NIEM tools

Jan 22 2020

# functions

## mapping

* going from local data requirements to niem content
  + subset
  + extension
* right now we tell people to use a spreadsheet

## search

* additional search targets
  + NIEM release repository
  + EIEM
  + local folder
  + git repository
  + IEPD

## artifact production

## validation

* sample instance
* iepd conformance

## iepd repository

* search / fetch
* publish / push

## user mgt

* publication
* collaboration
* credentials
* privileges
* groups
* session management
  + pause & resume
  + delayed login
    - let people get started before they have to create an account
    - roll over in-progress content to the new account

## help & guidance

* online / tool documentation
* human intervention & support
* tutorial mode

# separate user interfaces

* map & model grid interface
* data model search interface
* conformant instance / message validation against an iepd
* iepd files / folder interface
* iepd metadata / manifest entry forms
* niem version migration
* start page / landing page

# assumptions

* who is creating IEPDs?
  + What skill set is required?
* funding
  + level of effort
  + minimum ROM
    - LOE
      * 1 developer full time
* timeline
  + 5 years
  + 3 phase
* goal
* automated workflow
* flexible vs wizard-driven workflow
  + prefer flexible
* COAs
  + “maintain existing & develop new”
    - follow current release schedule
    - existing tool set is static

# constraints

# development

* agile development
* regular deployment
* customer participation

# architecture

* open source
  + License?
    - BSD-style
    - GPL-style
* hosting
  + publicly hosted
  + privately hosted
  + desktop hosted
    - “user-installable”
  + storage
* web interface
* service interface

# goals for this meeting

* requirements
* roadmap
  + phases

# iepd development process

* phases
  + scenario planning
    - identify sources of requirements
  + analyze requirements
    - enumerate data requirements
    - sources
      * manually entered
      * spreadsheet import
      * database schema import
    - other requirements
      * end up as other artifacts & documentation
    - populate mapping table
  + map & model
    - search
      * smart search
    - select content from
      * NIEM release
      * other IEPDs (via registry)
      * enterprise models
        + how?
    - new content
      * via extension XSDs
      * via mapping table
  + build & validate
    - schema production
    - sample instance validation
    - partial conformance testing
      * of schemas
      * of instances
  + assemble & document
    - files/folder interface
    - complete IEPD
      * including
        + manifest
        + schemas
        + sample data
        + documentation
        + other artifacts

eg

UML

* + - conformance testing
    - implementation support artifacts
  + publish
    - via
      * registry
      * github
      * download zip file
  + implement

# domain model management

* build a domain data model, rather than just an IEPD
* publish
  + data model
  + domain update

# separate APIs

* component search API
  + release repository
  + enterprise model / eiem
  + other iepd repository
* component browse API
* iepd registry
* iepd search API

# modeling support

* distinguishing between
  + new type
  + role
  + augmentation
  + association
  + abstract substitution
  + representation
  + container element

# additional capability

* code list support
  + code list specification
* metadata validation
* id/ref/uri validation

# Current tools

* Current tool gaps
  + mapping data requirements to NIEM subset + extension
  + defining & generating extension schemas
  + producing JSON Schema
  + IEPD assembly
  + validation of instances against an IEPD
* List
  + SSGT
    - search
    - subset building
    - browse
  + Contesa
    - parse IEPD
    - validate IEPD vs specs
  + Migration tool
    - update wantlist to new version
  + Code list tool
    - produce schemas from spreadsheets
  + Wayfarer
    - search
    - browse
  + Movement
    - search
  + model viewer website
    - browse

# MVPs

* capabilities
  + Landing page
  + Files interface for an IEPD
    - empty IEPD
    - download zip
    - upload zip
    - upload file
    - Message Description Manifest creation & editing
      * user details
      * iepd details
      * content details
        + schema assembly
        + message formats
    - Generate model from XML Schemas
    - Generate model from spreadsheet
    - Generate JSON schema from model
    - Generate XSD from model
    - Generate UML from model
    - validation
      * error handling
      * report generation
      * conformance assertion
    - populate with a good template
  + NIEM 5.0 targets
    - production of NIEM 5 XSDs
      * model-to-NIEM 5 XSDs
    - SSGT
      * add “model” schema generator
    - NIEM release
  + Generate JSON Schema from a schema set / zip
    - as a UI / standalone webapp
    - as a web service
    - process:
      * IEPD zip to XML Schema document set
      * XML Schema set to model
      * model to JSON Schema
    - cons
      * Requires JSON people to build XML Schemas
  + Generate JSON Schema from a spreadsheet
    - requires precise spreadsheet format
* list
  + 1st MVP
    - features
      * landing page
      * session mgt
      * file interface
      * spreadsheet-to-model
      * model-to-JSON Schema
    - capabilities
      * generate JSON Schema from a spreadsheet
      * XML-free JSON capability
  + Additional capabilities queue
    - model to XML Schema
    - XML Schema to model
    - service API
    - IEPD assembly
      * upload artifacts
      * result in IEPD package
        + following conventions
    - IEPD validation
      * validate XML Schemas vs. NDR
      * validate XML instance vs. IEPD
      * validate full IEPD vs. IEPD spec
        + check catalog for validity
        + check for mandatory artifacts
    - Map & Model interface
      * search for content
      * add found content to a model

# Model “building blocks”

* IEPD zip to XML Schema document set
* XML Schema set to model
* model to XML Schema document set
  + stabilize order of things in schemas
* model to JSON Schema
  + JSON Schema has to generate & accommodate a specific JSON-LD context
    - so you could build in aliases
* Zip + artifacts to Zip
* spreadsheet to model
* formats
  + Zip file
  + XML Schema document set
  + JSON Schema + JSON-LD context
  + OpenAPI
  + Spreadsheet
  + UML XMI
  + Clicky diagrams
  + Documentation
    - … Word doc

# Registry/Repository

* Public
* MilOps/DOD restricted
* NTAC perspective
  + would like to turn FHIR elements into NIEM definitions
  + openAPI vs. JSON Schema
    - Jim thinks openAPI is our best target
    - likely that json schema won’t get much use
    - service description part of an openapi is outside the “schema” part