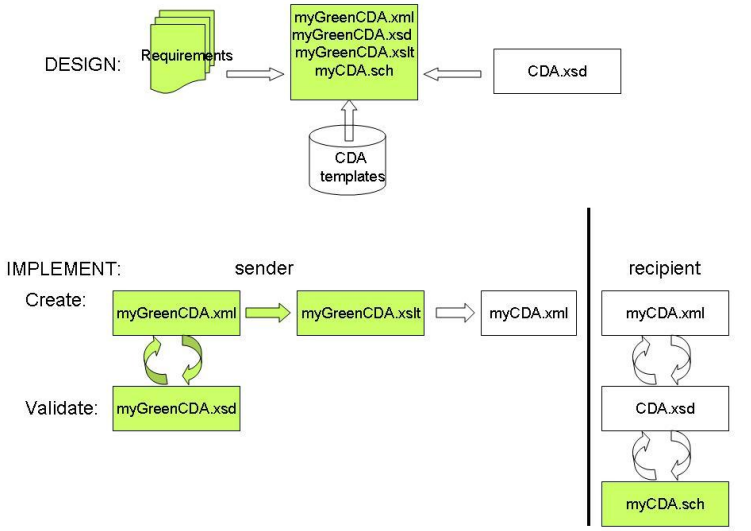
Please note that this effort will be evaluated in the first instance by people who know CDA and CCD very well indeed: to that first audience, the approach may seem unusual at first blush.

Following these objectives, greenCDA must be:

1. Rigorous. Automated conversion to normative CDA is a core capability.
2. Modular. The re-useable unit in a CDA IG is a template for a clinical statement.
3. 80% solution. To have all CDA features, use CDA! A greenCDA is an 80% solution.
4. Clinical/localization. Element and attribute names reflect content semantics not the abstract RIM.
5. Documented. The schema must be self-documenting.
6. Gentle learning curve.
7. Simple for content creators and must not push additional complexity on content recipients.

Process flow

The greenCDA instance is validated locally against the greenCDA schema, and transformed locally to normative CDA (myCDA.xml, in the diagram below) for transmission. The recipient's handling of CDA is unaffected. As is the case today, Implementation Guides will continue to generate IG-specific (green) rules for Schematron validation (myCDA.sch). Or, the recipient in a pre-negotiated exchange might choose to accept the sender's greenCDA.



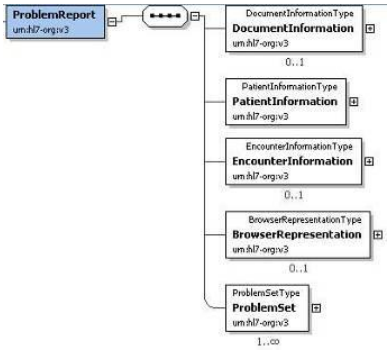
Overview of Method for designing greenCDA schema modules: CCD Problem Act

Start with a clinical statement and mark up the implementation guide. We used "CCD Problem Act".

1. Identify what a transform can generate automatically – fixed characteristics of the template, and structures that in the local environment are never populated. What remains is data that the greenCDA model needs to support.
2. Use best practices to model the information.
    a. Name elements for their meaning (semantics) not for their derivation from the RIM.
    b. Identify singleton elements that have no element children, and make them attributes.
3. Develop a transform to reverse the simplification – convert to full CDA.

With those principles & methodology, the CCD Problem Act and its schema become more compact. The corresponding full CDA Problem Act takes two pages.

```
<ProblemSet>
  <TimeRange start="20091124" end="20091126"/>
  <Problem status="55561003"
           healthStatus="271593001"
           age="27" ageUnit="y">
    <TimeRange start="20091125" end="20091126"/>
    <Topic codeSystem="2.16.840.1.113883.6.96"
           code="233604007"/>
  </Problem>
</ProblemSet>
```



Next steps: Refine & adapt premises; Explore constraints.