

Design Rationale

4750

Summer, 2002

Design Rationale

- Design rationale is information that explains why a system is the way it is
- Prospective use
 - Structure the decision-making process during design
- Retrospective use
 - Understanding earlier decisions to help decide how to improve (maintenance & enhancement)

Benefits of Design Rationale

- communication throughout life cycle
- reuse of design knowledge across products
- enforces design discipline
- presents arguments for design trade-offs
- organizes potentially large design space
- capturing contextual information

Issue tracking

- Design and planning answer questions
 - “Is” questions
 - “What do you do in the library?”
 - “What exactly is a patron ‘in good standing’”?
 - “Ought” questions
 - “How do you want this information to be communicated?”
 - “Is this feature more important to you than that?”

Issues and actions in meeting minutes

- Keeping track of actions is standard
- Can also keep track of open issues
 - Some of these are actions, if one person has responsibility to resolve
 - But others are more open-ended & just need to be remembered

For example...

It was decided to base the meeting scheduler on a shared calendar.

Action: Frank to sketch 1st-cut schema

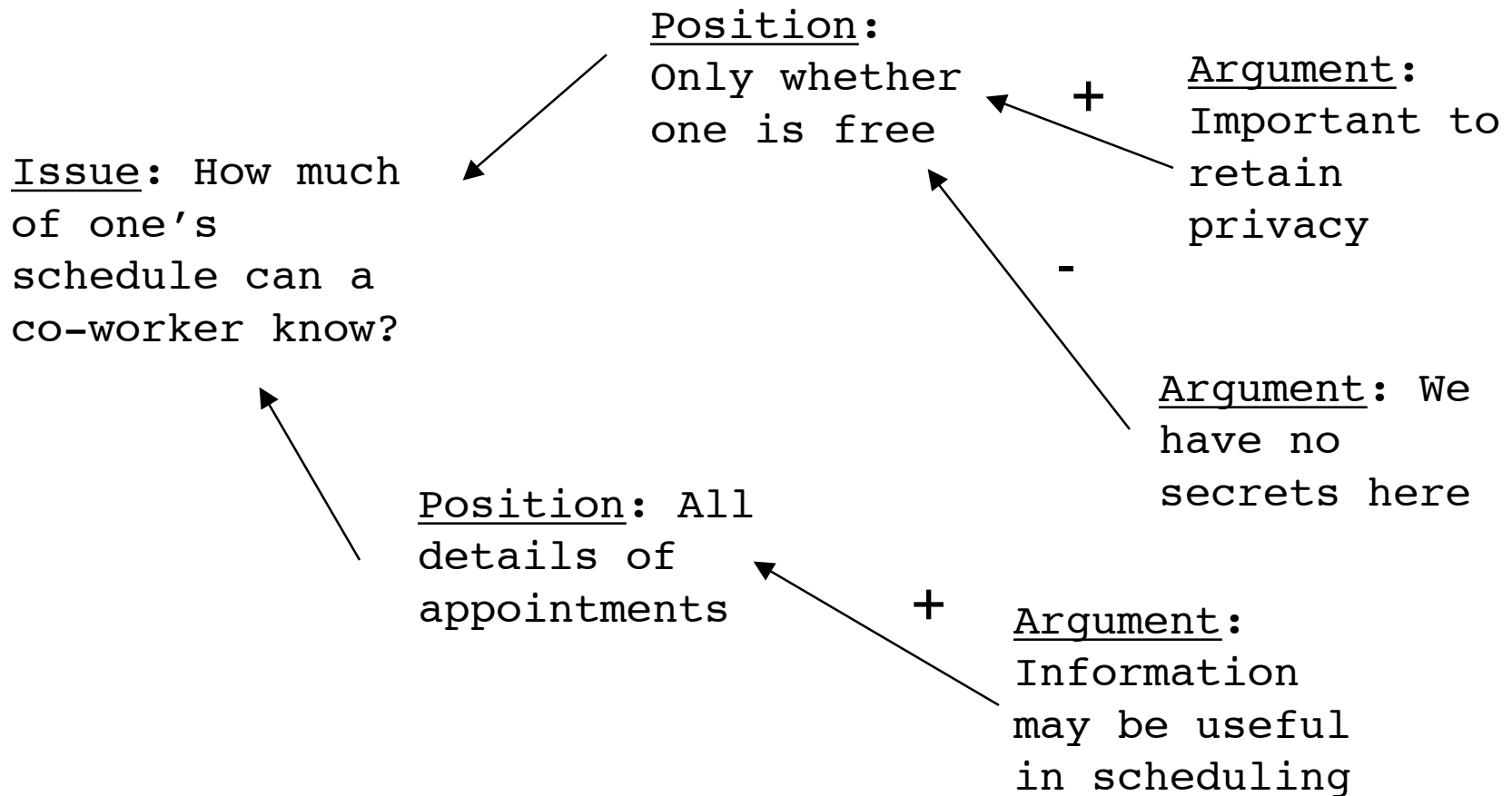
Issue: How much of one's schedule can a co-worker know?

IBIS: (Issue-based information systems)

For prospective community use

- Structured method for keeping track of issues, positions and arguments
 - Originated in architecture & urban planning
- Issue
 - An open question about which there are likely to be opposing points of view
- Position
 - A response to an issue by a stakeholder
- Argument
 - A reason put forward by a stakeholder for choosing or rejecting a position

Example IBIS graph



The Inquiry Cycle

(Potts, Anton & Takahashi, 1994)

- Insight: Questions are always prompted by something
 - Attach questions to the document fragments that gave rise to them

```
1.1 The system shall
blah, blah...
1.2 If the co-worker
is blah, blah, the
system shall inform
the user ...
1.2.1 Blah, blah,
blah...
...
```

Question: How much
of one's schedule
can a co-worker
know?

This question
annotates a
specific reqt.

The Inquiry Cycle

Integration of prospective/retrospective uses

- Answering the question should lead to a revision of the artifact

1.1 The system shall
blah, blah...
1.2 If the co-worker
is blah, blah, the
system shall inform
the user only that
the co-worker is
busy...
1.2.1 Blah, blah,
blah...
...

Question: How much
of one's schedule
can a co-worker
know?

Decision: Only
whether one is
free

This decision record
represents the rationale
for the new reqt.

Claims Analysis (Carroll)

- To support task-artifact cycle in which user tasks are affected by the systems they use
 - aims to make explicit consequences of design for users
 - designers identify tasks system will support
 - scenarios are suggested to test task
 - users are observed using system
 - psychological claims of system made explicit
- negative aspects of design can be used to improve next iteration of design

Design Space Analysis

(Mainly prospective, but for detailed design)

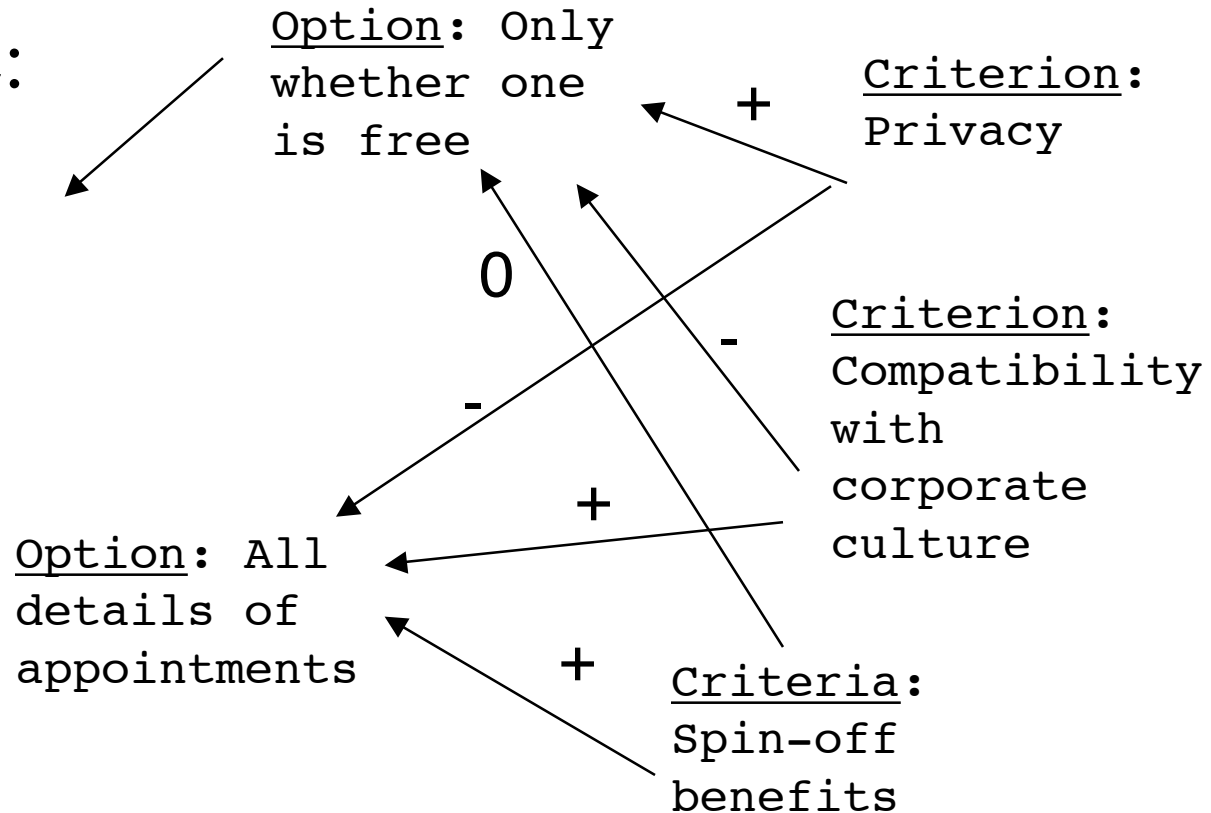
- structure-oriented
- QOC (Maclean et al) – hierarchical structure
 - questions (and sub-questions) represent major issues of a design
 - options provide alternative solutions to the question
 - criteria are the means of assessing the various options in order to make a choice
- DRL – similar to QOC with a larger language and more formal semantics

QOC: Design space analysis (really MCDA in disguise)

- Similar to IBIS, but uses explicit criteria

- Compare:

Question: How much of one's schedule can a co-worker know?



Experiences with issue tracking

- The facts:
 - NCR used IBIS for restaurant IS design
 - NTT has used Inquiry Cycle for CSCW technology development
 - Design Space Analysis has been used to record design rationale for interactive systems at Xerox
 - This is old research (late '80s-mid-'90s)
- But, really works for informal decision support.
 - Detailed support for decision *making* is overkill
 - Detailed documentation of *after-the-fact* rationale requires obsessive documentation culture
- Is there a good lightweight prospective method that provides retrospective summary documentation?....

PMI (+-!) Analysis

(Mainly for retrospective summary)

Option	Plus (+)	Minus (-)	Interesting (!)
Use a PDA	+ Portability + Familiarity	- Form factor too small	! Design as sequence of inputs, one per screen
Use a website	+ Familiarity + Expense		! Applet(?) accessible through browser
Use a kiosk	+ Control over design features	-Special-purpose device at fixed location - Expense	