

Consistency Management

Anthony Finkelstein

University College London
Department of Computer Science

a.finkelstein@cs.ucl.ac.uk
<http://www.cs.ucl.ac.uk/staff/A.Finkelstein>



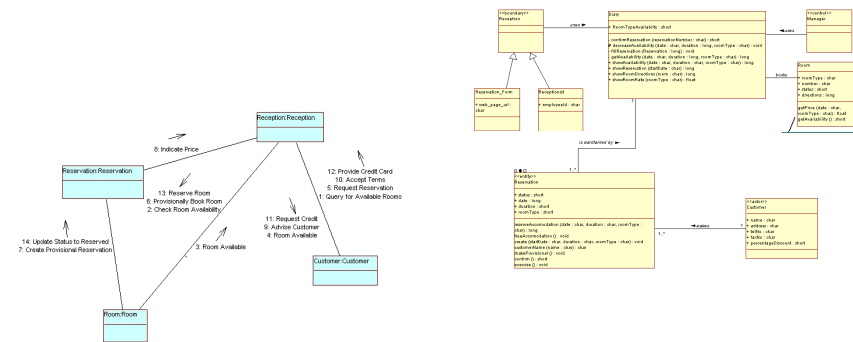
Our Setting



- Distributed software development
 - Distributed developers
 - Highly collaborative work
 - Large scale information management requirements
 - Use of many formal and semi-formal languages with more or less precise interpretations in the domain of computation
- Disagreements, deferral of commitment and wide range of sensitivity to inconsistency are particular features

A Simple Example

- An instance in a collaboration diagram must be an instance of a class in the class diagram



Approach

- It makes sense to view these as problems of “consistency”
 - let me explain ...

and why should this interest me?

Everyday Inconsistency

- Saying something in one place and another contradictory thing in another place
- The equivalent of the standard logical notion of inconsistency
 - In which we assert both that **the sky is blue** and **the sky is not blue**
- Of course in the “real world” this is NOT a problem ...



Action in the Presence of Inconsistency

- ... unless we try and base some actions upon the inconsistent information
 - Selecting a coat for example
- In this case we might expect the actions to **interfere**
 - Thus both wearing a raincoat and not wearing a raincoat
- This interference may not be immediate and, of course, depends on the veracity of the information



Why Inconsistency Occurs

- Inconsistency has many causes
 - Collaboration of multiple actors, each with different opinions, views and interpretations on the real world
 - Uncertainty leading to the preservation of an equivocal and hence inconsistent position
 - Errors
 - Deliberate falsification

A Foolish Consistency?

"A foolish consistency is the hobgoblin of little minds adored by little statesmen and philosophers and divines. With consistency a great soul has simply nothing to do ... speak what you think today in words as hard as cannonballs and tomorrow speak what tomorrow thinks in hard words again though it contradict everything you said today"

- R.W. Emerson



Seeking Inconsistency

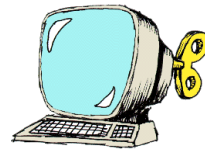
- There are many circumstances in which inconsistency is acceptable
 - indeed desirable
- Particularly in any situation where you are seeking to establish what to do prior to committing to a course of action
 - Where you want all the alternatives laid before you and as full and accurate information about the state of the world as it is possible to obtain

Managing Consistency

- Rather than thinking about removing inconsistency we need to think about “managing consistency”
- This means
 - preserving inconsistency where it is desirable to do so
 - identifying inconsistency at the point where decisions are required
 - removing (or otherwise remedying) inconsistency prior to taking action
- This requires a major change in the way we think.

Old Approach

- Classically our concern has been
 - to organise information management practices so as to prevent inconsistency from arising
 - to use database mechanisms to remove inconsistency as close to the source as possible
- Where inconsistency occurs
 - despite our best attempts to eliminate it!it is viewed as an “application level” concern to be handled on a case by case basis.



The State of the Art

- Shared file systems (with limited merging)
- Databases (with view mechanisms)
 - Canonical representations
 - Hard-coded checks
- Selective broadcasting



New Environment

- In a closed and relatively static environment this approach might be acceptable
- In the our setting of distributed software development
 - federated organisations
 - web publication
 - distributed collaborative workit patently does not work!
- We would like flexible and open consistency management with strategies, policies and tools defined at the generic level

Why is Consistency Management Difficult?

- In a trivial case such as the sky is blue, the sky is not blue the inconsistency is easy to spot (if not to deal with) by contrast inconsistencies in real settings present much more challenging problems
- The assertions can be vague or semantically ungrounded so that it is difficult to determine their precise meaning. It may be impossible to determine how the assertions relate to the real world and hence establish whether actions based on them will interfere.
 - Assertions may be made in different (formal) languages whose relationship to each other is uncertain

Why is Consistency Management Difficult?

- Inconsistencies can be **hidden** in a mass of other assertions
 - Assertions can be physically distributed in such a way that it is difficult to assemble the information so as to detect the inconsistency
- An inconsistency can itself be **distributed** across many assertions
 - There may be many inconsistencies, some related and some independent, in a set of assertions
 - Each inconsistency can be of varying importance in the domain.

Why is Consistency Management Difficult?

- The presence of inconsistency may **pollute** the set of assertions
 - And hence prevent the use of certain technical mechanisms for reasoning about the information
- The information, which itself may be **changing** rapidly, may be intertwined with a complex workflow
 - So that it is not easy to determine the points at which it is critical to establish consistency

Why is Consistency Management Difficult?

- In many cases inconsistencies reflect slips and minor errors or possibly delayed commitments which are relatively easy to **resolve**
 - Some inconsistencies however reflect serious conflicts with substantial knock-on consequences and may involve substantial negotiation



What Needs to be Done

- Ideally what we would like to build a toolset which should include:
 - tools to help establish, express and reason about the **relationships** between formal languages
 - tools to **check consistency** with respect to these relationships and to provide diagnostic feedback
 - where inconsistencies have been detected, tools to **visualise** the inconsistencies
 - tools to **track** inconsistencies and preserve diagnostic information in the face of on-going change



What Needs to be Done

- tools to **support resolution** either through the removal of inconsistency – rectification – or by reducing the scope or severity of the inconsistency – amelioration
- tools to **support the rationale** associated with consistency management
- We also need to be able to:
 - **specify policies** with respect to when consistency should be checked and when resolution mechanisms should be applied
 - enforce these policies at appropriate times and in an appropriate manner.

Technical Challenges

- ④ An in-depth understanding of the semantics of formal languages used in particular application domains
- ④ Construction of meta-models with a sound foundation and the ability to reason across different formal languages expressed in terms of those meta-models
- ④ Expression of constraints on the meta-model as consistency checks

Technical Challenges

- ➊ Ability to check consistency in a way that is fast and highly scaleable
 - We can anticipate that we may have to check the consistency of very large numbers of highly complex distributed documents and other information artefacts
- ➋ Rendering consistency checking unobtrusive and low cost. Devising algorithms that are computationally efficient in terms of space and performance
- ➌ Ensuring useful diagnostic feedback
 - At least localisation

Technical Challenges

- ➊ Devising visualisation techniques appropriate to showing consistency across complex information 'spaces'
 - Supporting consistency-based navigation through the information space
- ➋ Developing version and configuration control strategies that allow inconsistencies to be tracked as surrounding information changes

Technical Challenges

- ➊ Supporting conflict resolution
- ➋ Associating meta-data with inconsistencies
 - Diagnostic
 - Configuration
 - Rationale
- ➌ Developing schemes for specifying what checks to perform at what point in the workflow and what actions should be taken by way of resolution

A Quick Demo

To show where we got to ...

Some Interesting Open Issues

- Workflow integration
- 'Natural' Rule Languages
- Ontology integration
- Repair

