UNIVERSITY OF
OXFORD

Domain-Specific Languages for Interdisciplinary Research

Jeremy Gibbons
University of Oxford

GSDP workshop on DSL4EE, Marstrand, June 2011

1. Robustness in modelling

A trend towards greater heterogeneity in scientific research:

- large, distributed teams
- long-running collaborations
- dynamic organization
- variety of stakeholders
- interdisciplinary interests

Implicitly shared context becomes untenable.

Assumptions should be *explicit*, *documented*, *transparent*, *checkable*.

1.1. Transparency and repeatability

A cornerstone of the scientific method: *show your working*.

- model system
- perform experiments
- analyse data
- publish results

(This is especially important in controversial fields, such as economics and climate change!)

1.2. Computational science

Much ‘working’ these days is digital:

- spreadsheets
- databases
- MatLab and Mathematica workbooks
- Perl scripts
- workflows…

‘Performing experiments’ amounts to running simulations.

Now what does ‘show your working’ mean?

1.3. Climategate

UK House of Commons Science and Technology Committee report into UEA CRU:

> data disclosed in publications should be accompanied by sufficient detail of computer programmes, specific methodology or techniques used to analyse the data, such that another expert could repeat the work

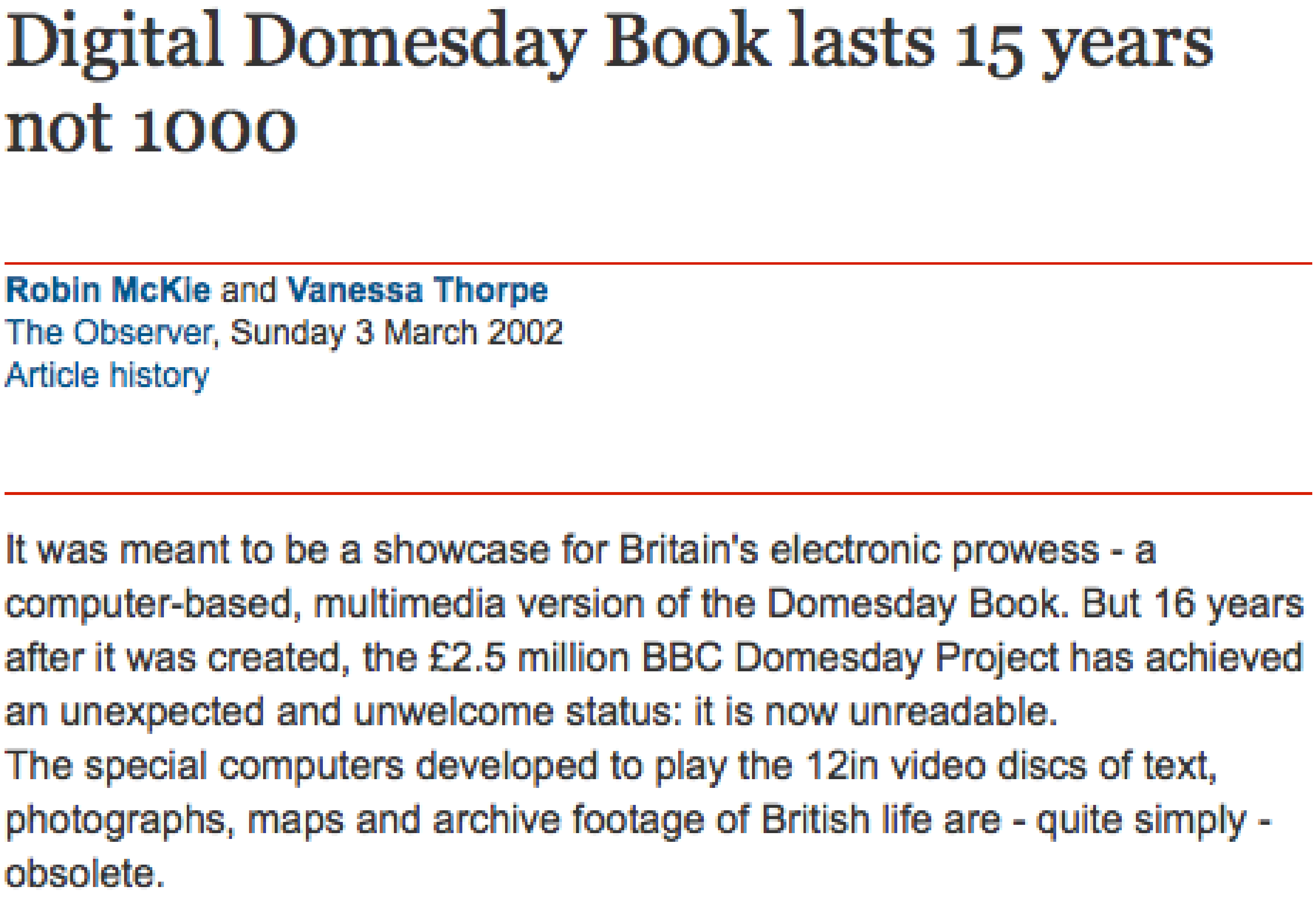
2500 SCIENTISTS SAY WE'VE CAUSED GLOBAL WARMING
I'D LIKE A SECOND OPINION
TANDBERG

1.4. Software verification and validation

Open source is no silver bullet:

- dependencies on libraries, OS, other vagaries
- bit rot
- run-time configurations: workflow
- proprietary systems

1.5. Digital preservation is difficult



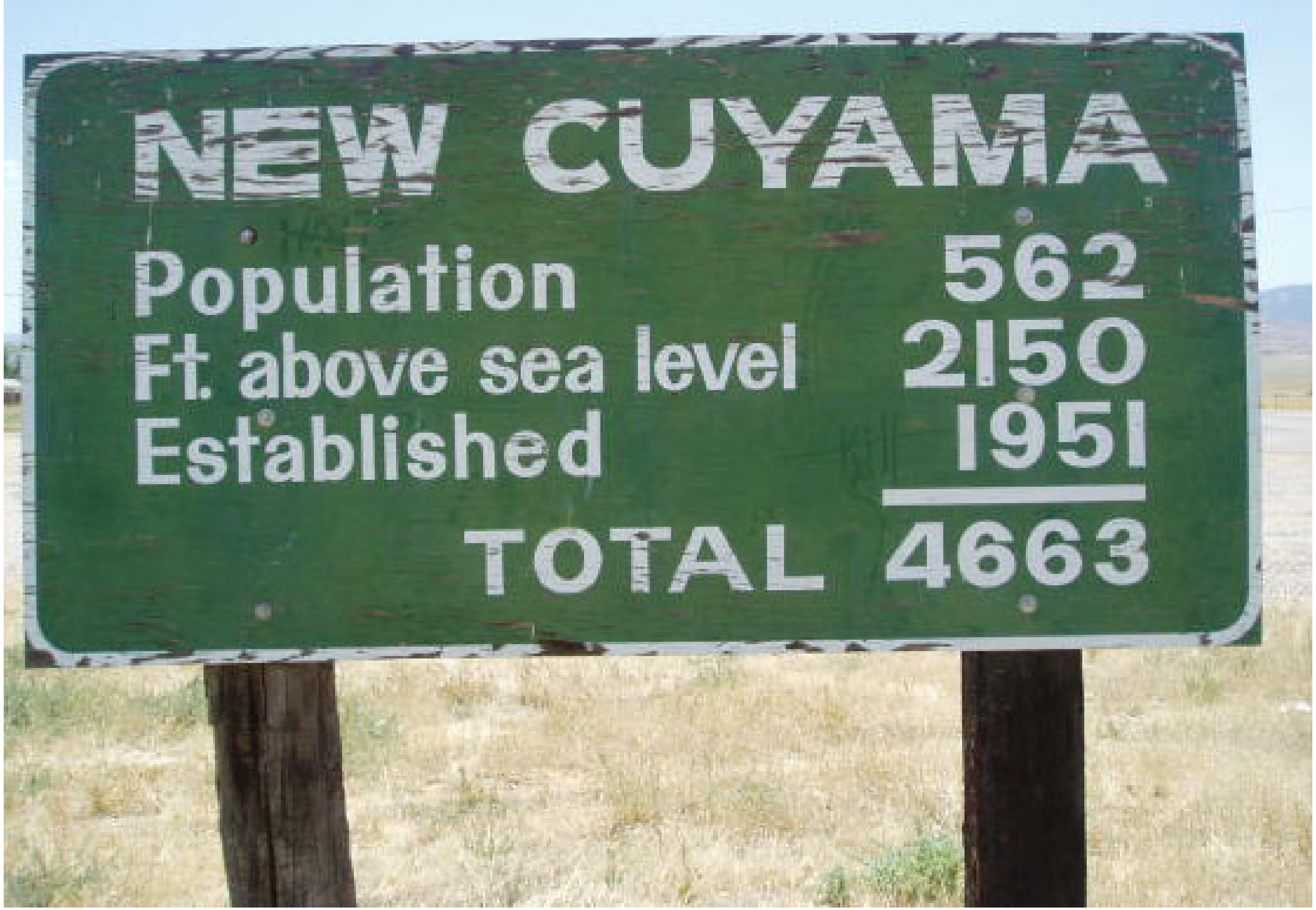
Digital Domesday Book lasts 15 years not 1000

Robin McKie and Vanessa Thorpe
The Observer, Sunday 3 March 2002
Article history

It was meant to be a showcase for Britain's electronic prowess - a computer-based, multimedia version of the Domesday Book. But 16 years after it was created, the £2.5 million BBC Domesday Project has achieved an unexpected and unwelcome status: it is now unreadable.

The special computers developed to play the 12in video discs of text, photographs, maps and archive footage of British life are - quite simply - obsolete.

1.6. And as for digital trustworthiness. . .



1.7. Exploratory versus dependable development

The issue of software verification and validation is even more challenging in applied fields than in professional development.

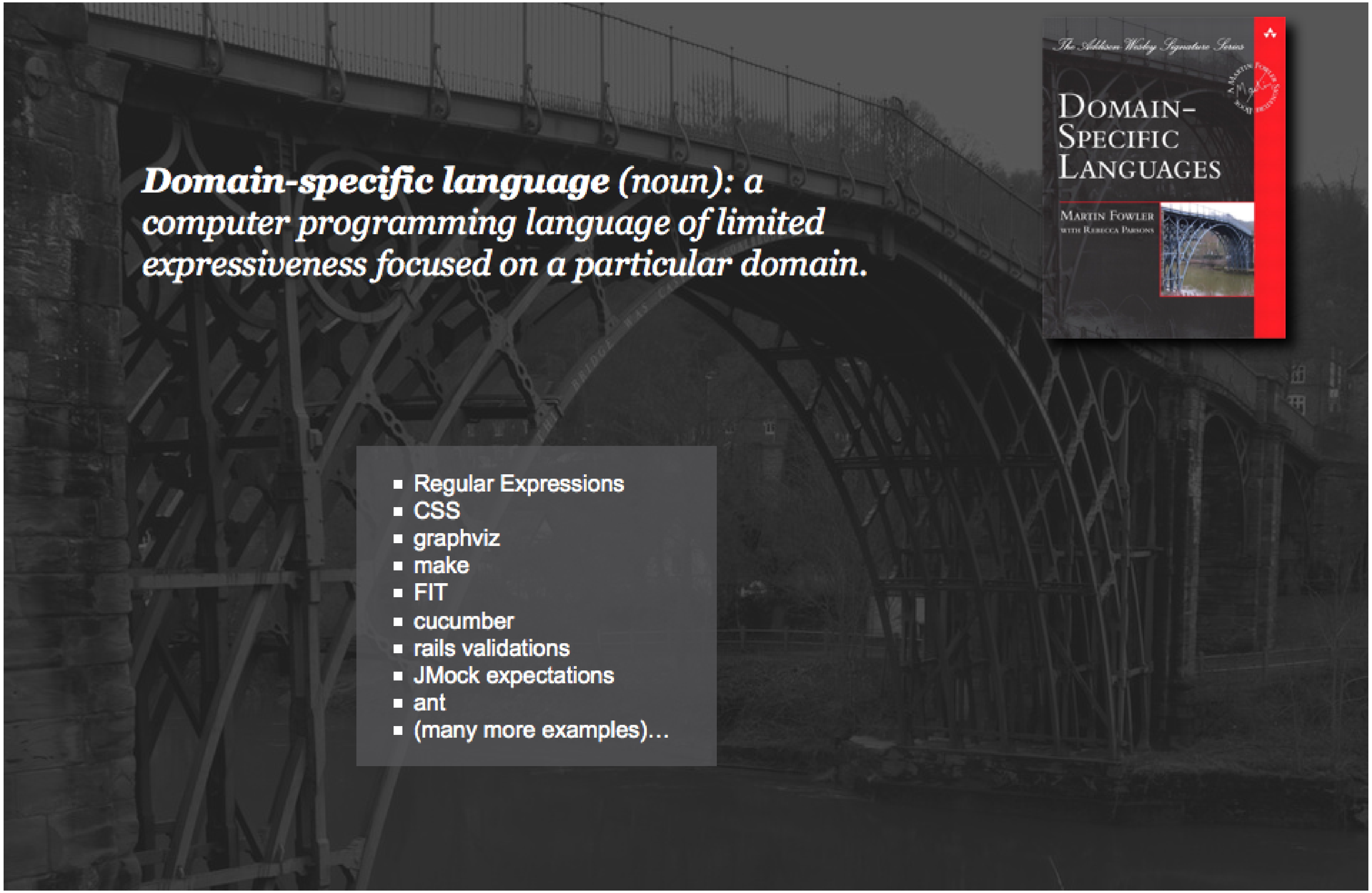
Quite rightly, scientists view programming as a means, not an end.

> Ignorance more frequently begets confidence than does knowledge *(Darwin, The Descent of Man)*

Also known as the *Dunning-Kruger effect*.

How to get dependable results without discouraging exploration of datasets?

2. Domain-specific languages



The Addison Wesley Signature Series
Domain-Specific Languages
Martin Fowler
with Rebecca Parsons

Domain-specific language *(noun): a computer programming language of limited expressiveness focused on a particular domain.*

- Regular Expressions
- CSS
- graphviz
- make
- FIT
- cucumber
- rails validations
- JMock expectations
- ant
- (many more examples)...

2.1. Abstraction

The essence of CS:

> convenient ways to express your thoughts precisely

Abstracting away from ‘irrelevant’ details.

A DSL is an abstraction of a particular *domain*,
supporting a domain specialist in building a *model*
(one presumably susceptible to subsequent ‘execution’).

2.2. Internal versus external DSLs

Sometimes, the ‘domain specialist’ is a software developer:

- focussing on a particular class of *program*
- often convenient to host DSL within a GPL
- ‘embedded DSL’

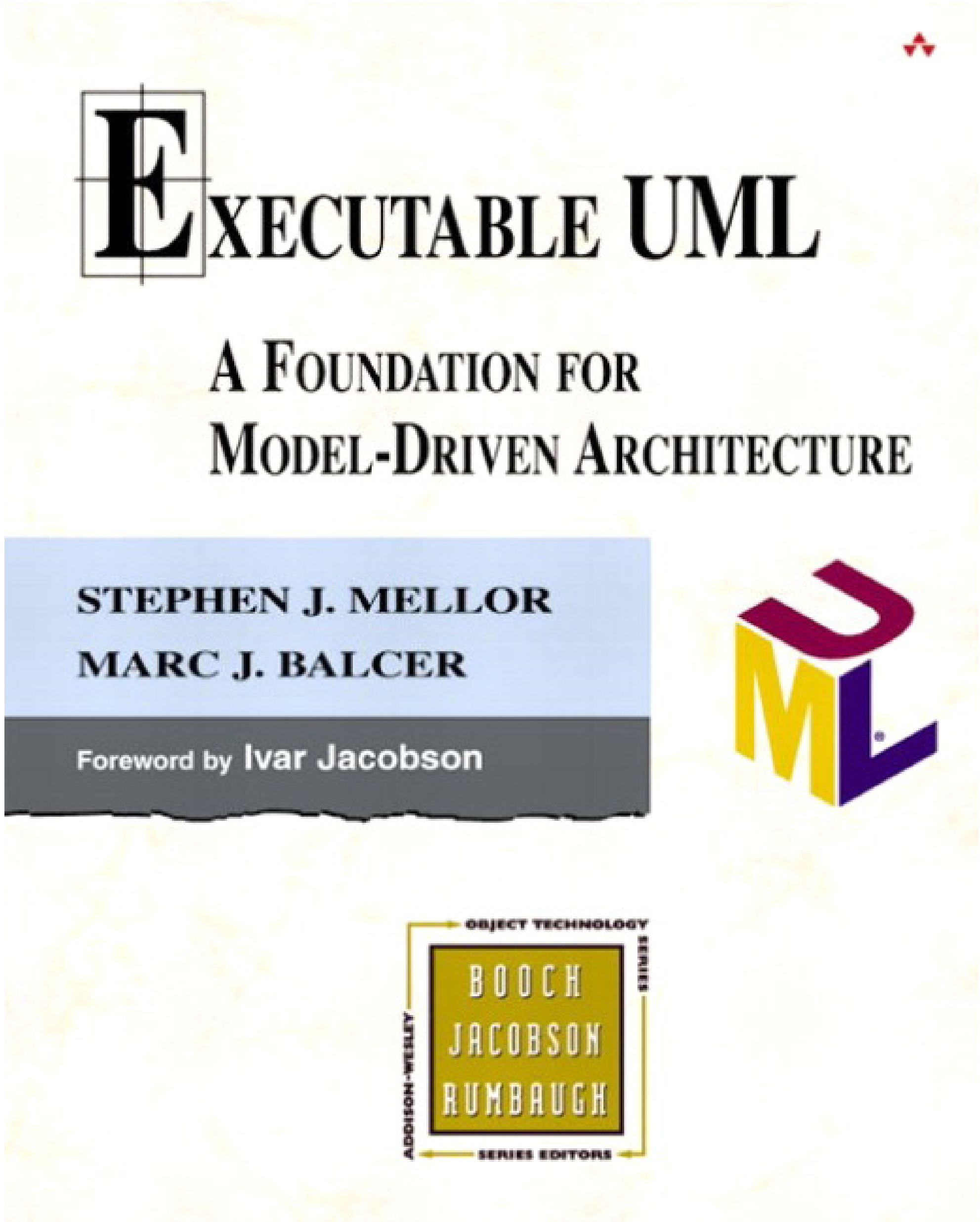
Sometimes their specialism is in another field altogether:

- focussing on modelling some *real-world phenomenon*
- don’t want to think about a PL at all
- non-textual notations, eg diagrammatic

2.3. Model-driven engineering

An alternative take on DSLs.

- platform-independent models
- platform-specific models
- translation or elaboration
- multi-purpose modelling



3. Semantic frameworks

At Oxford, we have been working on a series of projects developing what we call *semantic frameworks*:

- semantically rich domains
- heterogeneous collaborations (in time, space, field. . .)
- often low-budget
- transparency important

Initial work in clinical trials.
But very similar concerns in eg electronic governance.

Studiously trying to do *the simplest thing that could possibly work*.

3.1. CancerGrid

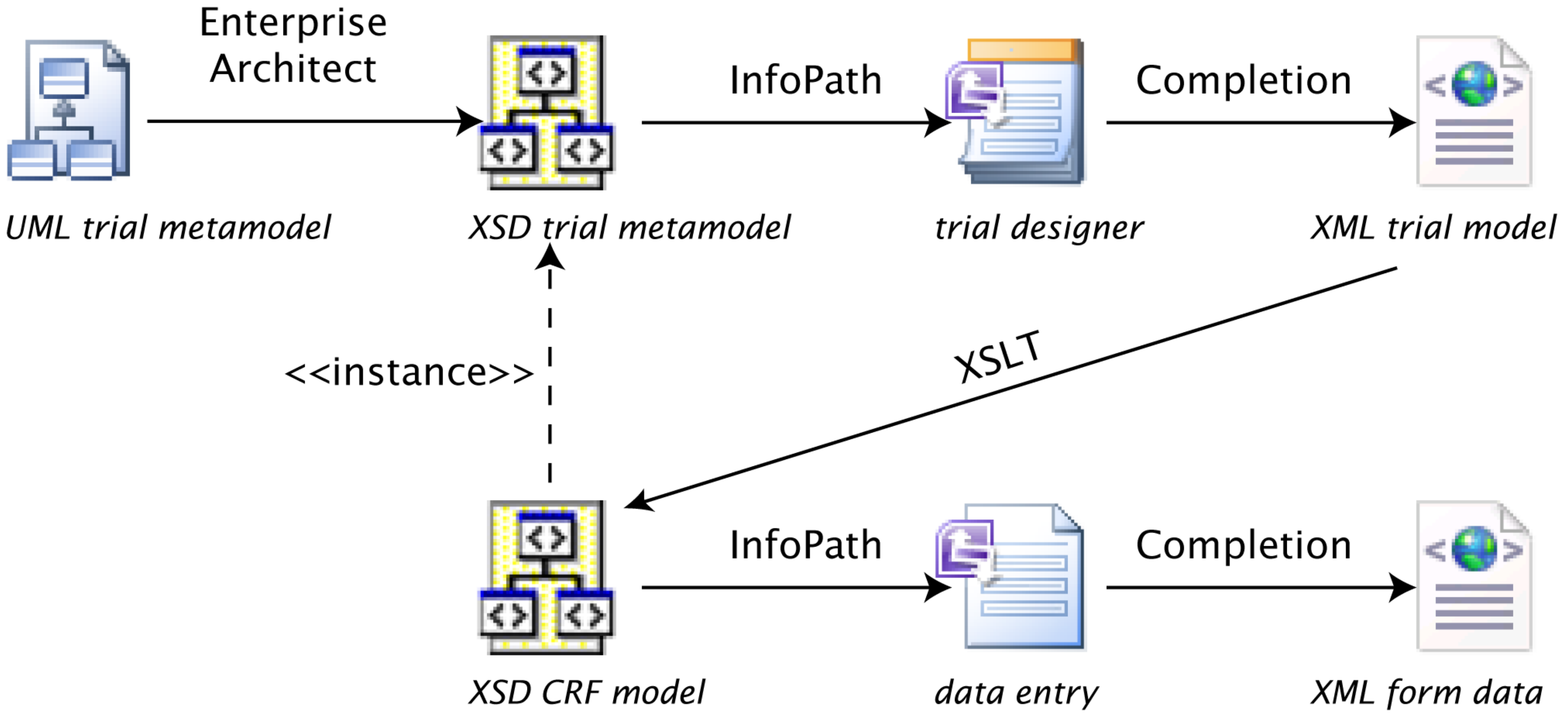
- UK Medical Research Council funded, 2005–2008
- supporting *randomized controlled trials*
- *metadata* to support data integration
- *models* to support tool generation

cancergrid

3.2. Metadata for meta-analysis

> “. . . the drug Tamoxifen—an oestrogen blocker that may prevent breast cancer cells growing—was the object of forty-two studies world-wide, of which only four or five had shown significant benefits. But this did not mean that Tamoxifen did not protect against breast cancer. When we put all the studies together it was blindingly obvious that it does. . . ”
> *(Richard Gray)*

3.3. Model-driven generation



3.4. Form follows function

It is the pervading law of all things organic and inorganic,
Of all things physical and metaphysical,
Of all things human and all things super-human,
Of all true manifestations of the head,
Of the heart, of the soul,
That the life is recognizable in its expression,
That form ever follows function. This is the law.

(Louis Sullivan, The Tall Office Building Artistically Considered)

3.5. Forms-based MDE

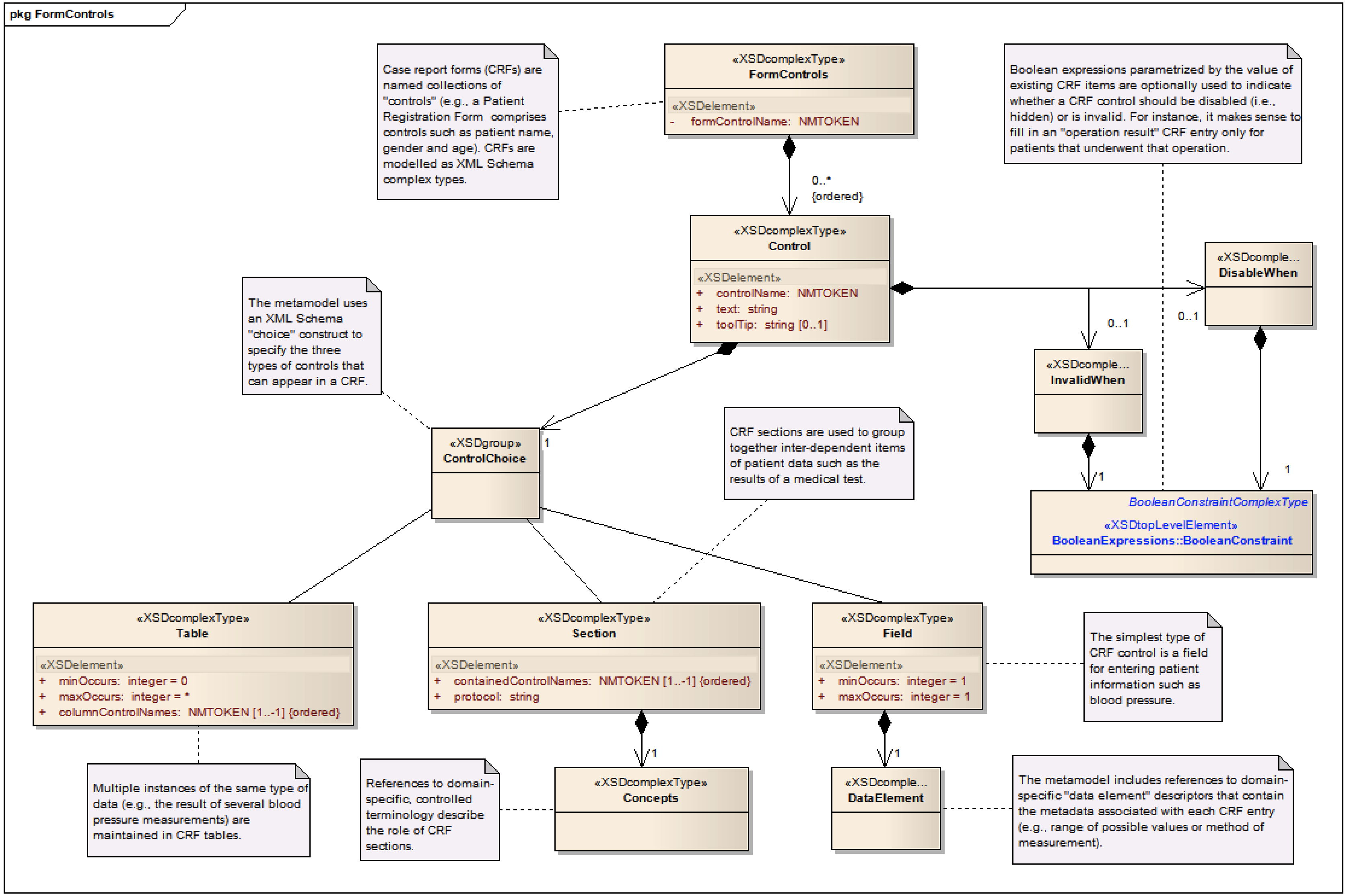
Off-the-shelf productivity software (eg Microsoft InfoPath and SharePoint) often suffices:

- *document schemas* as data models
- *conformant documents* as entities
- *form completion* as authoring
- *schema mappings* as model transformations

In some sense dual to Executable UML:

> Show me your flowchart and conceal your tables, and I shall continue to be mystified. Show me your tables, and I won't usually need your flowchart; it'll be obvious. *(Brooks)*

3.6. Domain metamodel



Case report forms (CRFs) are named collections of "controls" (e.g., a Patient Registration Form comprises controls such as patient name, gender and age). CRFs are modelled as XML Schema complex types.

Boolean expressions parametrized by the value of existing CRF items are optionally used to indicate whether a CRF control should be disabled (i.e., hidden) or is invalid. For instance, it makes sense to fill in an "operation result" CRF entry only for patients that underwent that operation.

The metamodel uses an XML Schema "choice" construct to specify the three types of controls that can appear in a CRF.

CRF sections are used to group together inter-dependent items of patient data such as the results of a medical test.

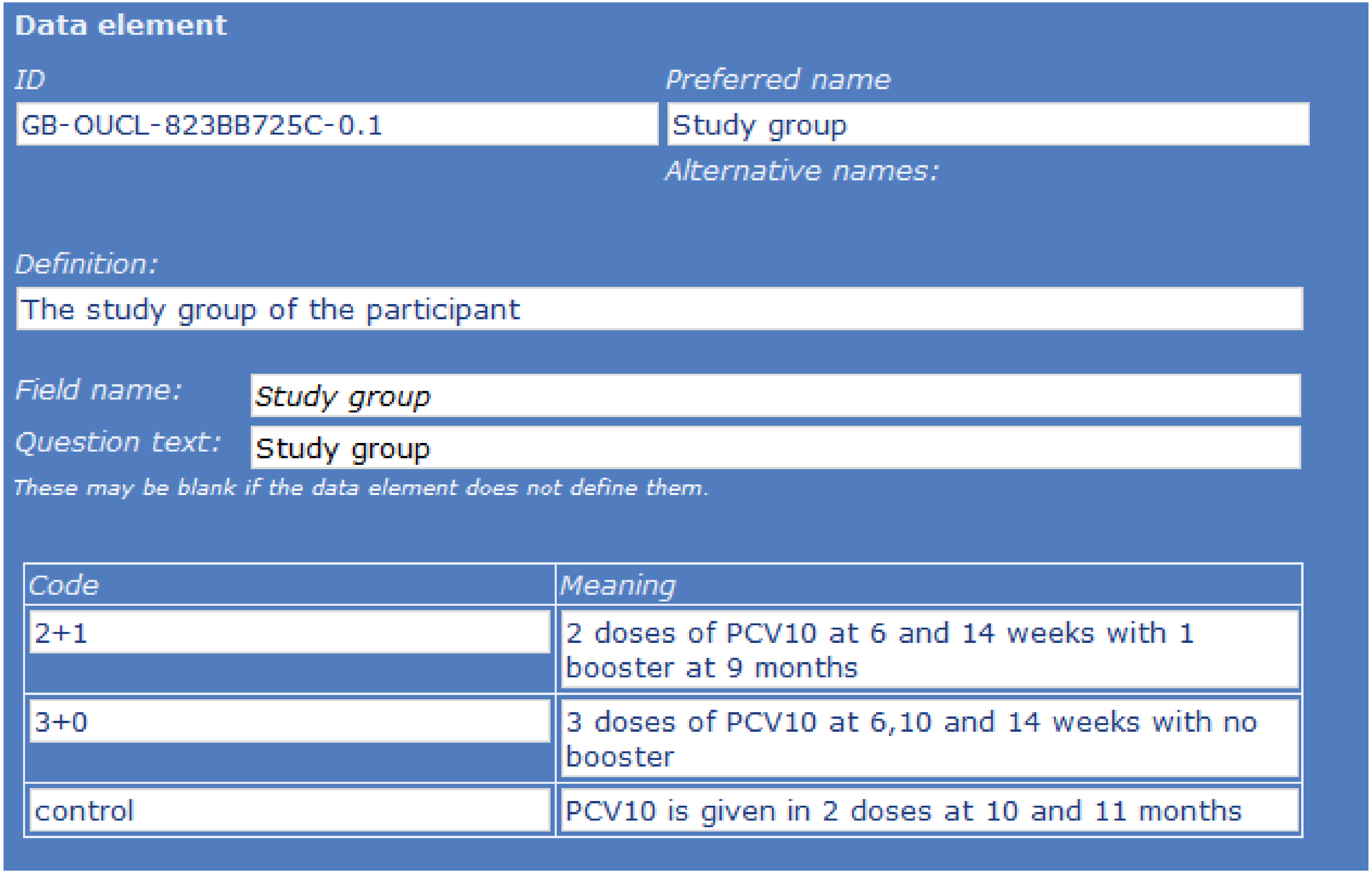
The simplest type of CRF control is a field for entering patient information such as blood pressure.

Multiple instances of the same type of data (e.g., the result of several blood pressure measurements) are maintained in CRF tables.

References to domain-specific, controlled terminology describe the role of CRF sections.

The metamodel includes references to domain-specific "data element" descriptors that contain the metadata associated with each CRF entry (e.g., range of possible values or method of measurement).

3.7. Experimental modelling: curating data elements



Data element

ID

GB-OUCL-823BB725C-0.1

Preferred name

Study group

Alternative names:

Definition:

The study group of the participant

Field name: *Study group*

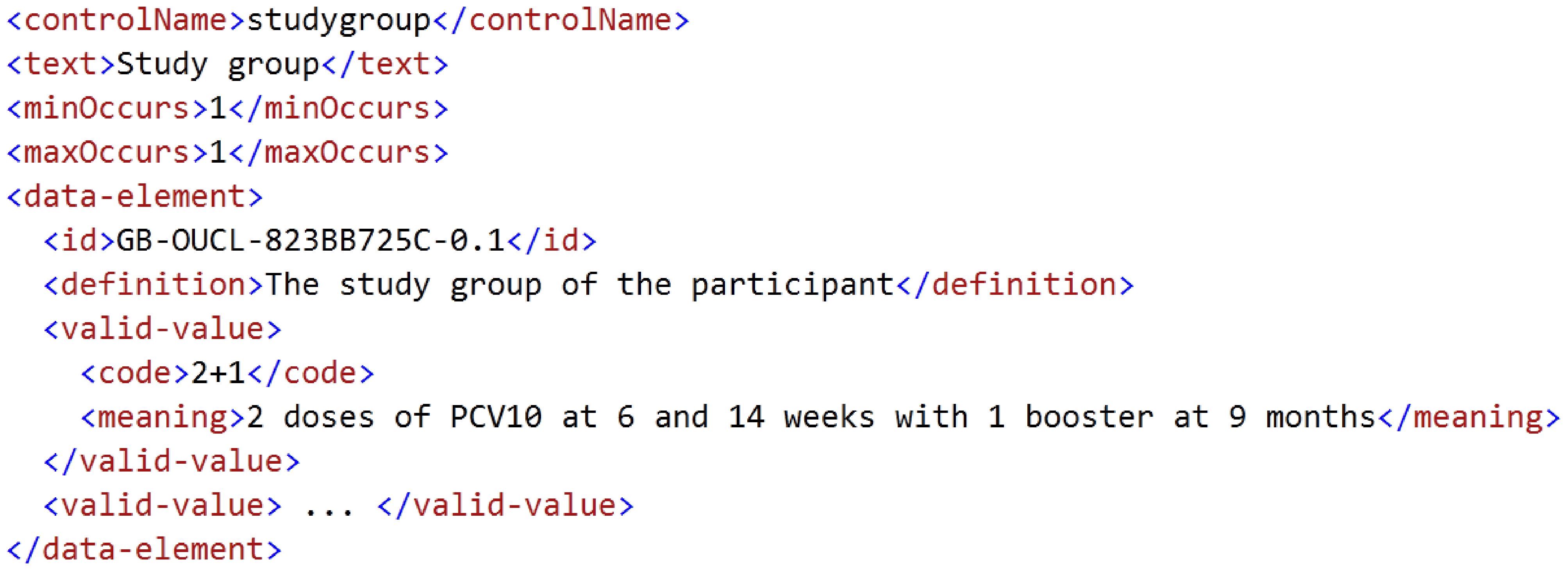
Question text: Study group

These may be blank if the data element does not define them.

Code	*Meaning*
2+1	2 doses of PCV10 at 6 and 14 weeks with 1 booster at 9 months
3+0	3 doses of PCV10 at 6,10 and 14 weeks with no booster
control	PCV10 is given in 2 doses at 10 and 11 months

3.8. The model

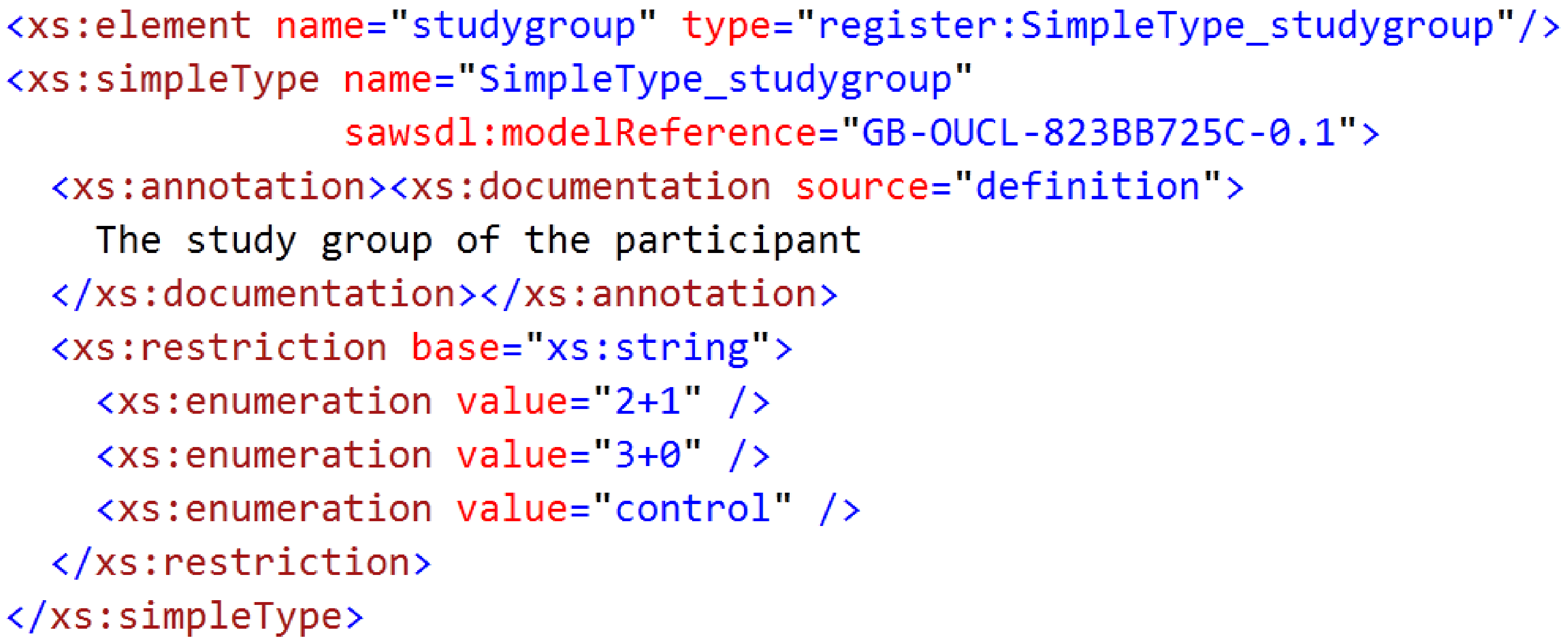
Experimental design (here, a trial protocol) specifies various trial-specific artifacts: clinical interventions, data elements, processes, service configurations, documentation. . .



```
<controlName>studygroup</controlName>
<text>Study group</text>
<minOccurs>1</minOccurs>
<maxOccurs>1</maxOccurs>
<data-element>
  <id>GB-OUCL-823BB725C-0.1</id>
  <definition>The study group of the participant</definition>
  <valid-value>
    <code>2+1</code>
    <meaning>2 doses of PCV10 at 6 and 14 weeks with 1 booster at 9 months</meaning>
  </valid-value>
  <valid-value> ... </valid-value>
</data-element>
```

3.9. Generation of customized software artifacts

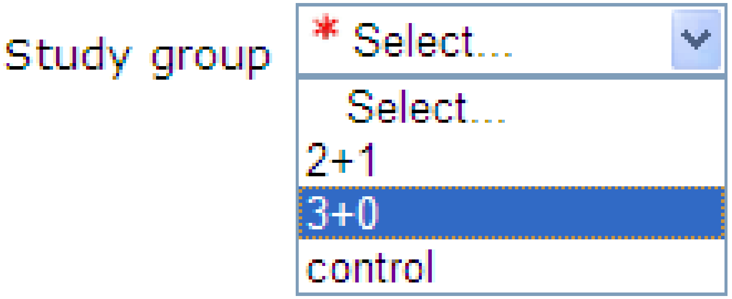
Traverse experimental model to extract and transform models of artifacts into artifacts themselves.



```
<xs:element name="studygroup" type="register:SimpleType_studygroup"/>
<xs:simpleType name="SimpleType_studygroup"
               sawsdl:modelReference="GB-OUCL-823BB725C-0.1">
  <xs:annotation><xs:documentation source="definition">
    The study group of the participant
  </xs:documentation></xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="2+1" />
    <xs:enumeration value="3+0" />
    <xs:enumeration value="control" />
  </xs:restriction>
</xs:simpleType>
```

3.10. Conduct the experiment

Clinician’s data entry is again form completion, but one level down.



Study group * Select...

This results in more structured data, which is stored for analysis.

3.11. Postmodernism

For any moderately complex system, we can't all agree on a single model; we shouldn't try to.

There is no one privileged view.

We need to allow for multiple models, and figure out how to make them *interoperable*.

3.12. Ongoing work

- *Accelerating Cancer Research Using Semantics-Driven Technology* (MSR)
- *Evolving Health Informatics* (RCUK)
- *HimalayaHelp* (Gates Foundation)
- *Hospital of the Future* (EPSRC)
- *Data Support Service* (MRC)
- *Union of Light-Ion Centres in Europe* (EU FP7)

4. Acknowledgements

. . . to the rest of the CancerGrid team:

- Jim Davies
- Radu Calinescu
- Charlie Crichton
- Steve Harris
- Andrew Tsui

. . . and to our funders: MRC, RCUK, EPSRC, MSR, EU.

5. Questions

- how important is robustness of modelling in your field?
- and how realistic is openness and transparency?
- sufficiently stylized to allow a semantic framework?
- dependent types? DSELs? cue Edwin. . .