PHUSE 2012 - Budapest

Deep Dive into ODM Validation

Ronald Steinhau – Entimo AG

Content

- Dimensions of ODM-Validation
- Special Problems
- Configurable ODM-Validation
- Test & Performance Aspects
- Scala - an effective solution Language
- Conclusion

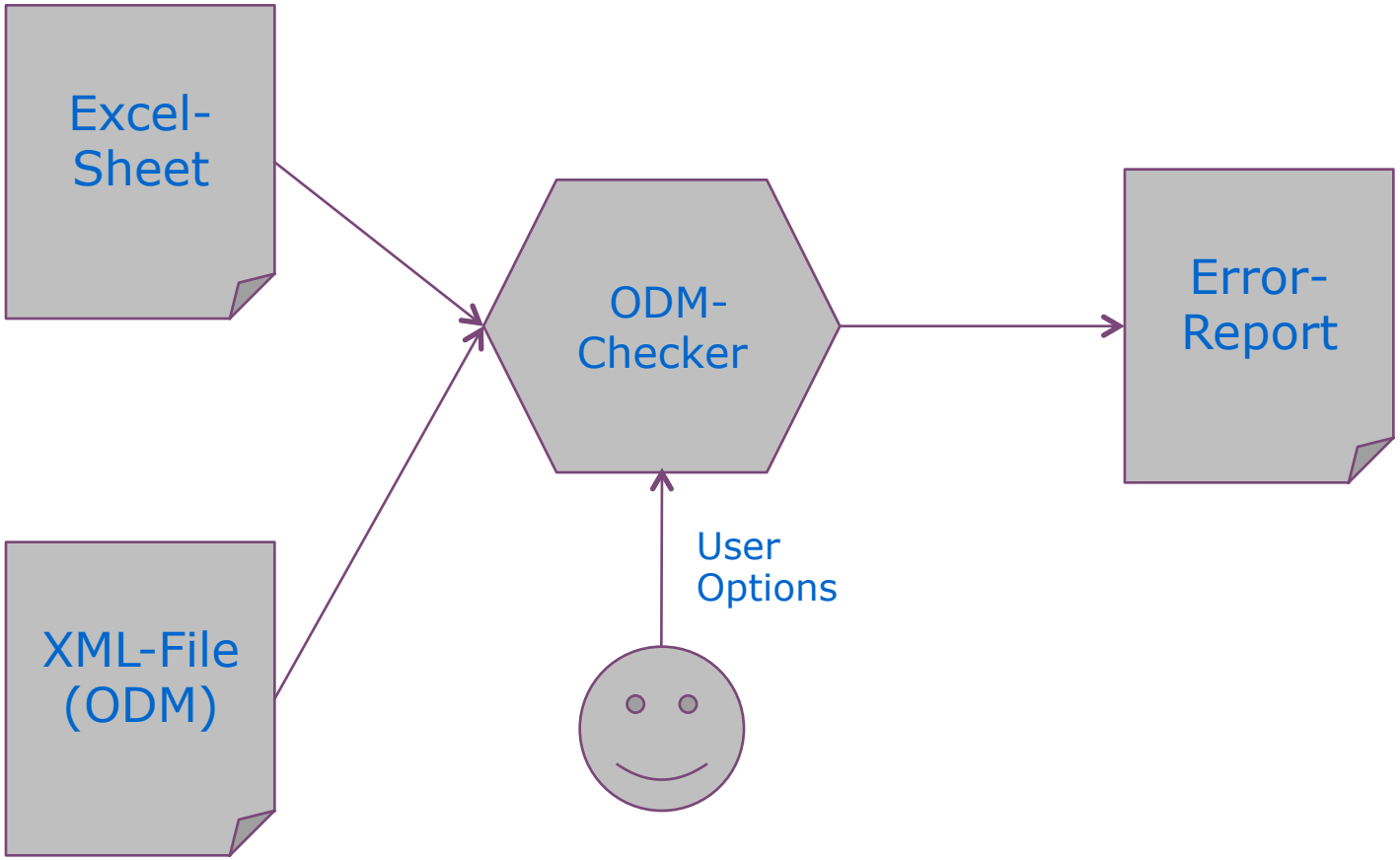
Dimensions of ODM-Validation

- Moving ODM Standards – slightly different checks
  - 1.2, 1.2.1 and 1.3 are already in focus
- Different types of checks
  - Structural checks (like XML-Schema check)
  - Value checks (only partially covered by XML-Schema)
  - Referential integrity checks (not covered by XML-Schema)
- ODM extensions
  - Define.xml – additional checks of all types
  - Custom extensions – individual checks helpful
- Useful error report
  - Why and where
  - Useful context information of errors
  - Adjustable error level

Special Problems

- Namespace related checks
  - Multiple versions of metadata per study
  - Multiple studies
  - Multiple forms
  - … can all have legal duplicate OID's
- Special Checks
  - SAS variable names length
  - SAS missing values
  - Value Lists for attributes
- Activating/Deactivating checks
  - By category
  - By extension
  - By single check

Configurable ODM-Validation (Overview)



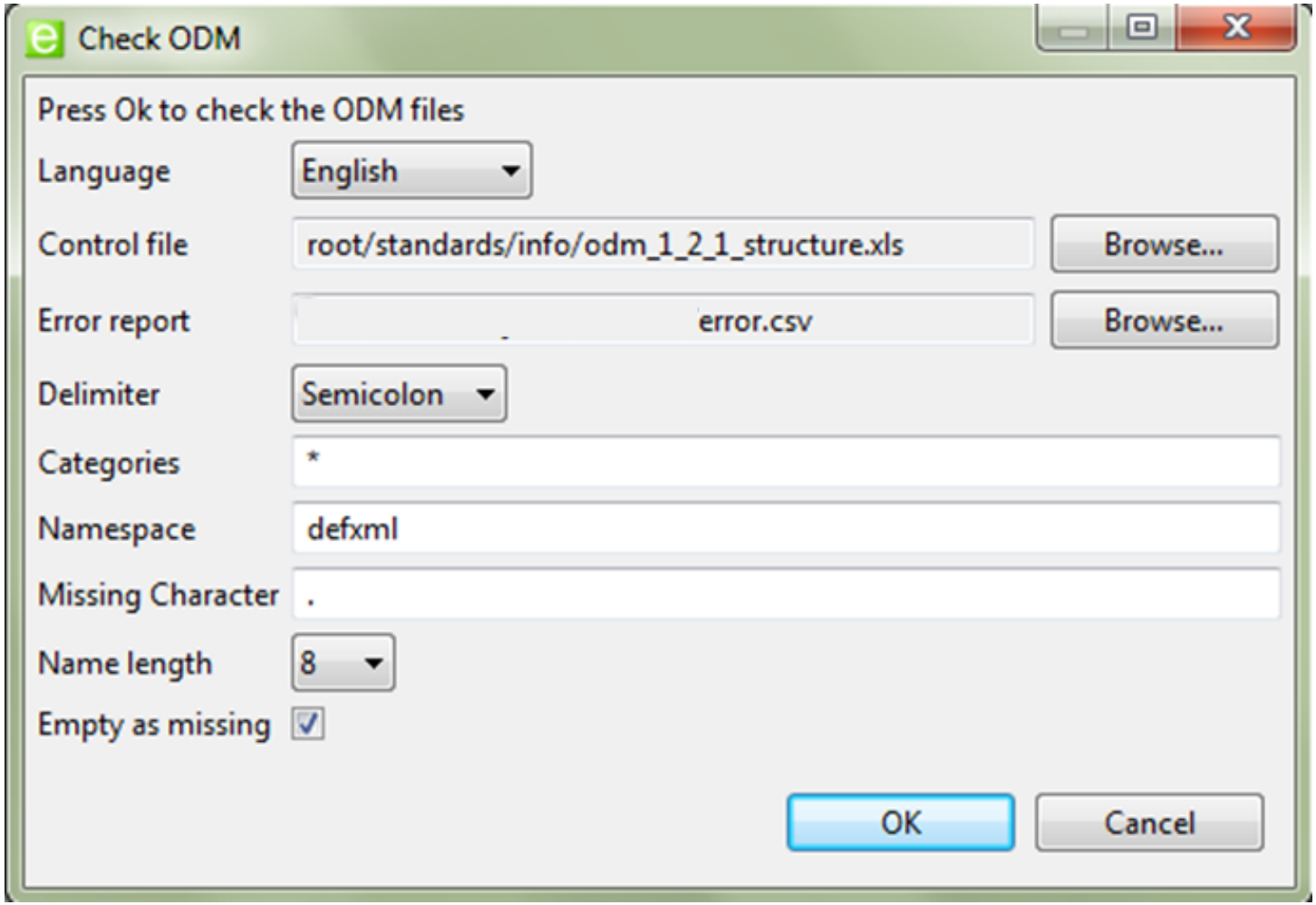
Configurable ODM-Validation

- Validation control by Excel sheet
  - One file/sheet for each ODM version
  - Extendable (just add/change lines)
    - Add custom extension checks
    - Future: add value patterns
    - Future: add custom check functions
  - Configurable
    - Identify categories
    - Identify extensions
  - Adaptable
    - Map name extensions
  - Can mimic XML-Schema like structural checks

EXCEL Control Sheet (partial)

element	no	parent	kind	name	type	value_list	locationOI	uniqueatt	val	category	cardin	chec	odm_
MeasurementUnit	35	BasicDefinitions	Body	Symbol						General	1:1	Y	
Symbol	36	MeasurementU	Body	TranslatedText						General	1:n	Y	
TranslatedText	37	Symbol	Attribute	xml:lang	languageTag					General	0:1	Y	
TranslatedText	38	Symbol	Body		text					General	1:1	Y	
MetaDataVersion	39	Study	Attribute	OID	oid			Y		Metadata	1:1	Y	
MetaDataVersion	40	Study	Attribute	Name	name			Y		Metadata	1:1	Y	
MetaDataVersion	41	Study	Attribute	Description	text					Metadata	0:1	Y	
MetaDataVersion	42	Study	Body	Include						Metadata	0:1	Y	
MetaDataVersion	43	Study	Body	Protocol						Metadata	0:1	Y	
MetaDataVersion	44	Study	Body	StudyEventDef						Metadata	0:n	Y	
MetaDataVersion	45	Study	Body	FormDef						Metadata	0:n	Y	
MetaDataVersion	46	Study	Body	ItemGroupDef						Metadata	0:n	Y	
MetaDataVersion	47	Study	Body	ItemDef						Metadata	0:n	Y	
MetaDataVersion	48	Study	Body	CodeList						Metadata	0:n	Y	
MetaDataVersion	49	Study	Body	ImputationMethod						Metadata	0:n	Y	
MetaDataVersion	50	Study	Body	Presentation						Metadata	0:n	Y	
Include	51	MetaDataVersic	Attribute	StudyOID	oidref		Study	C		Metadata	0:1	Y	
Include	52	MetaDataVersic	Attribute	MetaDataVersionOI	oidref		MetaDataV	C		Metadata	0:1	Y	
Protocol	53	MetaDataVersic	Body	StudyEventRef						Metadata	1:n	Y	
StudyEventRef	54	Protocol	Attribute	StudyEventOID	oidref		StudyEvent	Y		Metadata	1:1	Y	
StudyEventRef	55	Protocol	Attribute	OrderNumber	integer			Y		Metadata	0:1	Y	
StudyEventRef	56	Protocol	Attribute	Mandatory		Yes | No			N	Metadata	1:1	Y	
StudyEventDef	57	MetaDataVersic	Attribute	OID	oid			Y		Metadata	1:1	Y	
StudyEventDef	58	MetaDataVersic	Attribute	Name	name			Y		Metadata	1:1	Y	
StudyEventDef	59	MetaDataVersic	Attribute	Repeating		Yes | No			N	Metadata	1:1	Y	
StudyEventDef	60	MetaDataVersic	Attribute	Type		Scheduled | Unscheduled | Comr			N	Metadata	1:1	Y	
StudyEventDef	61	MetaDataVersic	Attribute	Category	text					Metadata	0:1	Y	

Check ODM Dialogue



Sample Error Report

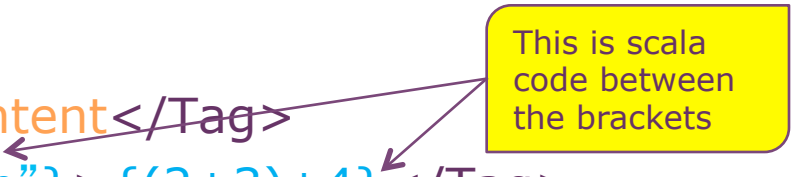
Line(s)	Check No	Tag/Type	Error
			user: ima
			check-datetime: 2011/Dez/22 16:49:55
			checking-rules: odm_1_2_1_structure.xls
			checked-file: C:_TEMP\\odm-checker\\workspace\\tstODM\\testdata\\.\\odm999.xml
			language: en
			odm-types: defxml
43	491	<ItemRef	Attribute Mandatory = 'YesNo' is not one of Yes | No
44	492	<ItemRef	Attribute KeySequence = '-..' is not of type integer
44	494	<ItemRef	Attribute Role = ' ' is not of type NMTOKENS
45	490	<ItemRef	Attribute OrderNumber = '-..' is not of type integer
51	56	<StudyEv	Attribute Mandatory = 'YesNo' is not one of Yes | No
53	55	<StudyEv	Attribute OrderNumber = '-..' is not of type integer
57	64	<FormRef	Attribute OrderNumber = '-..' is not of type integer
65	59	<StudyEv	Attribute Repeating = 'NoYes' is not one of Yes | No
65	60	<StudyEv	Attribute Type = 'CommonUnscheduled' is not one of Scheduled | Unscheduled | Common
66	65	<FormRef	Attribute Mandatory = 'NoYes' is not one of Yes | No
70	72	<ItemGro	Attribute OrderNumber = '-..' is not of type integer
82	89	<ItemRef	Attribute KeySequence = '-..' is not of type integer
82	91	<ItemRef	Attribute Role = ' ' is not of type NMTOKENS
83	87	<ItemRef	Attribute OrderNumber = '-..' is not of type integer
88	78	<ItemGro	Attribute SASDatasetName = '0...' is not of type sasName

Test & Performance Aspects

- Tests
  - Dozens of test-files prepared
  - All corner-cases covered
  - Automated test-runs to check all test-files
- Validation Performance
  - Parallel checks (structure, value, references)
  - Use of XML streaming (a must for large files)
  - Referential Integrity checks can increase memory usage
    - Because of possible forward referencing

Scala - an elegant solution Language

- Scala – a better and functional Java
  - XML build-in
    - val a = <Tag attr="x">content</Tag>
    - Val b = <Tag attr={"a"+"b"}>{(2+3)+4}</Tag>
  - First-Class function objects
    - val mul = (a:Int,b:Int) => a * b
      - Same as: def mul (a:Int, b:Int) = a*b;
    - val x = mul(3,4)
  - JVM compatible
    - Scala classes can use all Java classes (incl. types)
      Java classes can use Scala classes



Scala – an elegant solution Language

- Scala Actors
  - Easy concurrency
  - Messages (immutable arguments) sent to processors
  - Management of Queues built-in

```
class ValueChecker extends Actor {
   override def receive = {
      case x:Elem=>
        allAttributesToCheck.foreach { attr =>
           val value = x ~~ attr
           // check value for pattern …
        }
   }
}
```

Elem is an XML element (the built-in scala class for it)

A custom defined operator to extract the attribute value from XML

Conclusion

- ODM Validation – grows with future needs
  - XML-Schema check alone is not enough
  - Use configurable approaches
  - Allow custom extensions
  - Benefit from multi-core processing
- Scala
  - A promising language for XML-processing
  - Java-Interoperability guarantees future safety
  - Fast and elegant