ACS

Chemistry for Life®

American Chemical Society

American Chemical Society

ACS Publications - Ensuring XML Quality

Tamara Stoker and Keith Rose

The ACS Vision and Mission

Our Vision:

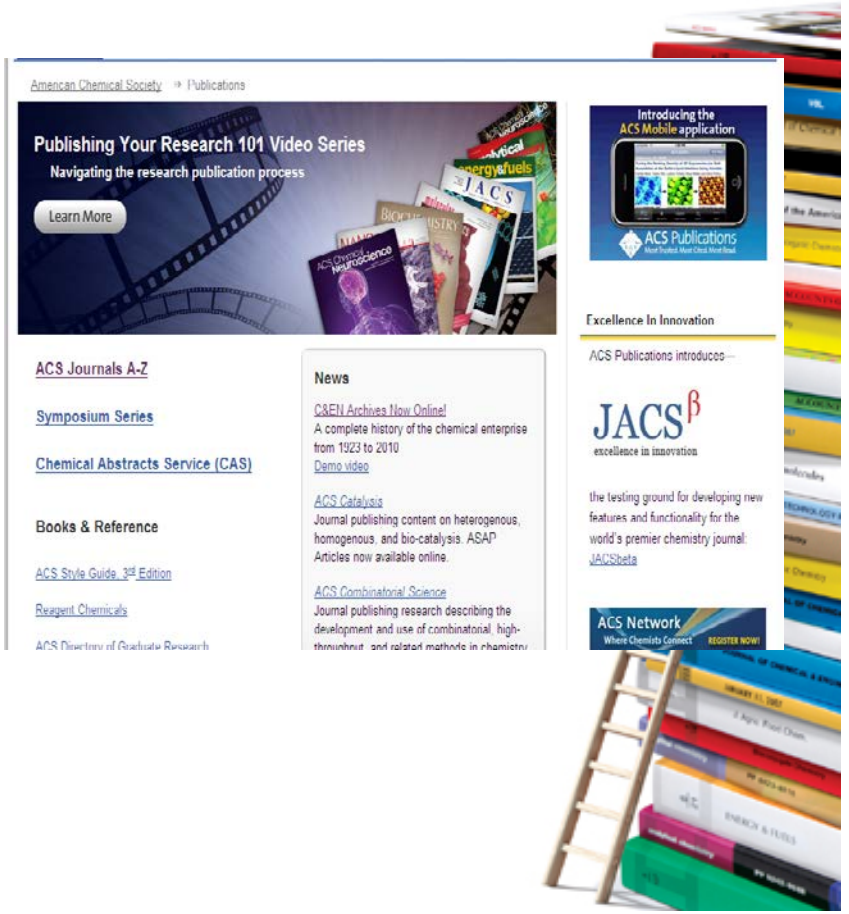
"Improving people's lives through the transforming power of chemistry"

Our Mission:

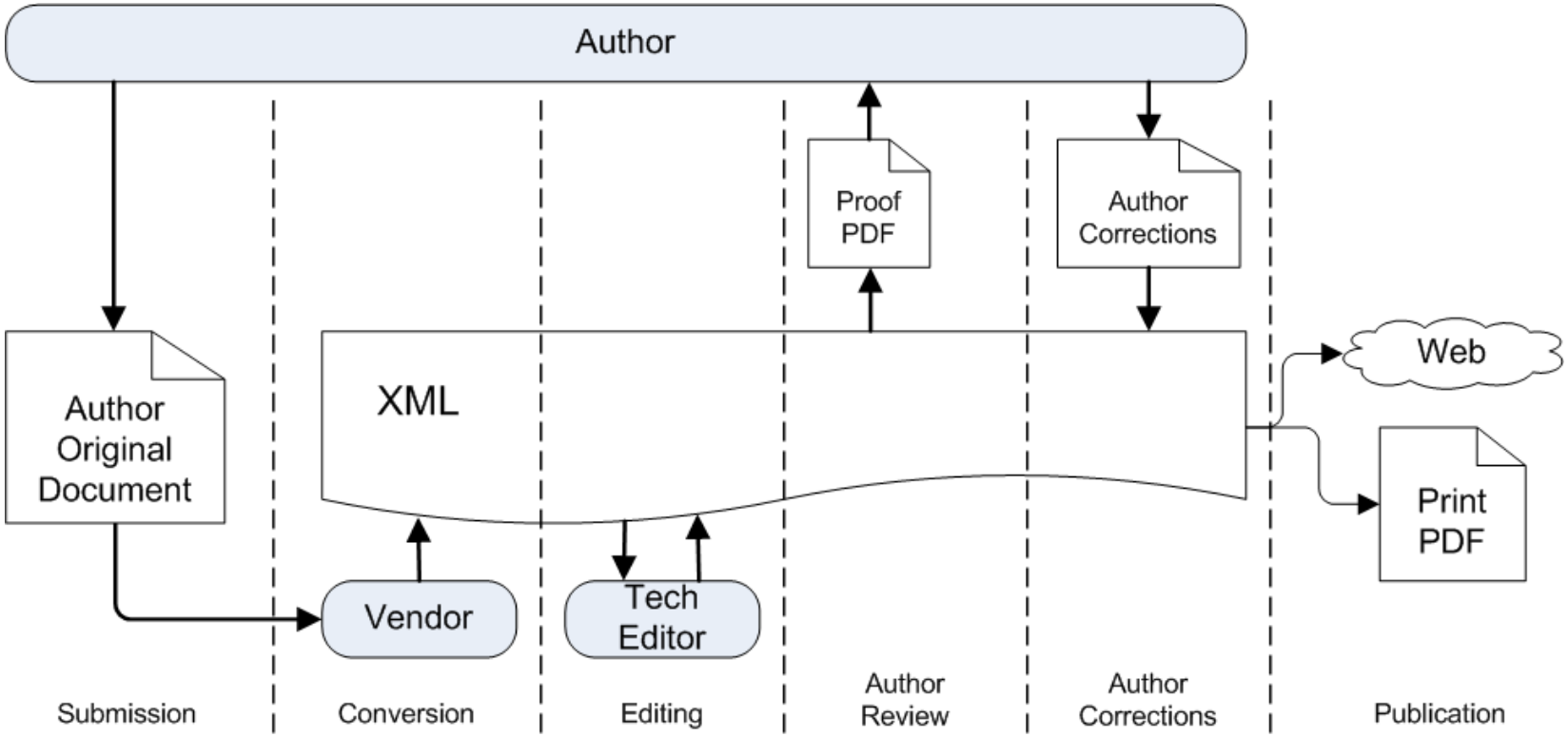
"Advance the broader chemistry enterprise and its practitioners for the benefit of Earth and its people."

ACS Publications:
High Quality, High Impact

- Now with 41 peer-reviewed journals including two new journals for 2012
- ~40,000 articles (manuscripts) to be published in 2012
- ~325,000 composed pages



The ACS Manuscript Workflow

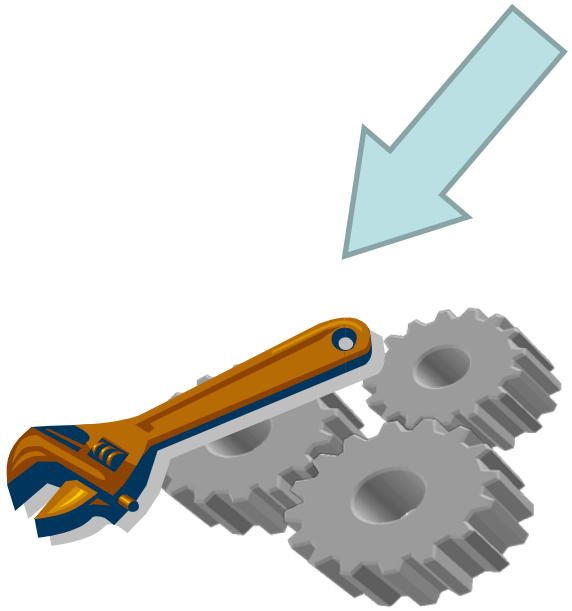


Trouble Ahead?

<pub-date><year>2012</year></pub-date>

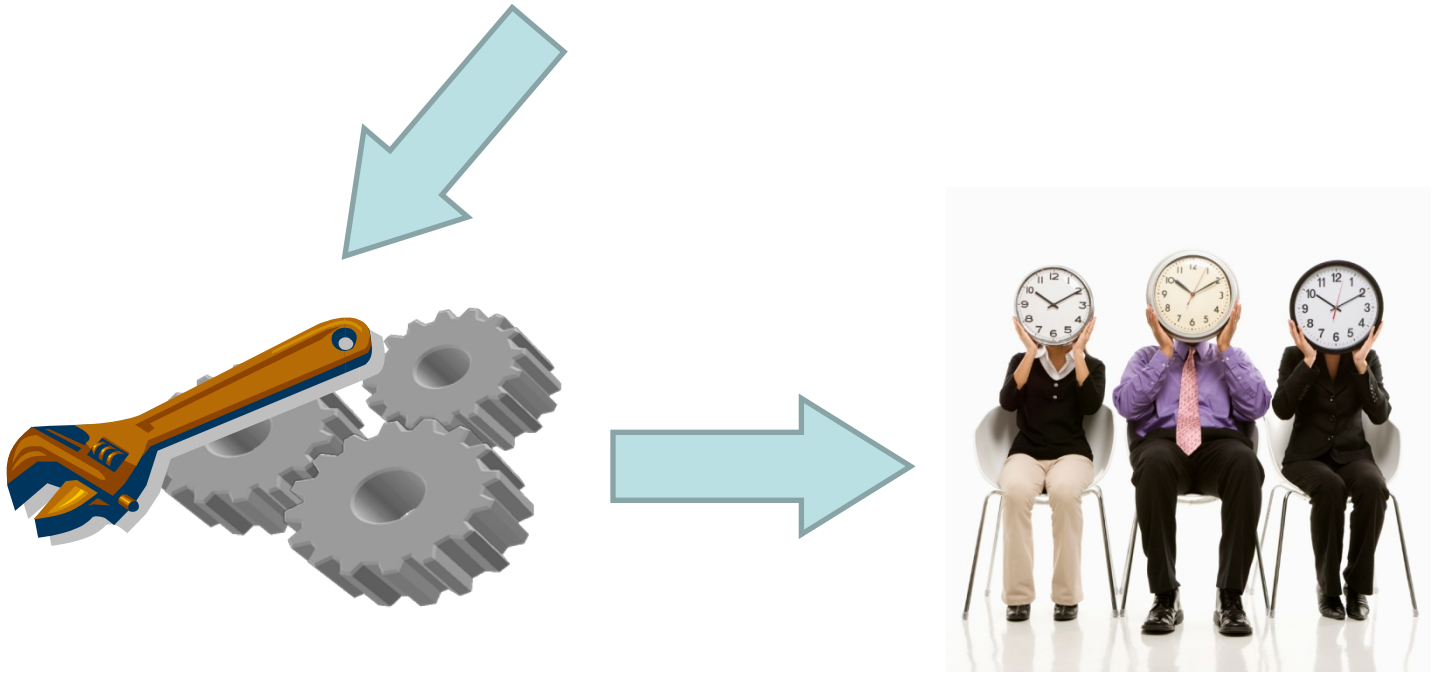
Trouble Ahead?

<pub-date><year>2012</year></pub-date>



Trouble Ahead?

<pub-date><year>2012</year></pub-date>



Dilemma

~40,000 articles

Dilemma

~325,000 pages!

Solution

- Fix What You Can
- Highlight The Rest

Automatic Edits
Fix What You Can

- Run before Technical Editors first see the article (pre-edits)
  - Simple content changes
  - Markup changes
- Run after Technical Editors finish (post-edits)
  - Markup clean-up
- THOUSANDS! Heavily developed early in the transition to XML, resulted in lots of time saved

Pre-Edit (Example)

Label and head elements whose content is enclosed entirely inside certain formatting tagging will have that formatting tagging stripped.

- Properties

```
#### ExtraFormatEdit constants
# These are the target elements to search for extraneous formatting tags.
ExtraFormatEdit.extraFormattingTargetElementPath = //label
ExtraFormatEdit.extraFormattingTargetElementPath = //head

# These are formats to strip from target elements (if the format applies to the entire
# target).
ExtraFormatEdit.extraFormattingTag = bold
ExtraFormatEdit.extraFormattingTag = italic
ExtraFormatEdit.extraFormattingTag = underline
ExtraFormatEdit.extraFormattingTag = overline
ExtraFormatEdit.extraFormattingTag = sc
```

- Before:

```
<label><bold>Label 1</bold></label>
```

After:

```
<label>Label 1</label>
```

Pre-Edit (Expression-based Examples)

Change “C=C” (carbon/carbon double-bond). Replace the equal sign (“=“) with the preferred Unicode double bond character (#xE5FB).

- Code:

```
T            \bC=C\e          C&#xE5FB;C                All
```

Change the capital “L” in L-arginine to use markup for “small-cap” (<sc>) anywhere in the body of the document.

- Code:

```
T            L-arginine       <sc>l</sc>-arginine       Body
```

Pre-Edit (Expression-based Example, with exclusions)

Change the British spelling of the word “aluminium” to “aluminum,” except in certain elements (<ack>, <aff>, etc.).

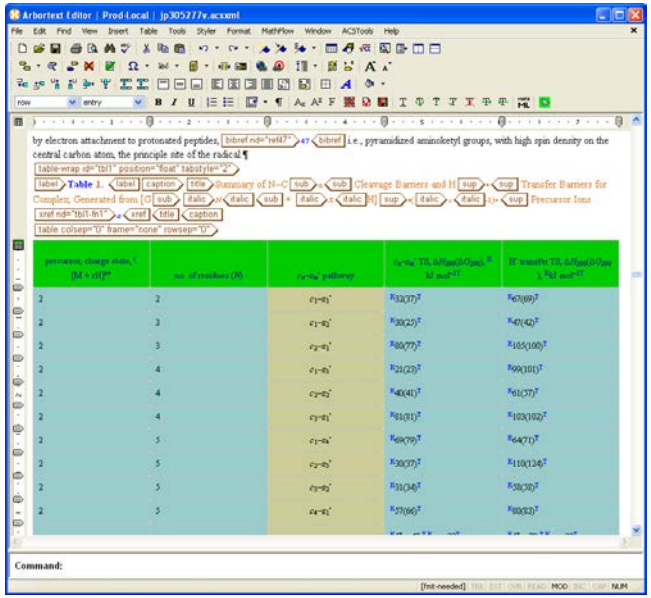
- Code:

```
T       aluminium       aluminum        All
F       aluminium       aluminum        ack
F       aluminium       aluminum        aff
F       aluminium       aluminum        author-notes
F       aluminium       aluminum        bio
F       aluminium       aluminum        contrib
F       aluminium       aluminum        corresp
F       aluminium       aluminum        disp-quote
F       aluminium       aluminum        index-entry-group
F       aluminium       aluminum        ref
F       aluminium       aluminum        ref-note
F       aluminium       aluminum        run-head-author
F       aluminium       aluminum        speech
```

Exclusions

Post-Edit (Example)

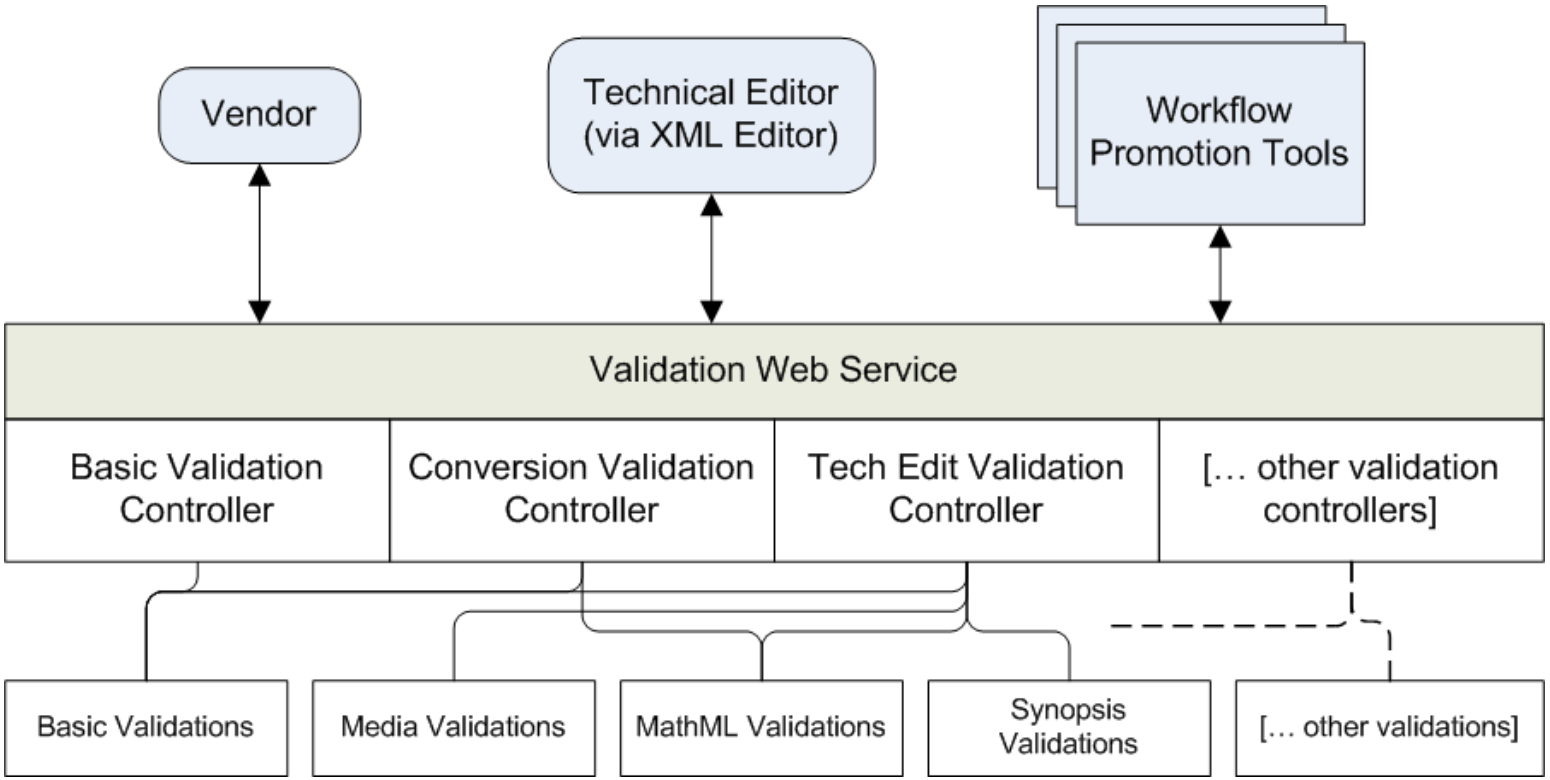
Clean up markup by removing extraneous processing instruction elements. These particular PIs provide cell-shading in tables in the XML editor. They are useful for Technical Editors while they are editing the tables, but may be removed after editing is complete.



```
// Find all cell-coloring processing
// instructions, and remove them.
NodeList cellColoringPIs =
          acsJournalUtil.xpathFindNodes(doc,
          "//processing-instruction('Pub')[contains(.,'_cellfont')]");

acsJournalUtil.removeNodes(cellColoringPIs);
```

Validations



Validations (Simple Example)

The <journal-meta> element must be present, and it must contain a <journal-id> element, which must contain exactly two characters.

```
xmlTestXPath(vid: 'journal-id',
  desc:'Journal Id must be present and be 2 characters long',
  context:'//journal-meta/journal-id',
  condition:'string-length(.)=2',
  messageId:'exact-string-length',
  severity:ACSPubsMessage.FATAL[…]
```

Validations (Simple example, cont'd)

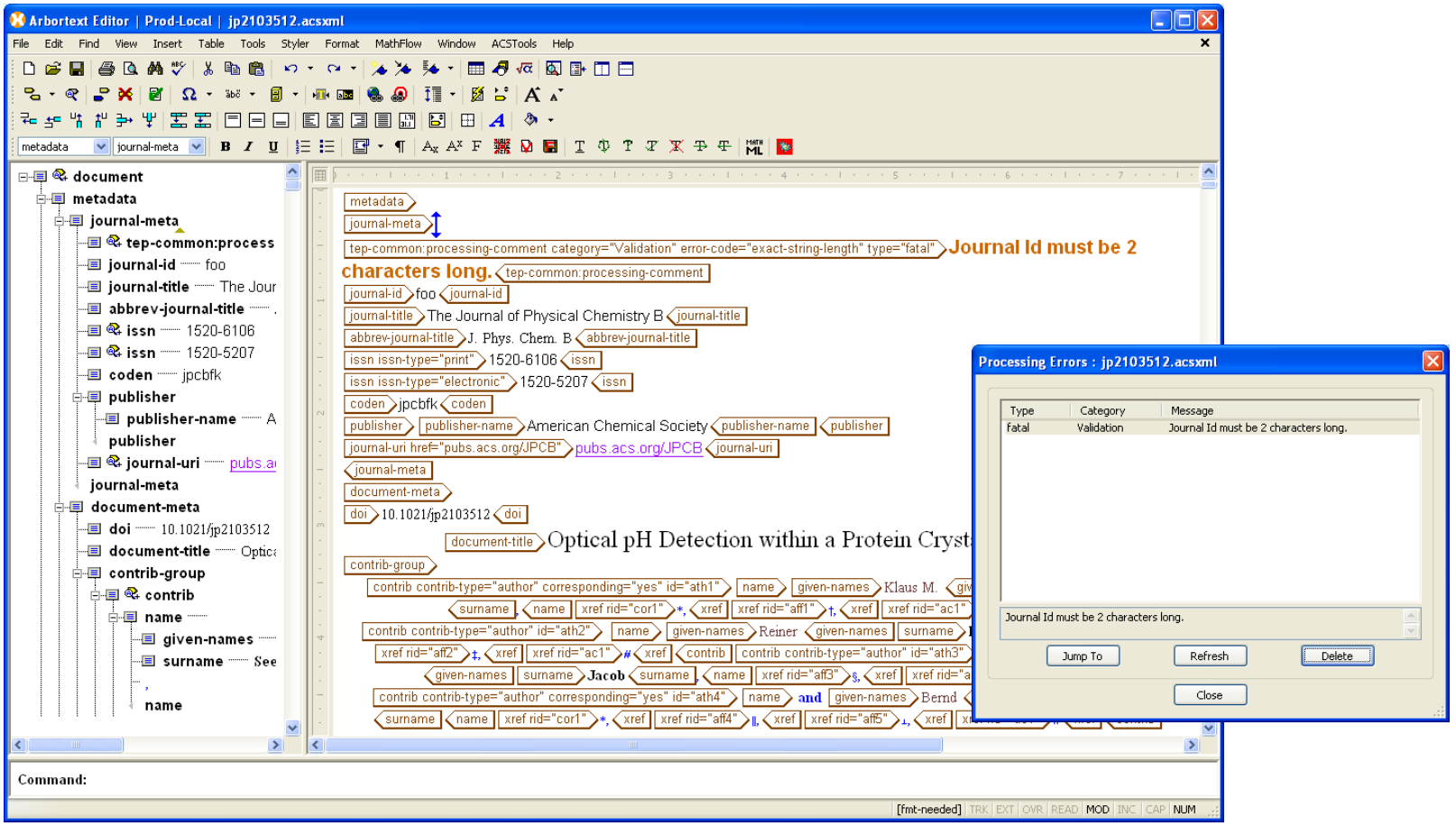
- Before:

```
<journal-meta>
  <journal-id>foo</journal-id>
  <journal-title>Industrial &amp;amp; Engineering Chemistry Research</journal-title>
  […]
</journal-meta>
```

- After:

```
<journal-meta>
  <tep-common:processing-comment category="Validation“
                  error-code="exact-string-length" type="fatal">
    Journal Id must be 2 characters long.
  </tep-common:processing-comment>
  <journal-id>foo</journal-id>
  <journal-title>Industrial &amp;amp; Engineering Chemistry Research</journal-title>
  […]
</journal-meta>
```

Validations (Simple example, cont'd) – What the TE Sees



Validations (Example of external dependency)

A “Supporting Information” file associated with an article exists in Documentum. Ensure that at least one <si> element makes reference to it.

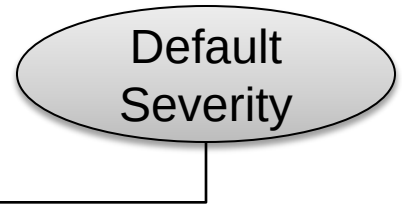
```
for ( $siFileName in [list of SI files for this MSC] )
{
  xmlTestXPath(vid:'filename-matches-href',
    desc:'filename must match the href for a si tag',
    context:'//document-meta',
    condition:"count(./si-group/si[@href='$siFileName']) > 0",
    messageId:'no-si-filename-match',
    params: acsFileName)
}
```

Validations (Example of changing severity)

One or more <si> elements exist, but no “Supporting Information” files exist in Documentum for this article.

```
if ($number_Of_SI_Files_In_Repository_For_This_Manuscript == 0)
{
  xmlTestXPath(vid:'no-si-files-in-dots',
    desc:'no si files in DOTS, any si is extra',
    context:'//si[not(ancestor::tep-common:deleted-content)]',
    condition:'false',
    messageId:'extra-si-tag')
}

[…]
message(id:'extra-si-tag', severity:ACSPubsMessage.FATAL,
  usageSeverity:['Compose':ACSPubsMessage.WARNING,
                 'TechEdValidationController':ACSPubsMessage.ERROR],
  category:TEPValidationUtil.CATEGORY,
  'Callout for this <si> tag does not match any SI filename in DOTS.')
```



Outsourcing - Vendor XML Quality

- Vendor relations manager: determine outsourcing value to ACS
- Vendor quality measures implemented in three ways:
  - Vendor XML scoring
  - Vendor validations
  - Vendor manual rejections

Vendor Scoring

- XML conventions

For elements that allow text, any deterministic leading and trailing white space within element content should be avoided or moved outside of the tag. […] the pink spaces below should be omitted, and the blue spaces should be relocated into the adjacent text:

Incorrect

```
:
    <head> Methods &amp; Materials </head>
    <fn> <label> d </label> per<bold> 2x
    </bold>d.</fn>
```

Correct

```
:
    <head>Methods &amp; Materials</head>
    <fn><label>d</label>per <bold>2x</bold> d.</fn>
```

Vendor Scoring: Scorecard

- XML “goodness” equates to time to edit
- Scoring integrated in existing workflow
- Conversion scorecard

Vendor Scoring: Sample Scorecard

	Score
EQUATIONS	
Minutes to fix equations that were **not** keyed that should have been?	2
Minutes to fix equations that were **not** MathML that should have been?	0
Minutes to make the paper match author copy (beyond 2 items above)	0
Equation Subtotal	**2**

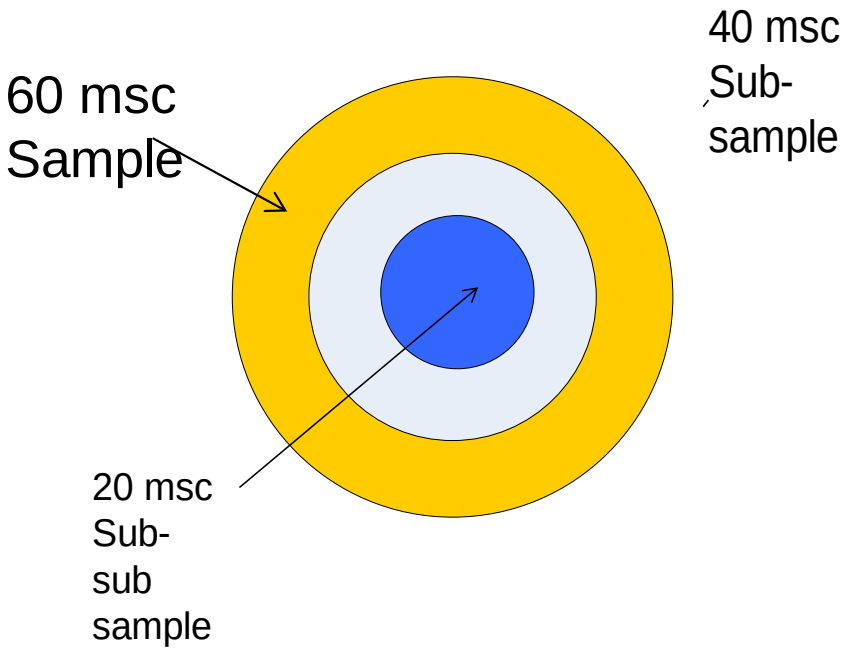
CITATIONS	
Minutes to fix citations having the wrong model?	1
Minutes to fix citations having the wrong type?	0
Minutes to fix missing or incorrectly tagged citations in the text?	0
Minutes to fix miscited bibrefs for multipart citations?	2
Minutes to fix any other citation problems (beyond 4 items above)	1
Citations Subtotal	**4**

Vendor Scoring: Mechanism

- Challenges: 40,000 plus manuscripts and 60+ Technical Editors (TEs)
- Scoring team established
- Statistically consistent random sampling

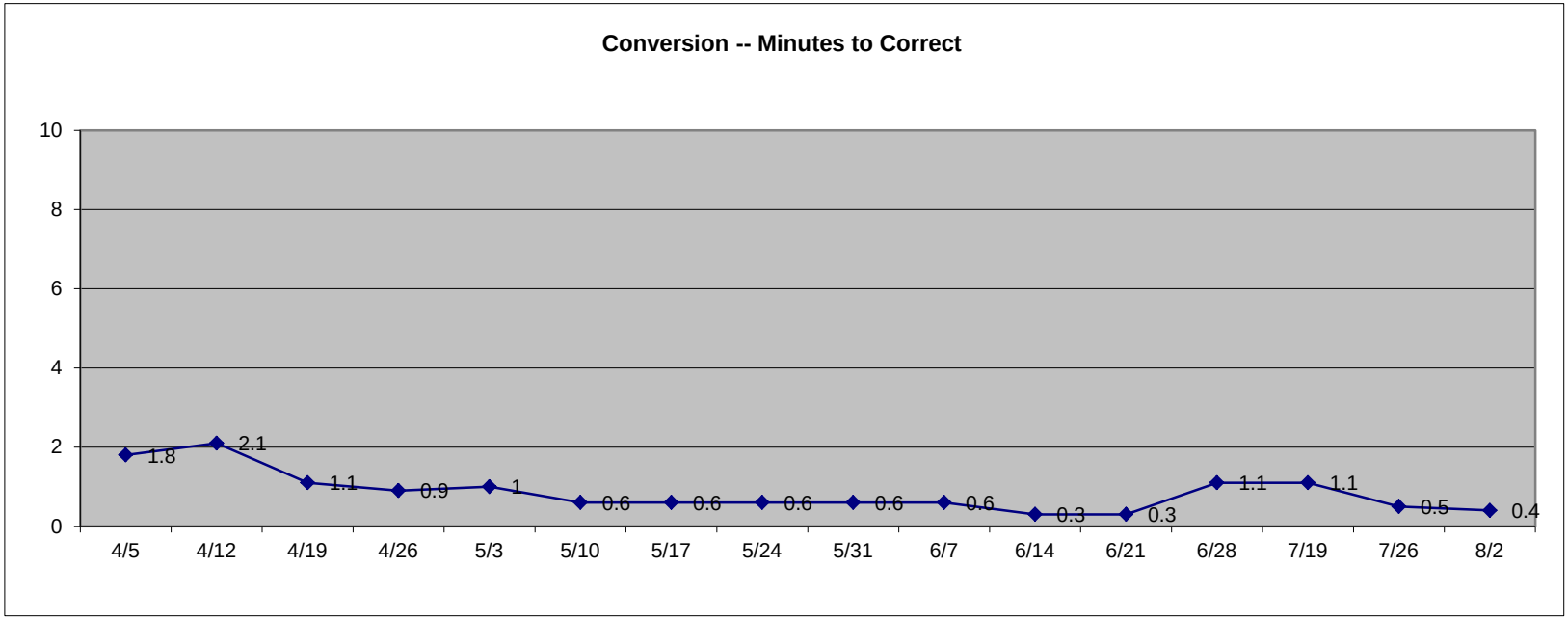
Vendor Scoring: Mechanism

- Statistically consistent random sampling



Vendor Scoring: Results

- Achieved an accurate weekly picture of vendor quality
- Improvement due to ACS and vendor focusing on and measuring the same issues
- Timely feedback allows ACS to respond quickly to environmental change



Vendor Scoring: Results

Vendor Conversion Scorecard																						
Manuscript		E1	E2	E3	S1	S2	S3	C1	C2	C3	C4	C5	M1	X1	T1	T2	T3	TG1	TG2	Total		
nl302113r	7/23/2012	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1		
jp301329g	7/24/2012	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ie3000553	7/24/2012	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	2		
am300951f	7/26/2012																					
Averages		0	0	0	0	0	0	0	0	0	0	0.3	0	0	0.7	0	0	0	0	1		

Vendor Validations

- Vendor calls ACS web service to invoke validations program
- Example:
  - Validation failures were detected for document np200906s

    Journal: np

    Msc Type: r-Review

    FATAL: The content of element type "metadata" must match "(journal-meta,document-meta,processing-meta?)".

Vendor Rejects Transactions

- Vendor rejects a transaction containing XML vendor composition system can't handle
- System developed to automate these rejections and four steps taken
  - Step 1: specific vendor instructions entered as note in our workflow system
  - Step 2: error code is looked up in table to find generic instructions
  - Step 3: the specific and generic instructions become text of an email sent to appropriate ACS staff
  - Step 4: manuscript auto-routed to appropriate stage of workflow
- Results include improved publication time and timely feedback to Technical Editors
- Rejection metrics are used to guide other quality efforts

Conclusion

- XML Quality
  - Measure
  - Auto-correct
  - Validate
  - Improve
- Monitoring to understand thus ensure current quality
- Continuous monitoring to spark future improvement efforts

Contact Us

- Tamara Stoker
  - tstoker@acs.org
- Keith Rose
  - krose@acs.org