

Modular Requirements Against Feature Interaction Problems

Jan Brederke

University of Bremen

“Naive” Feature-Oriented Description in Telephone Switching

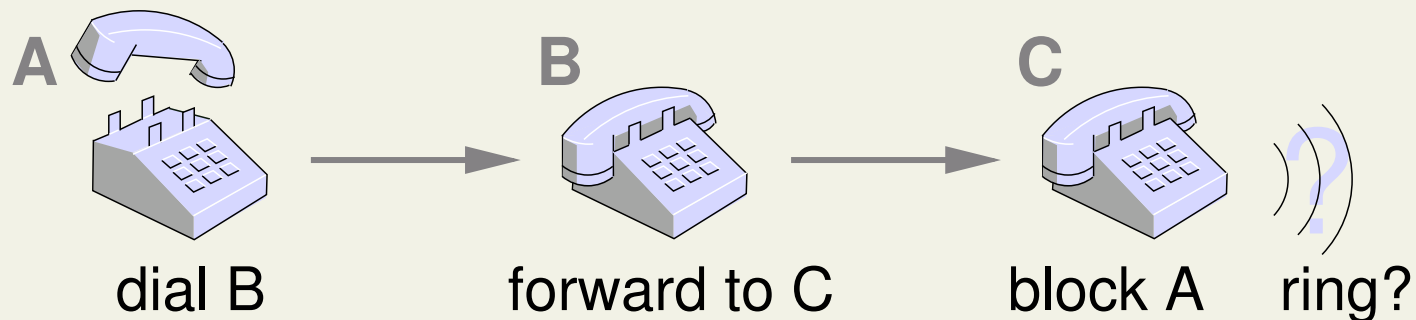
- base description plus separate feature descriptions
- attraction: behavioural “modularity”
 - easy change of system behaviour
 - make *any* change by just adding a new feature description
 - never change existing descriptions
- emphasizes individual features
 - makes them explicit
- de-emphasizes feature interactions
 - makes them implicit in the feature composition operator

Feature Interaction Problems in Telephone Switching

- features work separately, but not together
 - hundreds of (proprietary) features
 - combinations cannot be checked anymore
- telephone switching
 - users' expectation high

Feature Interaction Example: Call Forwarding & Terminating Call Screening

- CF
 - B forwards all calls to C
- TCS
 - when A is caller, C blocks him
- A calls B: can/should A reach C?



- notion of “caller” is crucial

My Approach: Families of Rigorous Requirements

- maintain all variants together
- reduce dependencies:
information-hiding requirements modules
- document remaining dependencies

Requirements Module

- Definition:
 - a set of properties that is likely to change together
 - likeliness of change: abstractness
 - abstractness: part of how many family members?
 - correlation of changes?
 - “cohesion”
- groups properties
- groups property groups, recursively:
 - hierarchy of modules
 - handle many “small” properties

Writing Requirements Modules in the Formalism Z

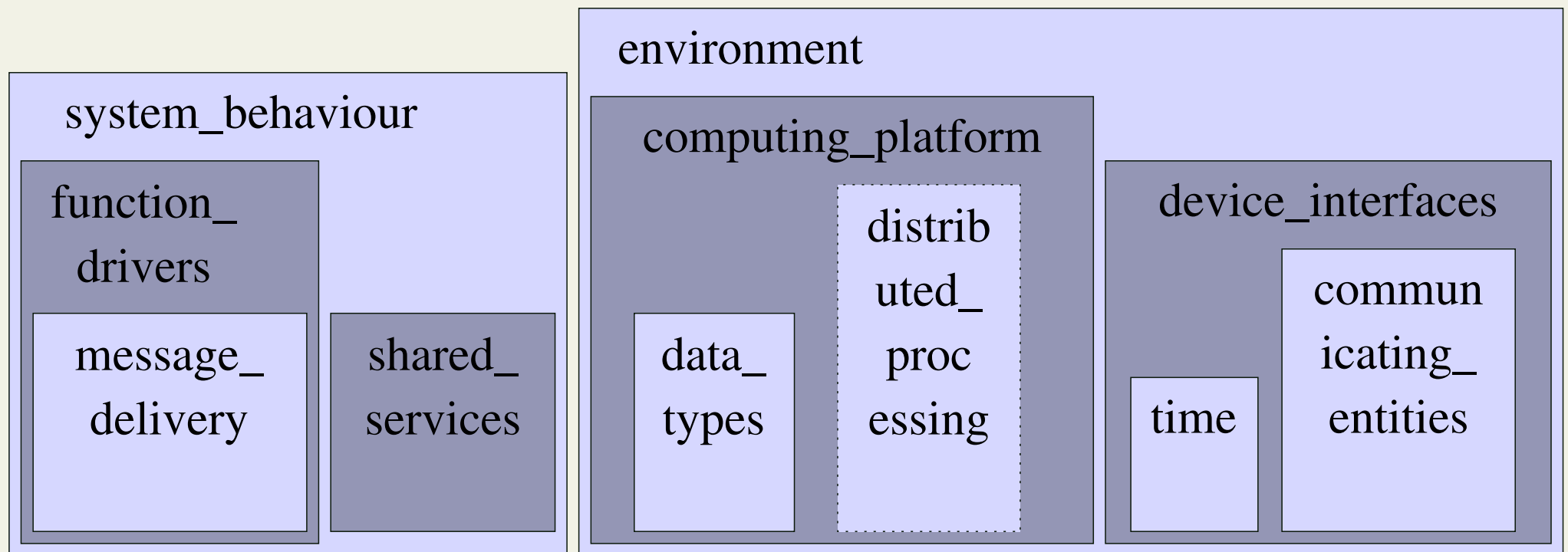
- formal grouping construct: “section” (in ISO-Z)
 - + groups all kinds of formal paragraphs
 - + dependencies explicit, via “parents” construct
- caveats:
 - not recursive
 - no interfaces
 - ▷ no access control for changing parts

A Small Extension for Z

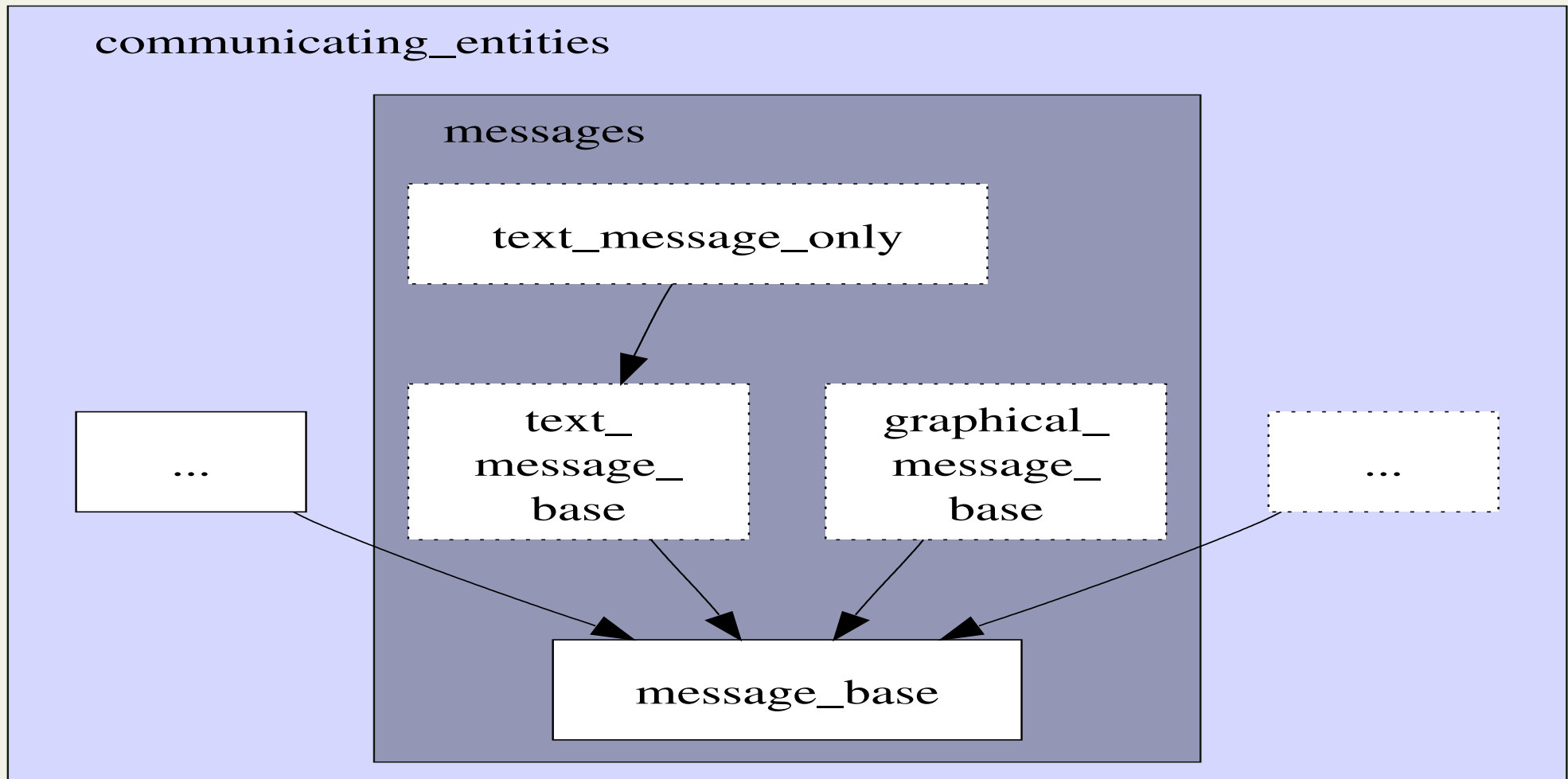
- **“chapter”**: formal grouping of sections, and chapters
 - recursive
- **interface** for sections and chapters
 - restricts access to changing sections/chapters

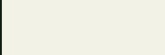
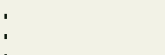

Example for Grouping: a Family of LAN Message Services

top-level requirements module architecture:

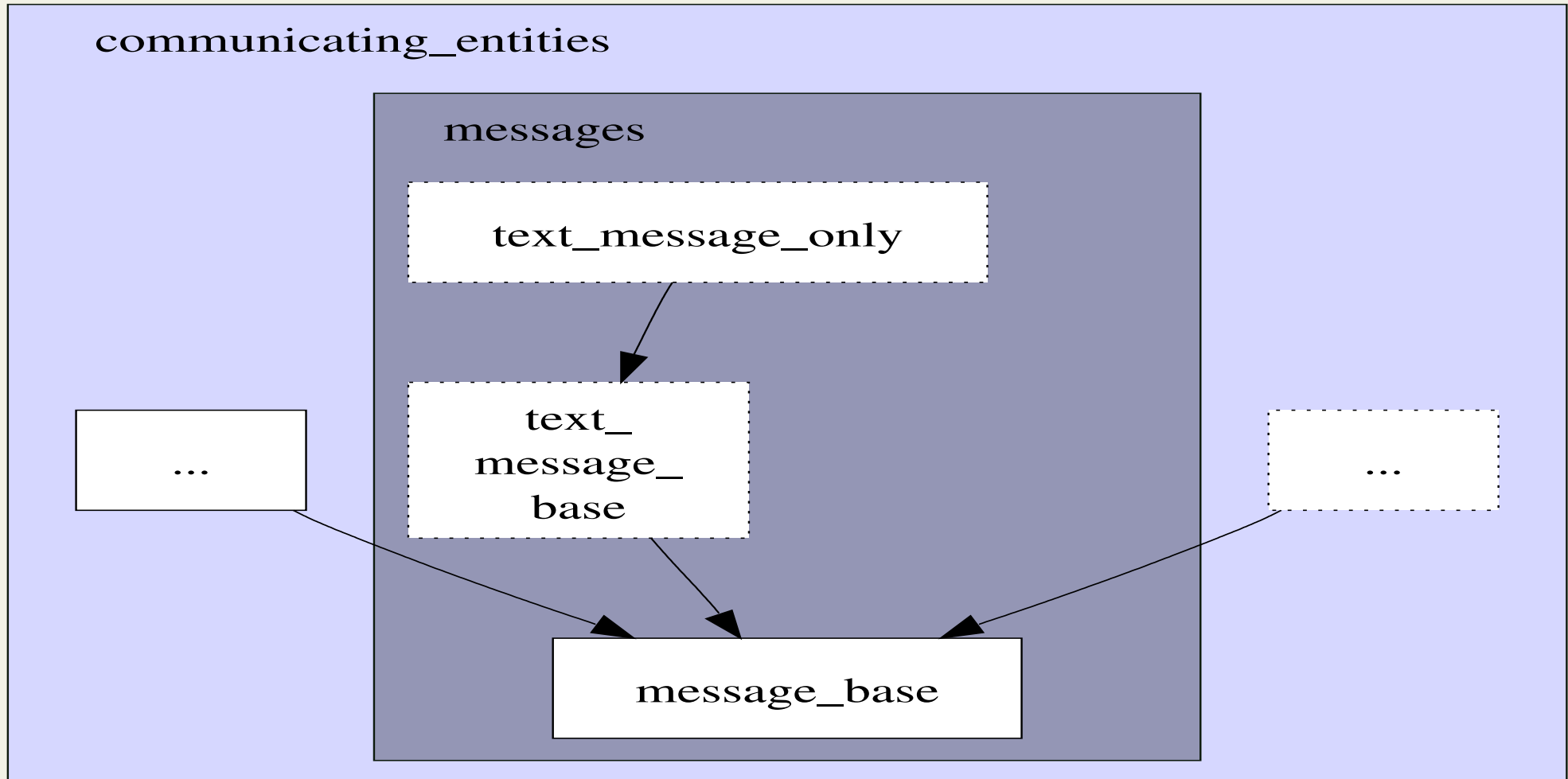


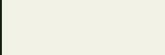
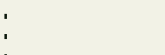
Example: Interfaces Restrict Access



legend:  public  private  dependency

Example: Generating One Family Member



legend:  public  private  dependency

References

- [Bre04] Brederke, J. *On feature orientation and on requirements encapsulation using families of requirements*. In Ryan, M. D., Meyer, J.-J. C., and Ehrich, H.-D., editors, “Objects, Agents, and Features”, vol. 2975 of “LNCS”, pp. 26–44. Springer (2004).
- [Bre05] Brederke, J. *Maintaining Families of Rigorous Requirements for Embedded Software Systems*. Habilitation thesis, University of Bremen, Germany (2005). *To appear*.

Reserve Slides

FI Example: Calling Card & Voice Mail

- #-button
 - (Bell) calling card:
start new call without re-authorization
 - (Meridian) voice mail:
end of mailbox number, end of password, . . .
- call voice mailbox using calling card??
 - either early disconnect, or
 - calling card feature crippled
- resolution by Bell
 - introduce new signal:
“#-button pressed at least 2 sec.”

FI Example: Call Waiting & Call Forward on Busy

- both activated simultaneously
 - in busy state
 - when another call arrives
- only one can get control
 - no resolution, except restrictions on features

FI Example: Originating Call Screening & Area Number Calling

- OCS
 - aborts calls to numbers in list
 - query Service Data Point (SDP) for list
- ANC
 - dialled number + area(calling number) → called number
 - example: Domino's Pizza
 - query SDP for called number

FI Example: Originating Call Screening & Area Number Calling (2)

- switch may restrict no. of queries
 - protection against infinite loops
 - e.g., one query per call
 - → OCS subscription prevents orders for pizza
- solution: one more query??

FI Example: Calling Number Delivery & Unlisted Number

- conflict of goals
 - CND reveals caller
 - UN prevents revealing caller
- resolution
 - weaken one feature
 - e.g.: CND delivers only 1-111-1111-1111 for unlisted number

Informal Feature Interaction Definition in Literature

- feature interaction:
the behaviour of a feature is changed by another feature
- not all interactions are undesired
- not precisely clear what a feature actually is

Non-Monotonous Changes

- telephone switching:
new features change the behaviour
 - of base system, or
 - of other features
- example: call forwarding
 - stops to connect to dialled number
 - ▷ restricts base system behaviour
 - and
 - starts connecting to forwarded-to number
 - ▷ extends base system behaviour

The Tool genFamMem 2.0

- extracts specifications in plain CSP-OZ (not yet: ISO-Z) from a family document,
- detects feature interactions by
 - additional type checks for families
 - heuristic warnings
- helps avoiding feature interactions by generating documentation on the structure of the family.

- available freely