A Classification of DICOM Constraints

Dongbai Guo

db.guo@oracle.com
Oracle

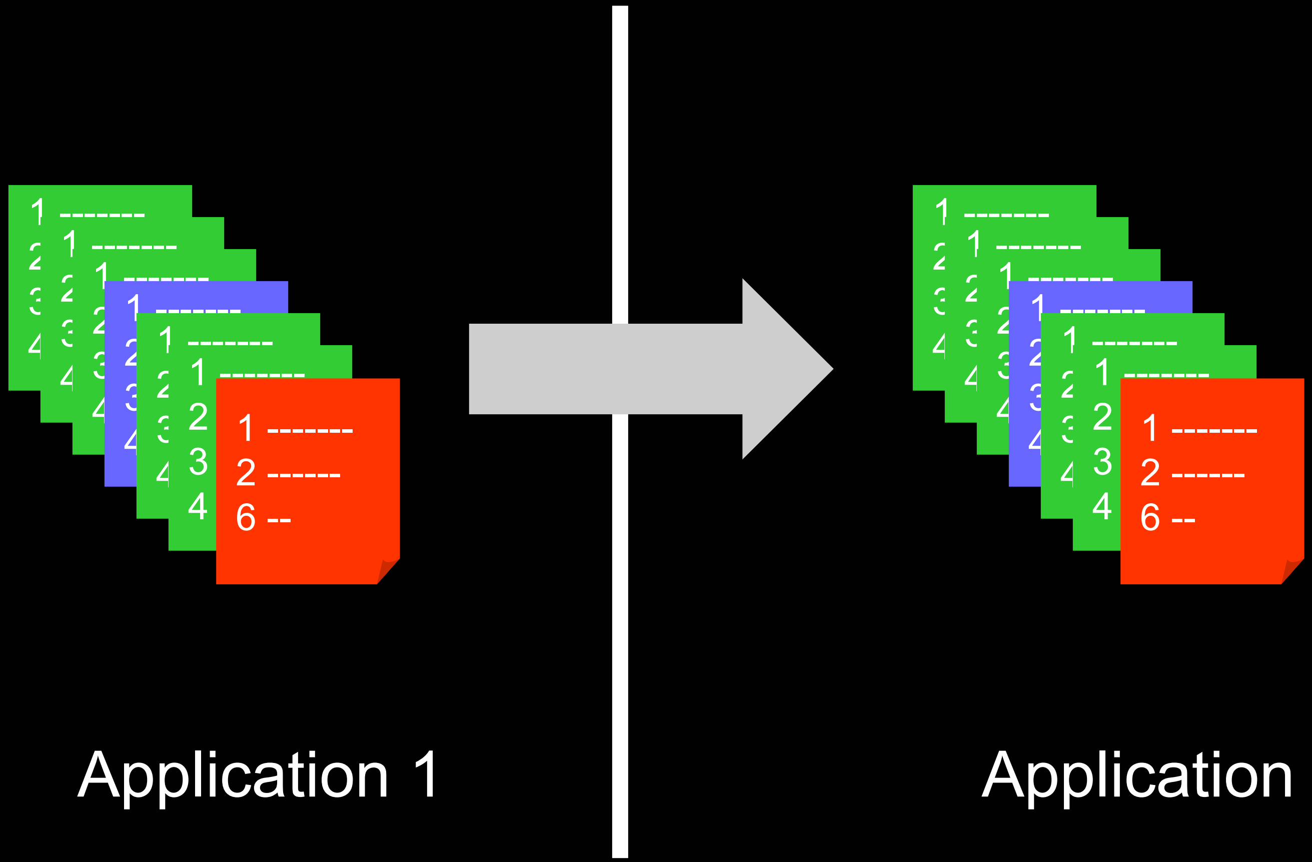
Agenda

- DICOM constraint, definitions
- XML and constraints
- Modeling DICOM constraints
- Classification of DICOM constraints
- Exceptions
- Implementation

Constraints

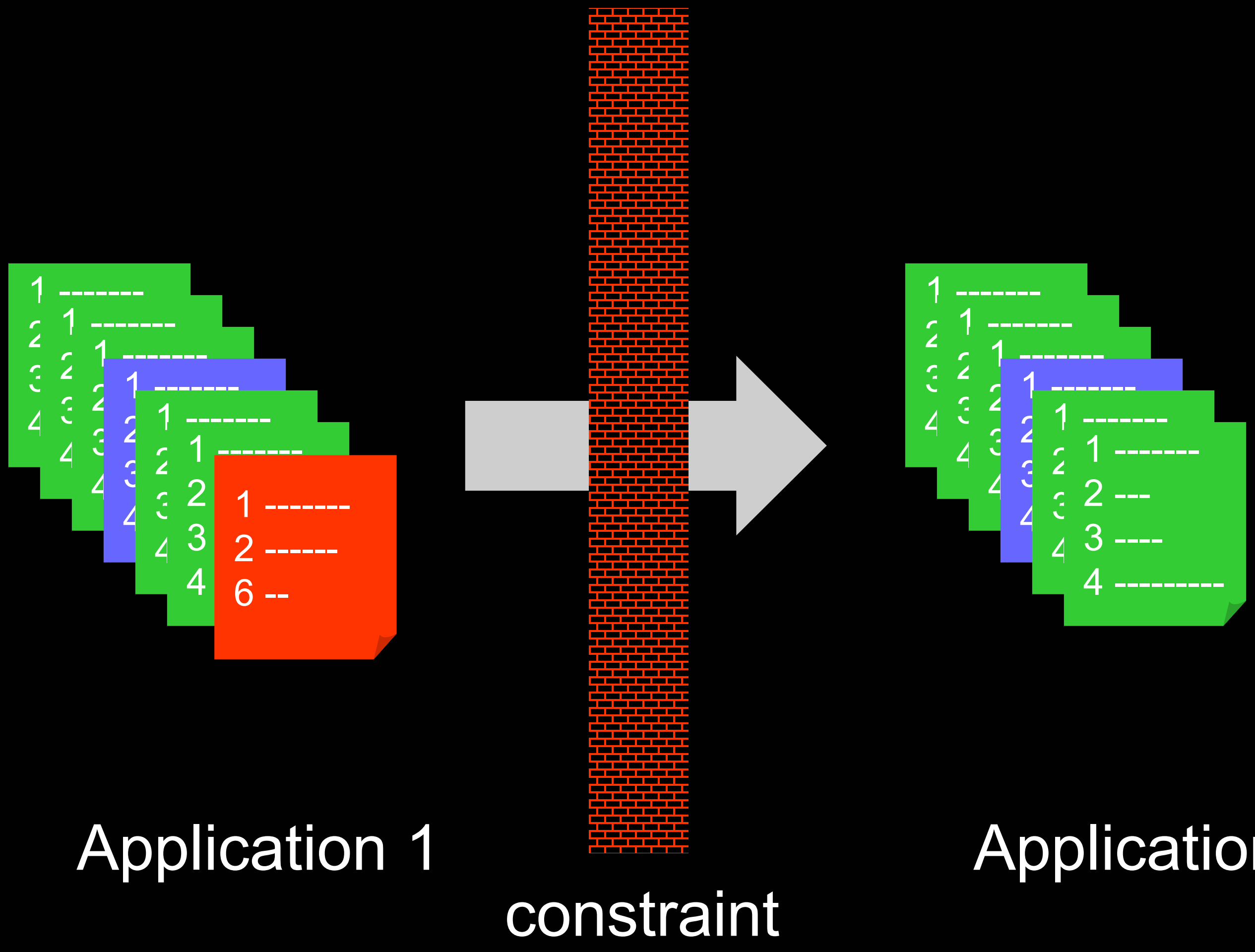
- DEFINITION: Structural and semantic relationships of or between DICOM attributes that can be verified by the receiving party to confirm the conformance of a DICOM object

Exchange DICOM Object: Without Constraints



ORACLE

Exchange DICOM Object: With Constraints



What is Not a Constraint

- **C.7.6.4 Contrast**/bolus
  - Required if contrast media was used in this image
  - In table A. 2-1 CT Image IOD modules, PS3.3-2004, page 95
- **C.10.6 Spatial transformation module**
  - Required if rotation or flipping are to be applied to referenced image(s)
  - **In table A.33-1**, PS 3.3 – 2004, page 155

Characteristics of DICOM Constraints

- A constraint may involve one or more attributes
- A constraint can often be expressed as one or more predicates
- A constraint predicate can be mapped into a procedural language construct
- A constraint can be efficiently validated if a small set of properties is computed for the involved DICOM attributes
- Certain predicates are repeatedly referenced in the definition of a DICOM object

Constraints and XML

- XML representation of DICOM metadata is used frequently for application integration
- Constraint can be defined external to XML document
- XML schema can enforce certain types of constraints

Drawbacks of Enforcing DICOM Constraints with XML Schema

- It is difficult and sometimes impossible to express DICOM constraints with XML schema
- XML schema with strong constraints
    - Is difficult to evolve
    - Rejects non-conformant DICOM objects
    - Cannot be customized
    - Can be cumbersome and therefore inefficient to manage

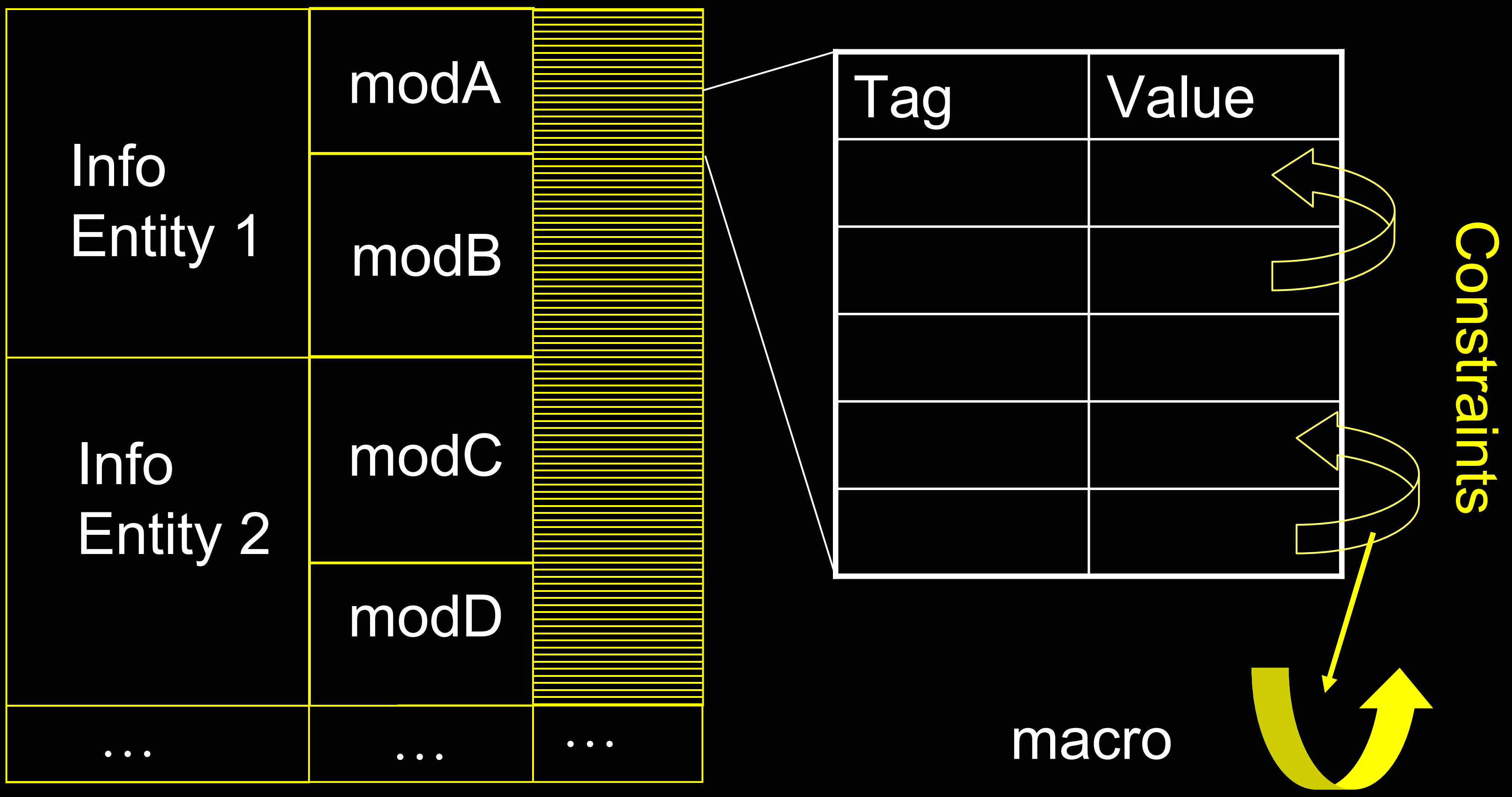
Modeling DICOM Constraints

- Predicates
  - DICOM specific functions
    - value, cardinality, length, notNull, exist
  - Logical operators
    - AND, OR, NOT, XOR, ➔
  - Relational operators
    - >, <, ==, >=, <=, !=, in, like, isPattern
  - Tag addressing
- Macros
- SOP class dependent
- Express most, not all, DICOM constraints

Predicate Grammar

- constraint ::= { (predicateDef)+ }
- predicateDef ::= name := predicate
- predicate ::= (predicate OP1 predicate) | (! predicate) | exprB | name
- exprB ::= funB(tag) | expr OP2 expr | expr in {string+}
- expr ::= value(tag) | cardinality(tag) | length(tag) | string
- funcB ::= notNull(tag) | exist(tag)
- OP1 ::= && | || | XOR | ➔
- OP2 ::= > | < | == | != | >= | <= | => | like | ispattern
- Tag ::= tag.tag | HHHHHHHH | tag:int | var

Modeling DICOM Components



Modeling Attribute Type

- Type 1, mandatory not null
  - `notNull(tag)`
- Type 2, tag value can be null if unknown
  - `exist(tag)`

Conditional Presence

- Type 1C and type 2C
  - `(predA ➔ ( notNull(tag) )`
  - `(predA ➔ ( exist(tag) )`
- Context Group Local Version (0008,0107) 1C
  - Required **if the value of Context Group Extension Flag (0008,010B) is "Y".**
  - PS3.3-2004, Table 8-8-1, page 69

    `(Value(0008010B)==“Y”) ➔ notNull(00080107)`

ORACLE

A Rough Categorization of DICOM Constraints

- Enumeration
- Cardinality
- Reference integrity
- Choice
- Context sensitive structure
- Context sensitive date value

Enumeration

- Attribute takes one of the enumerated values
- Patient Sex (0010, 0040)
  - **Enumerated Values**: “M”, ”F”, “O”
  - PS3.3-2004, Table C.2-3, page 214
  - `value(00100040) in { “M”, “F”, “O”}`
  - Equivalent to

```
( (value(00100040)== “M”) ||
  (value(00100040)== “F”) ||
  (value(00100040)== “O”)  )
```

Cardinality

- Value multiplicity or number of data values
- For sequence attributes
  - Anatomic Region Sequence (0008,2218), Zero or one Item may be present in this Sequence PS3.3-2004, Table 10-6, page 74
  - `cardinality(00082218)<=1`

Reference Integrity

- External UID references
- Attribute reference
  - Color lookup table descriptor (0028,110x)
  - Color lookup table data (0028,120x)
  - Name of physicians reading study (0008,1060)
  - Physicians reading study identification sequence (0008,1062)

Choice

- Only one out of many candidate structures may occur in a DICOM object
- Only one of following attributes shall be present: {institution code sequence (0008,0082), institution name (0008,0080)}, PS3.3-2004, Table 10-1, page 71
    - Predicate: `exist(00080080) XOR exist(00080082)`

Context Sensitive Structure

- The value of an attribute determines the structure of a DICOM object
- Common in DICOM structure report
  - Example **Table C.17-4 SR document content module attributes,** PS 3.3 – 2004, Page 780
  - Can be broken down to attribute level predicates

    ```
     (value(0040A040)==“TEXT” ➔
    notNULL(0040A160))
    ```
- Type 1C attributes, required if a sequence item is present

Context Sensitive Data Values

- The interpretation of one DICOM attribute is determined by the value of another
  - For example, attributes of value representation LO, LT, PN, SH, UT depend on the value of the character set attribute (0008,0005)
  - For certain implementations, the value of such attributes may map to two different data types, character vs. wide character

Exceptions and Oddities

- Not every condition is a constraint
- Person Identification Code Sequence (0040,1101)
    - The code meaning attribute of VR LO/PN
    - PS3.3-2004, Table 10-1, page 70

Implementation

- Logging is an integral part of constraint validation
- Constraint rules should be customizable
- Separation of compilation and runtime
- Supporting macros is important

Implementation: Compilation

- Compilation
  - Macro substitution
  - Type promotion and casting
  - Syntax and semantics checking
  - From constraints to OPTREE
  - Move to persistent storage
  - Preconditions

Summary

- Canonical validation rule definition
  - Unambiguous constraints
  - Precise conformance
  - Enhanced readability
  - Concise specification
  - Easier implementation
  - Better performance

QUESTIONS
ANSWERS