OCIL Proposals

OCIL Proposals

- Attempts to use OCIL led to these proposals.
- Some of the proposals affect other specifications.
- Most of the proposals have had reference implementations built and tested.
- We assume familiarity with OCIL and with the proposals released on the discussion list.
- After Dev Days revisions to the affected specs will be updated to reflect the consensus.
- NIST expected to include updates in next versions of specs.
- Next versions of specs expected to be included in next version of SCAP.

OCIL Proposals

- Enterprise Usability Proposals
- Capability Proposals

Enterprise Usability Proposals

- Questionnaire routing (XCCDF)
- Applicability (CPE-Language, XCCDF)
- Timestamp granularity (OCIL)
- Inclusion of AI (OCIL)
- Response provider on question result (OCIL)
- Multiple results (OCIL)

Questionnaire Routing

- Assume benchmark rules will not all be answered by the same person.
- Assume OCIL questioning will usually be managed centrally. Vendors responsible for method to assign questions to respondents.
- Allow author to recommend the type of person who should provide data for each rule.
- Use existing metadata tag in XCCDF – best practice instead of specification requirement.
- Optional feature, processing not required.

Routing Example

```
<Rule id="xccdf_com.example_rule_Rule1" selected="false">
        <title>Unused DBMS components should not be
    installed.</title>
        <description>Are there any unused DBMS components
    installed?</description>
        <metadata>
            <eocil:target_respondent_role
    href="http://iase.disa.mil/roles">DBA</eocil:target_respondent_
    role>
        </metadata>
        <check system="http://scap.nist.gov/schema/ocil/2">
            <check-content-ref href="example-ocil.xml"
    name="ocil:com.example:questionnaire:1"/>
        </check>
</Rule>
```

Questionnaire Routing

- Does the use case need to be explicitly included in SCAP? Does the concept of an OCIL questionnaire manager need to be explicitly defined?
- Should the field be explicitly defined or specified as a best practice use of metadata?
- Tag target respondent role at what level? XCCDF (Benchmark, Group, Rule, Check) or OCIL (Questionnaire, Question)
  - Current proposal supports XCCDF (BGR) using SCAP 1.2 metdata field.
  - Other options require additional changes to XCCDF and OCIL. Metadata might be useful at the XCCDF/Check and OCIL/(item) level anyway.
  - Is it a requirement to route questions within a questionnaire to different people (e.g., dependent questions)?

Applicability

- XCCDF issue for any rule using any check system; potential use elsewhere
- Desire machine-readable method to express applicability beyond existing platform tag
- No changes to CPE
- Addition of DataFactRefType to include externally defined attributes in a language expression
- Many of these facts will not be discoverable but tools may make applicability decisions (unknown implies include)
- Particularly useful for declaring the type of subject when OCIL is the check system
- Expand FactRefType to include other standard identifiers

Applicability Example

```
<platform-specification>
        <platform id="windows_7_mac-1_public">
                <title xml:lang="en-US">Windows 7 (MAC-1_Public)</title>
                <logical-test operator="AND" negate="false">
                        <fact-ref name="cpe:2.3:o:microsoft:windows_7:*:*:*:*:*:*:*:*:*"
system="http://www.mitre.org/CPE"/>
                        <data-fact-ref description="MAC-1" >
                                <asset_property xmlns="http://mil.disa.asset_properties/1.0">
                                        <mac_level>1</mac_level>
                                </asset_property>
                        </data-fact-ref>
                        <data-fact-ref description="Public" >
                                <asset_property xmlns="http://mil.disa.asset_properties/1.0">
                                        <confidentiality_level>Public</confidentiality_level>
                                </asset_property>
                        </data-fact-ref>
                </logical-test>
        </platform>
</platform-specification>
```

Applicability Examples

```
<platform-specification>
      <platform id=“network">
          <title xml:lang="en-US“>Network</title>
          <logical-test operator="AND" negate="false">
                <data-fact-ref description=“network" >
                    <asset_property xmlns="http://mil.disa.asset_properties/1.0">
                          <asset_type>network</asset_type>
                    </asset_property>
                </data-fact-ref>
          </logical-test>
       </platform>
      </platform-specification>
```

```
<platform-specification>
                <platform id=“CVE-CCE_combo">
                          <title xml:lang="en-US“>CVE and CCE combo</title>
                          <logical-test operator="AND" negate="false">
                                    <fact-ref name=“CVE-2012-1234” system="http://cve.mitre.org"/>
                                    <fact-ref name=“CCE-4321-6” system="http://cce.mitre.org"/>
                          </logical-test>
                </platform>
</platform-specification>
```

Applicability

- Should the CPE Language spec be transitioned to the SCAP Language spec?
- Does XCCDF results need to be updated to include applicability results?
  - Initially require inclusion of DataFactRef items in XCCDF/TestResult/target-facts/fact element?
  - XCCDF previously declined to include platform results; new functionality makes this feature more significant
- Should an initial asset properties schema be required by SCAP?

Timestamp Granularity

- OCIL benchmarks currently include only a start and end time for the questionnaire
  - Artifact the one exception
- Multiple respondents => multiple times
  - May want question times to determine validity of answers
- Add timestamp attribute to QuestionnaireResultType, TestActionResultType, QuestionResultType
  - Do we actually need it on anything other than the question (others can be derived)?
  - Not required but should be used if times vary by more than an hour.

<choice_question_result response="ANSWERED"
question_ref="ocil:ns:question:1234" timestamp="2012-06-01T16:45:10">

- Value could be passed to XCCDF as well (//rule-result[@time])

Inclusion of AI

- System and User types not well prescribed
- Add AI consistent with OVAL approach
  - Add “any” with lax processing and a recommendation to use AI
- Clarify that tools should not assume that the asset running the content is the subject of the questions
- Should “system” be renamed “subject”?

Response Provider on Question Result

- OCIL assumes all questions answered by the same person
  - ...but artifacts include a submitter element
- Add a submitter attribute to each question

  <choice_question_result response="ANSWERED"
  question_ref="ocil:ns:question:1234"
  submitter="ocil:com.example:user:1">
  - Should it be done at the questionnaire level instead?
  - Allow a default submitter for the entire questionnaire?
  - Attribute of type "ProviderValuePattern"
    - Reference the id of the user
    - Add id patterns to UserType and SystemTargetType to indicate the id for each item

    <user id="ocil:com.example:user:1">

Multiple Result Sets

- Spec allows; schema does not (unless results are identical for all systems)
- Use case assumes tools allow a user to answer for more than one asset at a time
  - Results should be allowed in a single file
  - Does this use case need to be explicitly defined?

Multiple Result Sets

- Add ResultType under ResultsType to allow 1..* result elements to be included
- Move existing ResultsType items under ResultType
- Keep targets at the Results level
  - Allows reuse of users
  - Systems won't be reused; should they be defined inside each Result instead (more like OVAL)?

Multiple Result Sets

```
<results>
    <result start_time="2012-06-01T00:00:00" end_time="2012-06-01T00:10:00"
    subject_id="ocil:com.example:system:1">
            <questionnaire_results>
            ...
    </result>
    <result … subject_id="ocil:com.example:system:2">
            …
    </result>
    <targets>
            <system id="ocil:com.example:system:1">
                    <ai:computing-device>
                        <ai:hostname>myBox</ai:hostname>
    …
    </targets>
<results>
```

Multiple Result Sets

- Change minoccurs on questionnaires, test_actions, and questions to 0 so results can travel without the content
  - Do we want something like OVAL directives to allow shorter results files (questionnaire results only, no artifacts, etc.)?
  - Add OCIL reference attribute to results element for use when full questionnaire is not included

OCIL Capability Proposals

- Exceptional answer handling
- When_else handler
- Answer restrictions
- Multiple response questions
- Information only
- String question length
- “Other” option
- Sequence questions
- Matching questions

Exceptional answer handling

- When should exceptional answers be available for selection?
  - Spec is not clear
  - Everybody bases display on presence of handler
    - If Test Action has when_not_applicable, give user option of choosing Not Applicable
  - This has consequences
    - A question cannot be displayed without knowing the calling test action
    - A question referenced by multiple test actions may have different allowed exceptional answers

Exceptional answer handling

- Solution – Explicitly identify the allowed exceptional answers in the question
  - If listed in question, user may choose that answer
  - If not listed, may not be chosen
  - Does not impact legal results of a test action or questionnaire, only limits how the user can answer

Exceptional answer handling

```
<numeric_question id="ocil:org.namespace:question:1">
   <question_text>How many times a month do you floss your
   teeth?</question_text>
   <allowed_exceptional_answers>
        <answer>UNKNOWN</answer>
        <answer>NOT_APPLICABLE</answer>
   </allowed_exceptional_answers>
</numeric_question>
```

Exceptional answer handling

- Bonus enhancement – enabled by previous change
  - New processing rule - If no handler for exceptional answer then return that exceptional answer as result
    - Provides a default handler for exceptional answers

when_else handler

- Unhandled answers return ERROR
- Must explicitly handle every response you don't want to return ERROR
  - Depending on question, may be huge number of responses to handle
  - Example: "What is a prime number between 1 and 10000?"
    - Thousands of incorrect answers that should return FAIL

when_else handler

- Solution: when_else handler
  - Added to all types that extend QuestionTestActionType
  - Like an “else” statement
  - If answer not handled, invoke when_else handler
    - If no when_else, still return ERROR
  - Applies only to non-exceptional answers
  - Same capabilities as the other handlers

when_else handler

```
<numeric_question_test_action
   question_ref="ocil:org.namespace:question:1"
   id="ocil:org.namespace:testaction:1">
   <when_equals>
        <result>PASS</result>
        <value>2</value>
        <value>3</value>
        <value>5</value>
        <value>7</value>
        <value>11</value>
        <value>13</value>
        <value>17</value>
        <value>19</value>
        <value>23</value>
   </when_equals>
   <when_else>
        <result>FAIL</result>
   </when_else>
</numeric_question_test_action>
```

Answer restrictions

- No way to constrain answers to open-ended questions → bad data

  Question: What percentage of the day do you sleep?

  Answer: -123456789

- Could try to use handlers for validation…
  - Validity of an answer is unrelated to handling
  - Handlers performing double duty is bad

Answer restrictions

- Solution: Add restrictions to open ended questions
  - Numeric
    - Integer vs. Decimal
      - Decimal precision
    - Min/Max values
  - String
    - Pattern
  - Choice *
    - Number of choices

Answer restrictions

```
<numeric_question id="ocil:org.namespace:question:1">
   <question_text>What percentage of people are left
   handed?</question_text>
   <answer_restriction>
        <datatype>DECIMAL</datatype>
        <precision>2</precision>
        <range>
                <min>0</min>
                <max>100</max>
        </range>
   </answer_restriction>
</numeric_question>

<string_question id="ocil:org.namespace:question:2">
   <question_text>What is a 8 letter word starting with a q that is
   not followed by a u and does not end with s (please use
   lowercase)?</question_text>
   <answer_restriction>
        <pattern>^q[a-tv-z]{6}[a-rt-z]$</pattern>
   </answer_restriction>
</string_question>
```

Multiple Response questions

- Choice questions only support a single answer
- OCIL spec highlights this, suggests workaround
- We want to select multiple answers

Multiple Response questions

- Solution: Modify Choice questions to handle multiple answers
  - Backwards compatible – nothing changes for traditional Choice questions

Multiple Response questions

- ChoiceQuestionType
  - Optional attribute “multi” (true/false)
  - default_answer_ref → list of Choice IDs
  - answer_restriction
    - When using multi, limit selections in answer

Multiple Response questions

```
<choice_question id="ocil:org.namespace:question:1" multi="true"
   default_answer_ref="ocil:org.namespace:choice:1
   ocil:org.namespace:choice:2 ocil:org.namespace:choice:3
   ocil:org.namespace:choice:4 ocil:org.namespace:choice:5">
   <question_text>Which days of the week do you
   work?</question_text>
   <choice id="ocil:org.namespace:choice:1">Monday</choice>
   <choice id="ocil:org.namespace:choice:2">Tuesday</choice>
   <choice id="ocil:org.namespace:choice:3">Wednesday</choice>
   <choice id="ocil:org.namespace:choice:4">Thursday</choice>
   <choice id="ocil:org.namespace:choice:5">Friday</choice>
   <choice id="ocil:org.namespace:choice:6">Saturday</choice>
   <choice id="ocil:org.namespace:choice:7">Sunday</choice>
</choice_question>
```

Multiple Response questions

- ChoiceQuestionTestActionType
  - when_choice – no changes
  - when_count
    - Set of choices
    - How many must be selected – all/any/operator + value
  - when_choices
    - Operator – AND, OR, XOR, ONE
    - Negate
    - Set of choice_ref, choice_count, and choices

Multiple Response questions

```
<choice_question_test_action question_ref="ocil:org.namespace:question:1"
    id="ocil:org.namespace:testaction:1">
    <when_choice>
          <result>PASS</result>
          <choice_ref>ocil:org.namespace:choice:6</choice_ref>
    </when_choice>
    <when_count>
          <result>PASS</result>
          <check_count operation="greater than or equal">2</check_count>
          <choice_ref>ocil:org.namespace:choice:1</choice_ref>
          <choice_ref>ocil:org.namespace:choice:2</choice_ref>
          <choice_ref>ocil:org.namespace:choice:3</choice_ref>
          <choice_ref>ocil:org.namespace:choice:4</choice_ref>
    </when_count>
    <when_choices>
          <result>FAIL</result>
          <choices operator="OR">
                    <choice_ref>ocil:org.namespace:choice:7</choice_ref>
                    <choice_count>
                              <check_count operation="less than">2</check_count>
                              <choice_ref>ocil:org.namespace:choice:1</choice_ref>
                              <choice_ref>ocil:org.namespace:choice:2</choice_ref>
                              <choice_ref>ocil:org.namespace:choice:3</choice_ref>
                    </choice_count>
          </choices>
    </when_choices>
</choice_question_test_action>
```

Information only

- Everything in OCIL evaluates to PASS/FAIL
- How can you ask a question without having to decide if the answer is good or bad?
  - Simple data collection – completing a survey
  - Data aggregation – combine multiple answers
  - Offline processing – using the data outside of OCIL

- Unacceptable to say “Ignore the result status, I just want your data”

Information only

- Solution: New ExceptionalResultType value INFO_ONLY
  - Use as result of Questionnaire/Test Action when just want to collect data and not judge
  - Has no impact on questionnaire result calculation logic
  - Cannot be selected by the user
  - Will map to “informational” in XCCDF

Display hints

- OCIL purposely doesn't dictate how the questions are presented to users…
- But being able to include something to influence the display may be useful
  - "What is your name?" vs. "What did you do on your summer vacation"
  - Not just aesthetic concern, but impediment to data entry

Display hints

- Solution: Extensible hints added to each QuestionType

  Extensible base type to allow custom vendor hints

  \+ Type specific standard hints

  - String question
    - essay
    - short_answer
  - Choice question
    - dropdown
    - radio

Display hints

```
<string_question id="ocil:org.namespace:question:2"
   display_hint="short_answer">
   <question_text>What is your name?</question_text>
</string_question>

<string_question id="ocil:org.namespace:question:4"
   display_hint="essay">
   <question_text>What did you do on your summer
   vacation?</question_text>
</string_question>

<string_question id="ocil:org.namespace:question:5"
   display_hint="ocil:hint:vendor_x:mycustomformat">
   <question_text>How else would you like this question
   displayed?</question_text>
</string_question>
```

“Other” option

- Commonly seen in surveys:

What is your favorite flavor of ice cream?

- Vanilla
- Chocolate
- Strawberry
- Other (please specify) _____

- To do the same in OCIL today – annoying
  - Choice question with 4 choices
  - If 4th choice selected, ask a string question

“Other” option

- Solution: “other_option” added to Choice question
  - Allows inline “follow-up” as initial response
  - Full capability of numeric and string structures
    - Same handlers
    - Same restrictions

“Other” option

```
<choice_question id="ocil:org.namespace:question:1">
    <question_text>What is your favorite flavor of ice cream?</question_text>
    <choice id="ocil:org.namespace:choice:1">Vanilla</choice>
    <choice id="ocil:org.namespace:choice:2">Chocolate</choice>
    <choice id="ocil:org.namespace:choice:3">Strawberry</choice>
    <other_option type="string"/>
</choice_question>

<choice_question_test_action question_ref="ocil:org.namespace:question:1"
    id="ocil:org.namespace:testaction:1">
    <when_choice>
          <result>PASS</result>
          <choice_ref>ocil:org.namespace:choice:1</choice_ref>
          <choice_ref>ocil:org.namespace:choice:3</choice_ref>
    </when_choice>
    <when_choice>
          <result>FAIL</result>
          <choice_ref>ocil:org.namespace:choice:2</choice_ref>
    </when_choice>
    <when_other>
          <when_pattern>
                    <result>PASS</result>
                    <pattern>^.*berry.*$</pattern>
          </when_pattern>
          <when_pattern>
                    <result>FAIL</result>
                    <pattern>^.*mint.*$</pattern>
          </when_pattern>
    </when_other>
</choice_question_test_action>
```

Sequence questions

- Questions where user is asked to put items in order

Put these events in chronological order:

- America is “discovered”
- Neil Armstrong walks on the moon
- OCIL 2.0 is released
- Invention of the wheel

Sequence questions

- SequenceType
  - Ordered set of ChoiceType instances
- SequenceQuestionType
  - List of choices – just like a Choice question
  - Option to randomize display
  - Default answer – a sequence
- SequenceQuestionTestActionType
  - when_sequence handlers – contain a sequence

Sequence questions

```
<sequence_question id="ocil:org.namespace:question:1">
    <question_text>Put these events in chronological order, starting with the
    oldest</question_text>
    <choice id="ocil:org.namespace:choice:1">America is "discovered"</choice>
    <choice id="ocil:org.namespace:choice:2">Neil Armstrong walks on the moon</choice>
    <choice id="ocil:org.namespace:choice:3">OCIL 2.0 is released</choice>
    <choice id="ocil:org.namespace:choice:4">Invention of the wheel</choice>
    <default_answer>
           <choice_ref>ocil:org.namespace:choice:2</choice_ref>
           <choice_ref>ocil:org.namespace:choice:3</choice_ref>
           <choice_ref>ocil:org.namespace:choice:1</choice_ref>
           <choice_ref>ocil:org.namespace:choice:4</choice_ref>
    </default_answer>
</sequence_question>


<sequence_question_test_action id="ocil:org.namespace:testaction:1"
    question_ref="ocil:org.namespace:question:1">
    <when_sequence>
           <result>PASS</result>
           <sequence>
                      <choice_ref>ocil:org.namespace:choice:4</choice_ref>
                      <choice_ref>ocil:org.namespace:choice:1</choice_ref>
                      <choice_ref>ocil:org.namespace:choice:2</choice_ref>
                      <choice_ref>ocil:org.namespace:choice:3</choice_ref>
           </sequence>
    </when_sequence>
</sequence_question_test_action>
```

Matching questions

- Questions where user is asked to match items from one list with items from another list

Match these baby animals to their non-baby counterparts:

List A	List B
• Kitten	• Cow
• Puppy	• Dog
• Calf	• Cat

Matching questions

- MatchingPairType
  - one choice from set A, one choice from set B
- MatchingQuestionType
  - 2 sets of choices – set A and set B
  - Option to randomize display of each set
  - 1 to 1 vs. 1 to many
- MatchingQuestionTestActionType
  - when_matches handlers
    - List of matching pairs
    - How many must be matched to satisfy handler

Matching questions

```
<matching_question id="ocil:org.namespace:question:1">
   <question_text>Match these baby animals to their non-baby
   counterparts</question_text>
   <set_A>
        <choice id="ocil:org.namespace:choice:1">Kitten</choice>
        <choice id="ocil:org.namespace:choice:2">Puppy</choice>
        <choice id="ocil:org.namespace:choice:3">Calf</choice>
   </set_A>
   <set_B>
        <choice id="ocil:org.namespace:choice:4">Cow</choice>
        <choice id="ocil:org.namespace:choice:5">Dog</choice>
        <choice id="ocil:org.namespace:choice:6">Cat</choice>
   </set_B>
</matching_question>
```

Matching questions

```
<matching_question_test_action id="ocil:org.namespace:testaction:1"
   question_ref="ocil:org.namespace:question:1">
   <when_matches>
        <result>PASS</result>
        <check>all</check>
        <match>
                <item_A>ocil:org.namespace:choice:1</item_A>
                <item_B>ocil:org.namespace:choice:6</item_B>
        </match>
        <match>
                <item_A>ocil:org.namespace:choice:2</item_A>
                <item_B>ocil:org.namespace:choice:5</item_B>
        </match>
        <match>
                <item_A>ocil:org.namespace:choice:3</item_A>
                <item_B>ocil:org.namespace:choice:4</item_B>
        </match>
   </when_matches>
</matching_question_test_action>
```

Next Steps

- Update specifications and start formal acceptance process
- Watch the lists for information
- Does OCIL need a formal adoption program?