Applying a Formal Ontology Approach in Government

Ian Bailey

www.modelfutures.com

IS/IT in Government

- Why are large IT programmes in Govt often seen to have failed ?
  - Govt sets standards for IT, project management, quality, accessibility.
  - Most Govt Depts have tightly controlled infrastructures and application portfolios
  - Govt Depts are no longer worried about paying to get the best consultants
  - In-house training in Govt Depts is easily as good as industry if not better

The Reality of Govt IT

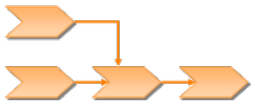
- Actually, the projects are run just the same as the ones in industry…but…
  - Much easier to get more funding for project overruns in industry – it's not tax-payers' money, so not subject to as much scrutiny
  - Easy to build-in contingency – it's much harder to persuade the tax-payer to part with an extra 25% on the assumption that the project *will* go pear-shaped
  - Easier to move the goal-posts – if a project doesn't deliver what is was supposed to, change the requirement
  - Greater readiness to take legal action against sloppy work
  - Greater readiness to allow on-the-fly innovation

Information Systems

- Moving the goalposts
  - As mentioned before this is reasonably common practice in industry…and perhaps a little in Govt projects too
  - But why does this have to happen ?
  - How do we get it so badly wrong every time ?
- Problem is usually associated with large-scale information systems
  - Often don't deliver the required functionality
  - Complaints from users that the old system was better
  - Byzantine work-around procedures required to conduct business-as-usual
- Is it a problem of requirement capture or implementation?

How was your System Built ?

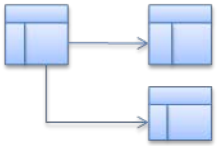
Stage 1:
Process Modelling



business analyst

Reliant on interview or observation, both of which are flawed approaches

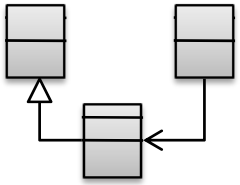
Stage 2 (opt 1):
Data Modelling



data modeller

Model produced depends on “school”, …and there are plenty of those

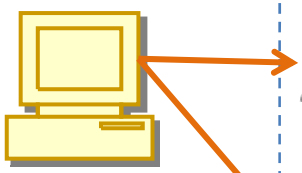
Stage 2 (opt 2):
Model-Driven Architecture



object modeller

Model is again dependent on school and methodology (e.g. RUP, GoF, etc.). Database implementation tends to be auto-generated

Stage 3:
Implementation



in-house IT

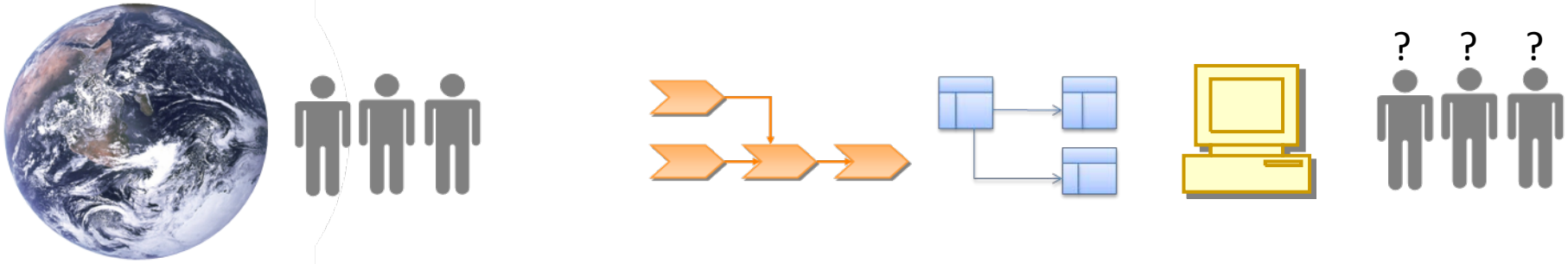
Tendency to “know better” and also dangerous tendency to “try out new technology”

“solutions” provider

Tendency to “leverage experience from other projects” and “make use of COTS” – i.e. re-badge the last thing they developed (BOHICA)

IT & The Real World

- By piling methodology upon methodology (few of which, if any, are defensible) we move our final implementation further and further away from the real world.



- The problems we're supposed to be solving exist in the real world.
- The users exist in the real world.
- *The funding exists in the real world*

- The IT lives in a parallel universe
- Very rare that two applications will implement the same information in the same way – leading to problems of interoperability

It Gets Worse…

- The methods used are rarely defensible or repeatable
  - Will two different business analysts come up with the same process model ?
  - Will two data modellers, *even given the same process model*, come up with the same data models ?
  - Will two architects, *given the same process model and data model*, come up with the same system design ?
  - Will two programmers, *given the same system design*, come up with the same application
- Semantic Overlap, Syntactic Disconnect
  - Lack of repeatability
  - Number of degrees of separation from the real world
  - Two applications which (at least partially) cover the same things are never going to be interoperable (unless they are designed together)

How to Make Systems Interoperable - 1

- Just use one system
  - The approach favoured by ERP consultants
  - Easy to sell to management (simple financial case)
- Adapt the business processes, not the software
  - Run an out of the box solution
  - Re-train staff to a new way of working
  - Arcane procedures often put in place for even the simplest things
  - Enormously demoralising effect
- But…
  - …reality is that it's never "out of the box"
  - …and it never scales to the whole enterprise
  - …so you end up with localised adaptations and tunings
  - …and you still have no interoperability

How to Make Systems Interoperable - 2

- Corporate Data Models
  - always, always, always, always fail
  - bit of a fashion in the 90s
  - fail for the reasons discussed before – they are inevitably someone's view of the world
  - …or designed by committee
- Data model Repositories
  - make it difficult to buy COTS
  - hard to account for every implementation route – RDBMS, XML, OODB, etc.
  - they can work for reference data though
- People hate being told what to do
  - Specialist communities have specialist terminology for good reason
  - Expecting them to work with someone else's view of the world isn't going to work

How to Make Systems Interoperable - 3

- Enterprise Application Integration
  - Most are point-to-point
  - Some are hub-and-spoke
  - Both need a lot of TLC
  - Work well in an tightly-controlled IT environment
  - Still problems of semantic integration though – someone has to decide when “Contact” means the same as “Address Record” or “Under Fire”
  - Often, the first cut is based on common terms

How to Make Systems Interoperable - 4

- Service-Oriented Architecture
  - Superb idea
  - Usually misinterpreted or mis-sold
  - Should be about componentising the business and the IT and creating an agile enterprise composed of interoperable and interchangeable services
  - …sadly the technology is leading the business requirement
  - Needs to be top-down designed – governance and reusability is key to SOA
  - Often just moves the interoperability problem onto the service bus

How to Make Systems Interoperable - 5

- Design them properly in the first place
  - Eh?
  - Surely not ?
  - But what would the consultants do for a living ?
- Use a consistent, defensible method to develop them
  - Heresy!
- One that ensures two different projects will come up with the same data structures to define the same things in the real world
  - Surely this is the stuff of fairy tales ?

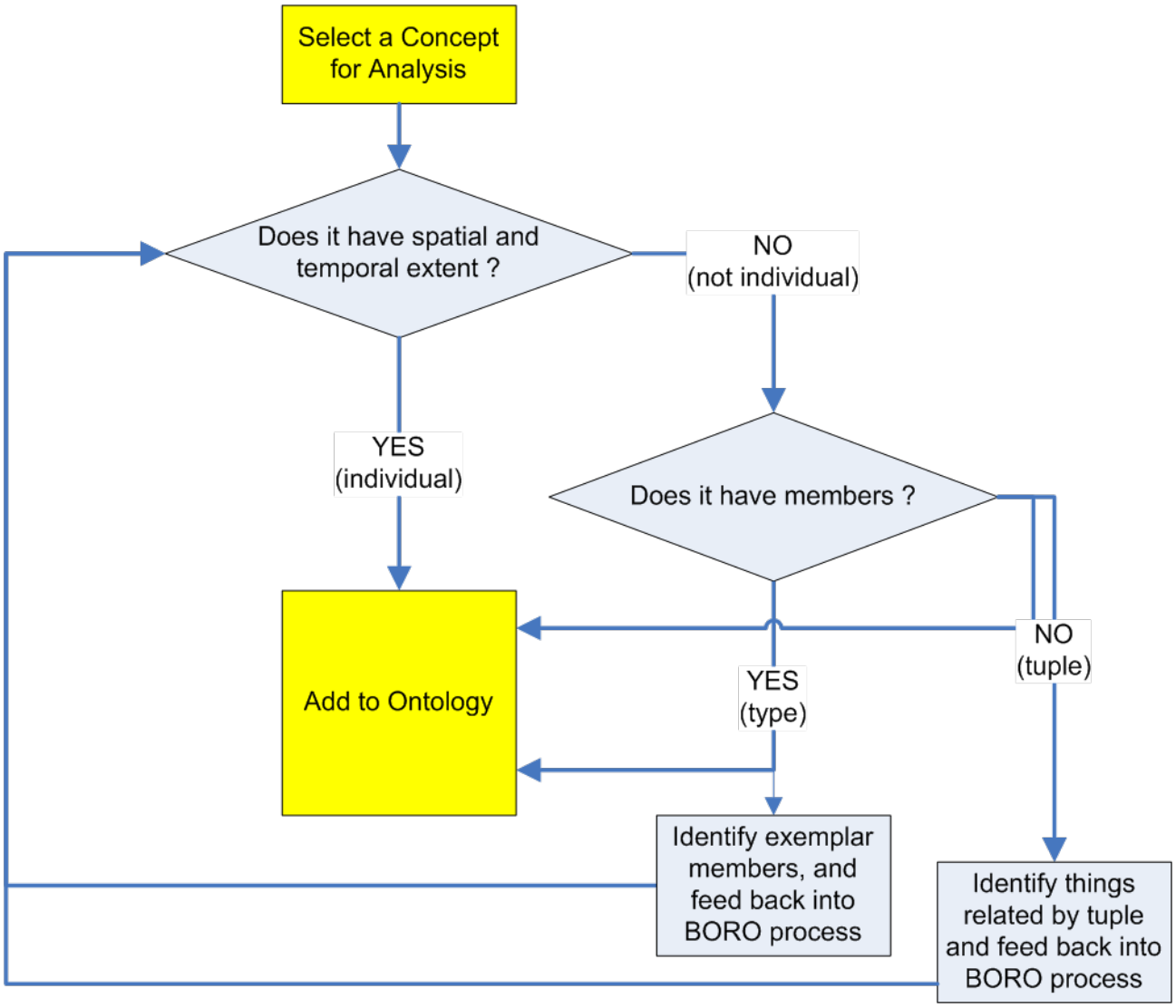
Forensic Approach

- Why did all those smart process and business modelling methods ignore the legacy data ?
  - Might have looked at the legacy data structure, but probably not the data itself
  - Nasty stuff legacy data – full of “data quality problems”
- What are those data quality problems ?
  - Often typos, sometimes inconsistent codification
  - Sometimes the user has put inappropriate data in a text field
  - But is it inappropriate ?
  - What if the application didn’t support their work particularly well, and they just needed to put this data “somewhere”
- The legacy data can tell you much more about what a business does than any process model or data model

The BORO Method™

- A simple (boring, even) method for re-engineering legacy data into a new model
- Process is grounded in:
  - Set theory
  - Physical reality
- Process is repeatable
  - Requires experience and discipline, but when followed to the letter, it always produces the same results
- Process is defensible
  - Not based on anyone's view of the world (this is an ontology in the purest sense)
  - Always traces back to things in the real world

BORO Flowchart



Where's it Used

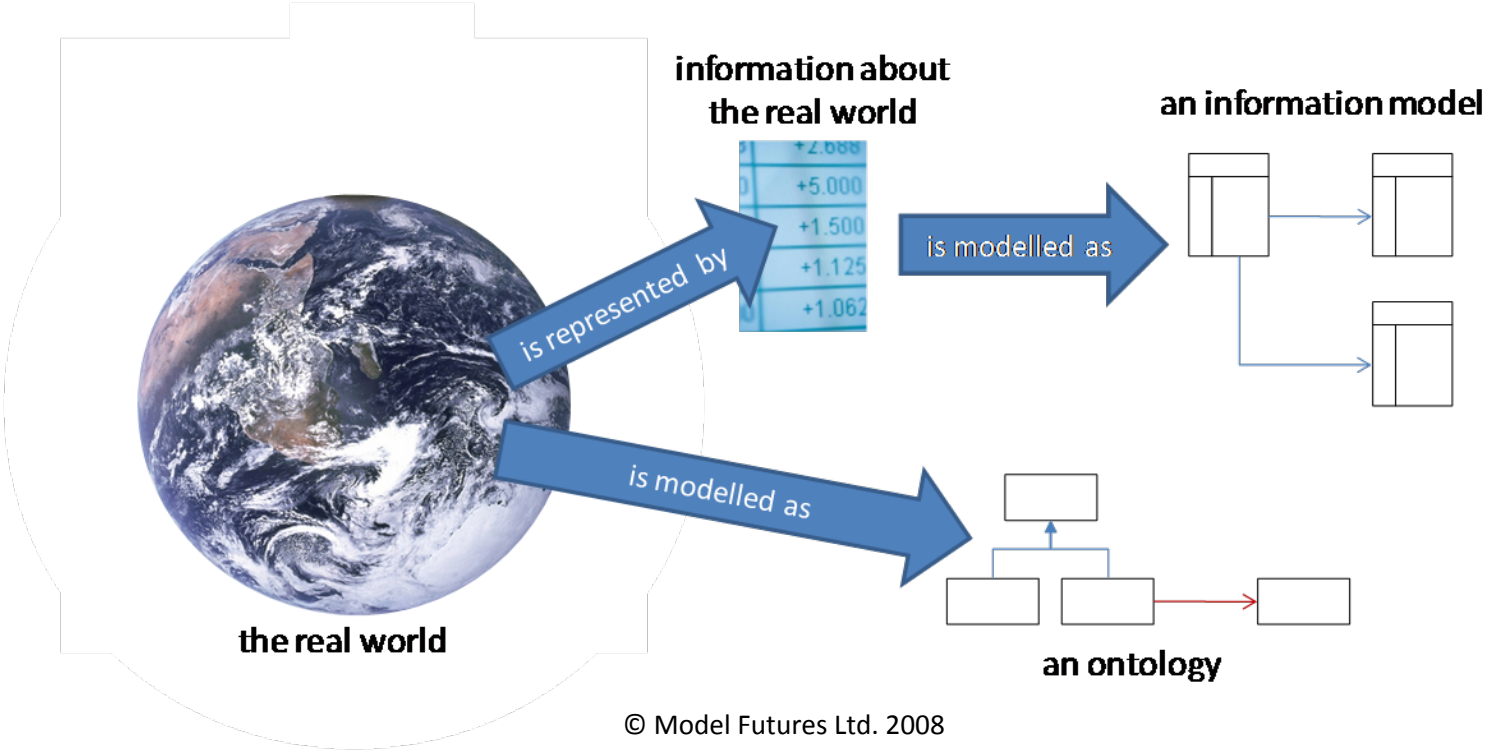
- Finance
- Oil & Gas
- MOD
  - MOD Ontology Demonstrator
- IDEAS
  - AUSCANUKUS + Sweden + NATO
  - Developing a common ontology for enterprise architecture

Ontology

- That was the title of the presentation, after all
- Caveat Emptor
  - Just using a “semantic web” approach will not give you an ontology
  - OWL is just a syntax
  - Without the necessary philosophical and mathematical rigour you get the same problems as in traditional information systems
  - Most ontologies you will encounter are in fact just old data models, re-rendered in OWL
- IT definition of semantic
  - The structure behind the syntax
- Mathematical/philosophical understanding is different
  - The things in the real world that are being referred to
- IT semantics are still one step removed from the real world
  - Need tighter coupling to reality

Getting Closer to the Real World

- BORO develops an ontology that is tightly coupled to the real world
- A data/information model is two steps removed
  - This is very hard to get across to information management professionals
  - Can take months before the penny drops



Last Slide

- Problem with information systems is their disconnect from the real world
  - Arbitrary model development is to blame
  - And there's no reason why this problem should magically go away just because you're using semantic web technology
- Legacy data is the key
  - it's not nice and shiny and new, but it makes a better witness than a business analyst
- Use a method that is repeatable and defensible
  - BORO seems to work
  - Needs more extensive testing
  - Other methods may be out there too