


Towards a  
Taxonomy of *Inconsistency* Patterns  
in Multi-View Modelling

ICMM @ STAF 2026, 2 July

dr.ir. Vadim Zaytsev, University of Twente, 

**All models are wrong**

VIA 9GAG.COM



# Seven ways models lie to each other



Code	Label	As primary	As secondary
C1	<u>Structural mismatch</u>	13	7
C2	<u>Interface contract mismatch</u>	4	5
C3	<u>Behavioural contradiction</u>	3	6
C4	<u>Requirement satisfaction gap</u>	7	1
C5	<u>Terminology divergence</u>	3	3
C6	<u>Traceability disruption</u>	7	6
C7	<u>Temporal skew</u>	3	9



<https://circus.github.io/tip/>



# *Inconsistency* considered harmful

- stale trace links
- mismatched ports
- missing implementation elements
- uncovered requirements
- contradictory behaviour
- naming drift
- outdated versions
- . . .

# Contributions

- evidence map
  - 19 sources
  - 46 extracted examples
- core dataset
  - 40 examples kept
- seven-category taxonomy
- coding rules
  - primary / secondary codes
  - temporal overlays
  - support-only rows
- public website to browse, inspect, validate

- source
- categories
- domain
- views
- artefacts
- quotes
- summary



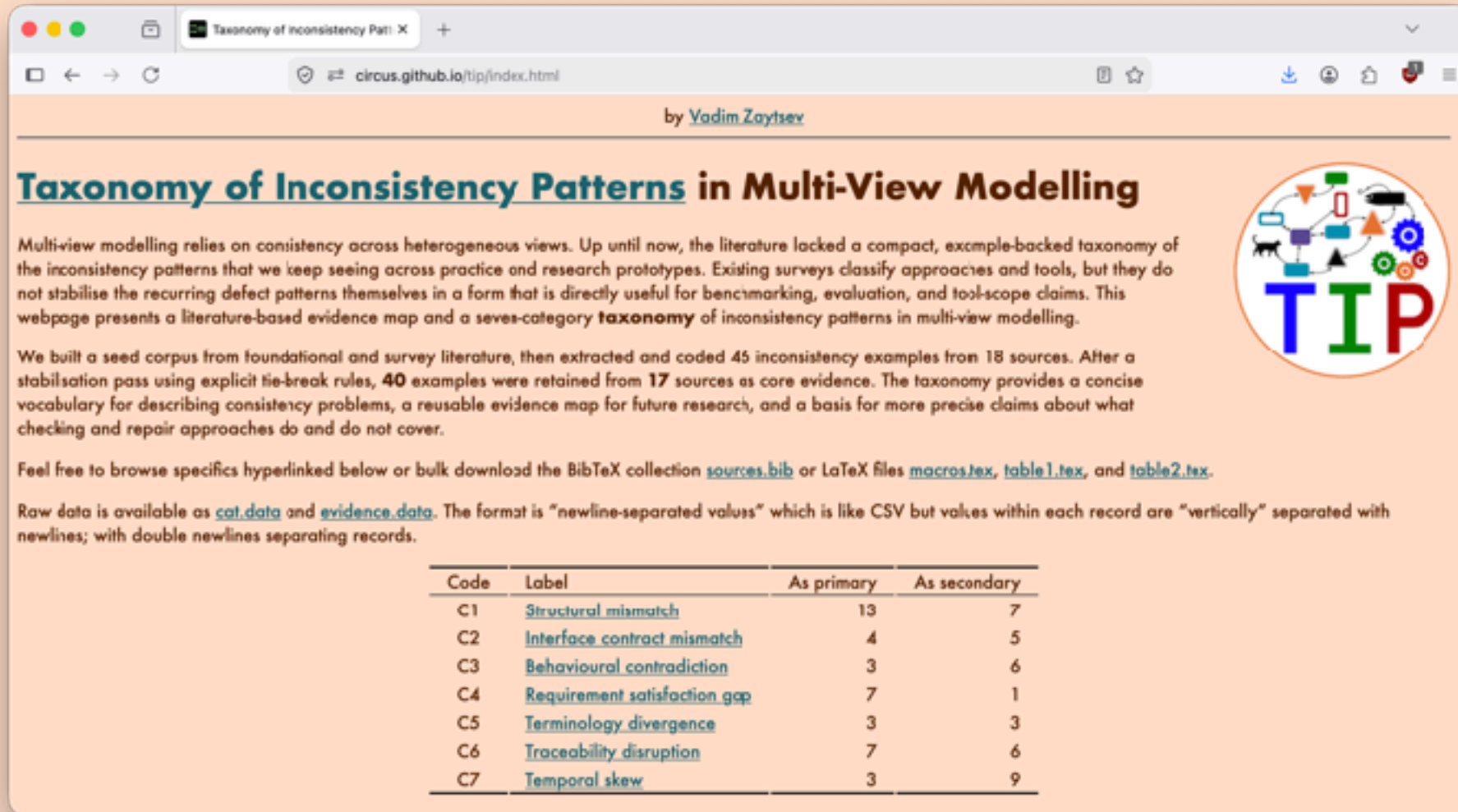
<https://circus.github.io/tip/>

# Seven suspects (failure modes?)

- **C1 Structural mismatch** – the parts do not line up
- **C2 Interface contract mismatch** – the plug does not fit
- **C3 Behavioural contradiction** – the choreography conflicts
- **C4 Requirement satisfaction gap** – the promise is not justified
- **C5 Terminology divergence** – the names betray us
- **C6 Traceability disruption** – the thread is broken
- **C7 Temporal skew** – the views come from different times



# Let's inspect the evidence



by [Yadim Zaytsev](#)

## Taxonomy of Inconsistency Patterns in Multi-View Modelling

Multi-view modelling relies on consistency across heterogeneous views. Up until now, the literature lacked a compact, example-backed taxonomy of the inconsistency patterns that we keep seeing across practice and research prototypes. Existing surveys classify approaches and tools, but they do not stabilise the recurring defect patterns themselves in a form that is directly useful for benchmarking, evaluation, and tool-scope claims. This webpage presents a literature-based evidence map and a seven-category **taxonomy** of inconsistency patterns in multi-view modelling.

We built a seed corpus from foundational and survey literature, then extracted and coded 45 inconsistency examples from 18 sources. After a stabilisation pass using explicit tie-break rules, **40** examples were retained from **17** sources as core evidence. The taxonomy provides a concise vocabulary for describing consistency problems, a reusable evidence map for future research, and a basis for more precise claims about what checking and repair approaches do and do not cover.

Feel free to browse specifics hyperlinked below or bulk download the BibTeX collection [sources.bib](#) or LaTeX files [macros.tex](#), [table1.tex](#), and [table2.tex](#).

Raw data is available as [cat.data](#) and [evidence.data](#). The format is "newline-separated values" which is like CSV but values within each record are "vertically" separated with newlines; with double newlines separating records.

Code	Label	As primary	As secondary
C1	<a href="#">Structural mismatch</a>	13	7
C2	<a href="#">Interface contract mismatch</a>	4	5
C3	<a href="#">Behavioural contradiction</a>	3	6
C4	<a href="#">Requirement satisfaction gap</a>	7	1
C5	<a href="#">Terminology divergence</a>	3	3
C6	<a href="#">Traceability disruption</a>	7	6
C7	<a href="#">Temporal skew</a>	3	9



<https://circus.github.io/tip/>



# Let's inspect the statistics



Code	Label	As primary	As secondary
C1	<u>Structural mismatch</u>	13	7
C2	<u>Interface contract mismatch</u>	4	5
C3	<u>Behavioural contradiction</u>	3	6
C4	<u>Requirement satisfaction gap</u>	7	1
C5	<u>Terminology divergence</u>	3	3
C6	<u>Traceability disruption</u>	7	6
C7	<u>Temporal skew</u>	3	9



# Small count $\neq$ small problem

- Low (**primary**) counts (3)
  - **C3** Behavioural contradiction
  - **C5** Terminology divergence
  - **C7** Temporal skew
- harder to formalise?
- less often written as crisp examples?
- treated as background context?
- difficult to extract from published papers?
- under-supported by tools?



# On the nature of coding

- **Ambiguity** is good

*• A requirement is no longer tested because the trace link decayed after evolution.*

- **C4** Requirement satisfaction gap
- **C6** Traceability disruption
- **C7** Temporal skew
- No **C8** Miscellaneous?
  - ambiguity is a coding property, not a category



# Takeaways

<https://circus.github.io/tip/>

- sharper and clearer **claims** for tools
- better **questions** as reviewer
- more **balance** for benchmarks
- reusable **challenge** cases
- more honest **comparison** across papers
- shared **vocabulary**
- **browseable** evidence
- **name the failure before fixing it**
- **questions?**

