

# CODE REVIEW COMPREHENSION

Reviewing Strategies Seen Through  
Code Comprehension Theories

Pavlina Wurzel Goncalves

Pooja Rani

Margaret-Anne Storey

Diomidis Spinellis

Alberto Bacchelli



University of  
Zurich<sup>UZH</sup>

**UVIC**  
UNIVERSITY OF VICTORIA CANADA

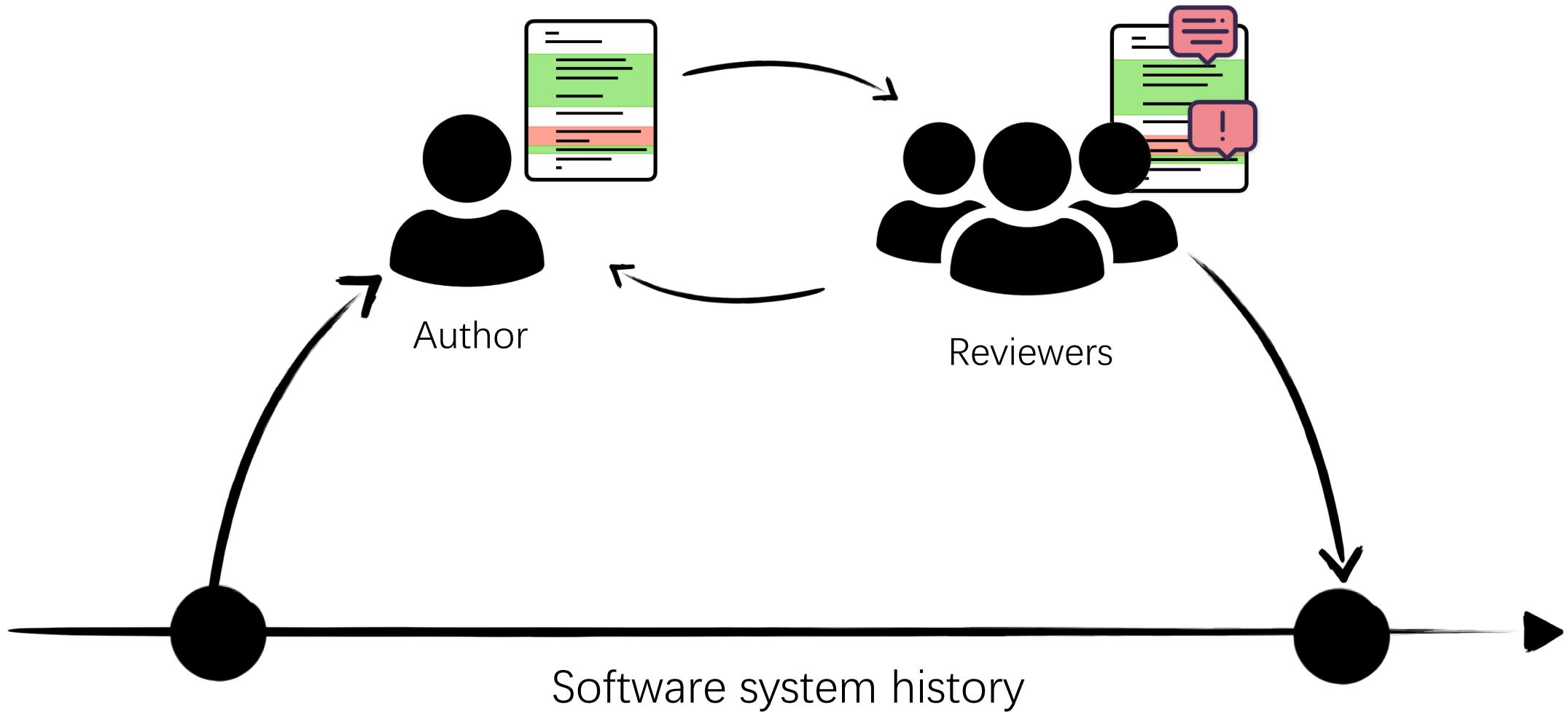


ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS

**TU Delft**  
Delft  
University of  
Technology

ICPC  
28.4.2025  
Ottawa, CAN

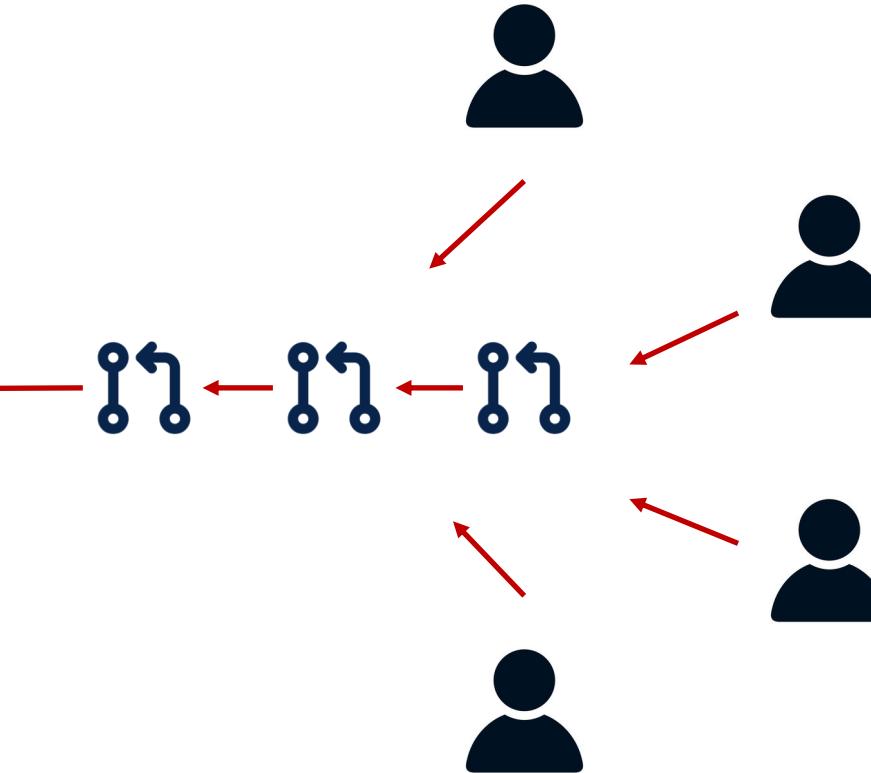
# Code Review



# Code review is understanding

- *Bacchelli, Bird (2013)*

**SOFTWARE  
SYSTEM**



Incremental

Iterative

Interactive

Code comprehension driven by  
**opportunistic strategies**



**Opportunistic reviewing strategies?**

**10  
great reviewers**

**25  
reviews**

## Think-aloud Observation Interview



## Theory-driven thematic analysis

**\*\*I:\*\*** So personality also plays a part

**\*\*71:\*\*** Of course. If someone is stubborn,

**\*\*I:\*\*** And in terms of the programmer, what is crucial to the conflicts? So that they

**\*\*71:\*\*** Perhaps some openness to new opinions, your own truth, and rather find out what you want rather than advocate for your things and do it. So, like that. Verifying new views. People can always talk about why it is that happened to me, and not only to me, that overwrite the code and close the issue "I'm sorry, bye."

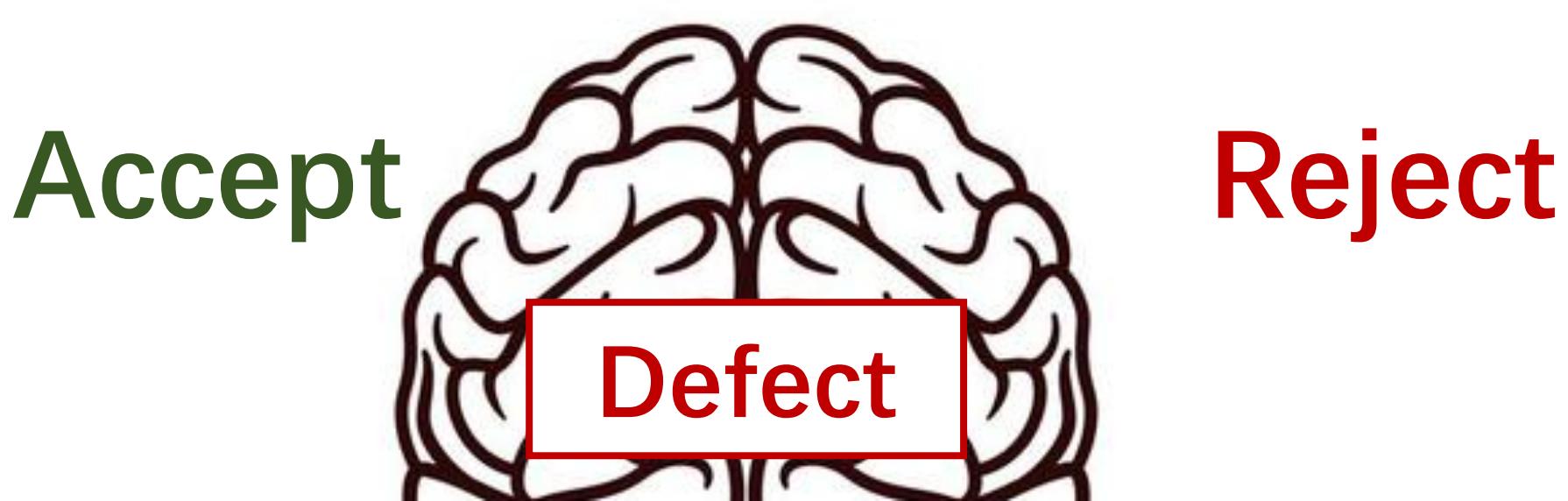
**\*\*I:\*\*** That they do not talk about it at all.

**\*\*71:\*\*** Yes. Then they say, "I did not do it." Which I find strange. That, it's too much work. It's really not nice.

**\*\*I:\*\*** Like, he can do it, but he has permission".

**\*\*71:\*\*** Yeah. Exactly. I think this is what makes people especially not like each other, but "Ehhh, why are you

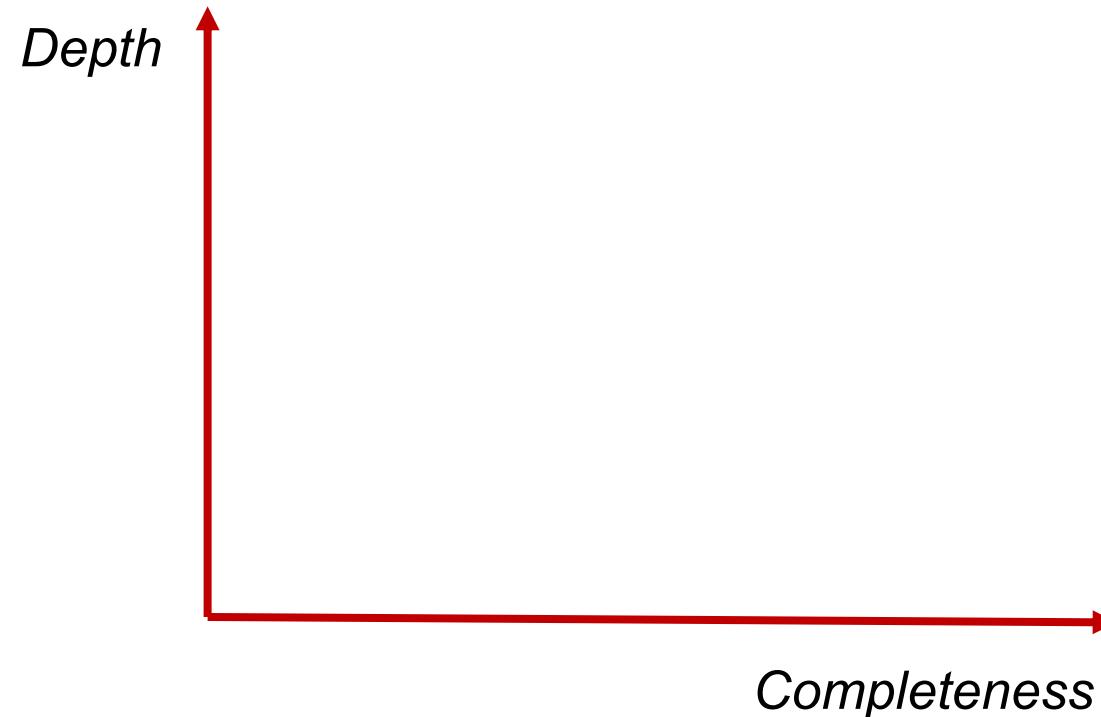
# Understanding central to finalizing the review, giving a verdict and identifying issues



# Review comprehension scope

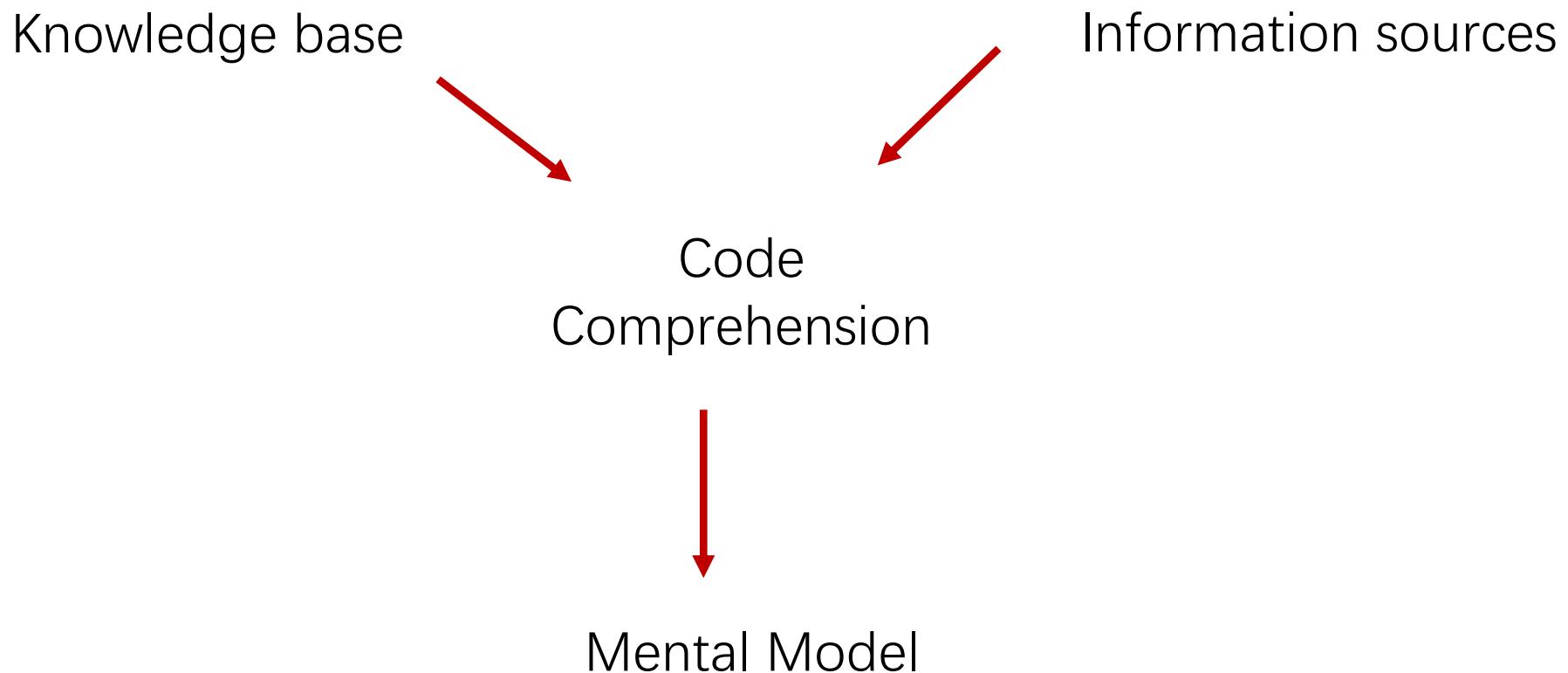
(Piaget, 1964)

- Full
- Partial
- Focused
- Shallow

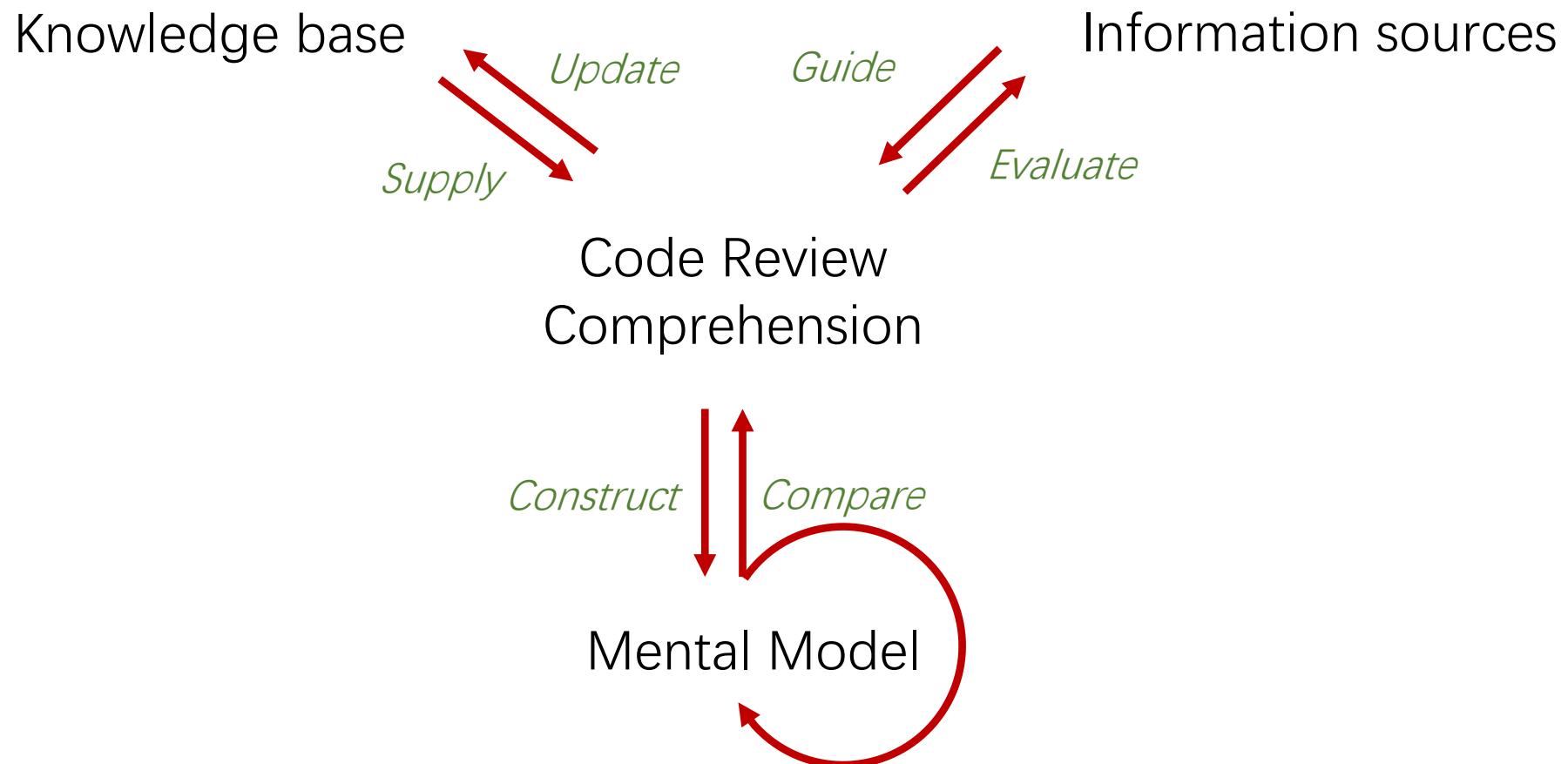


# Code comprehension

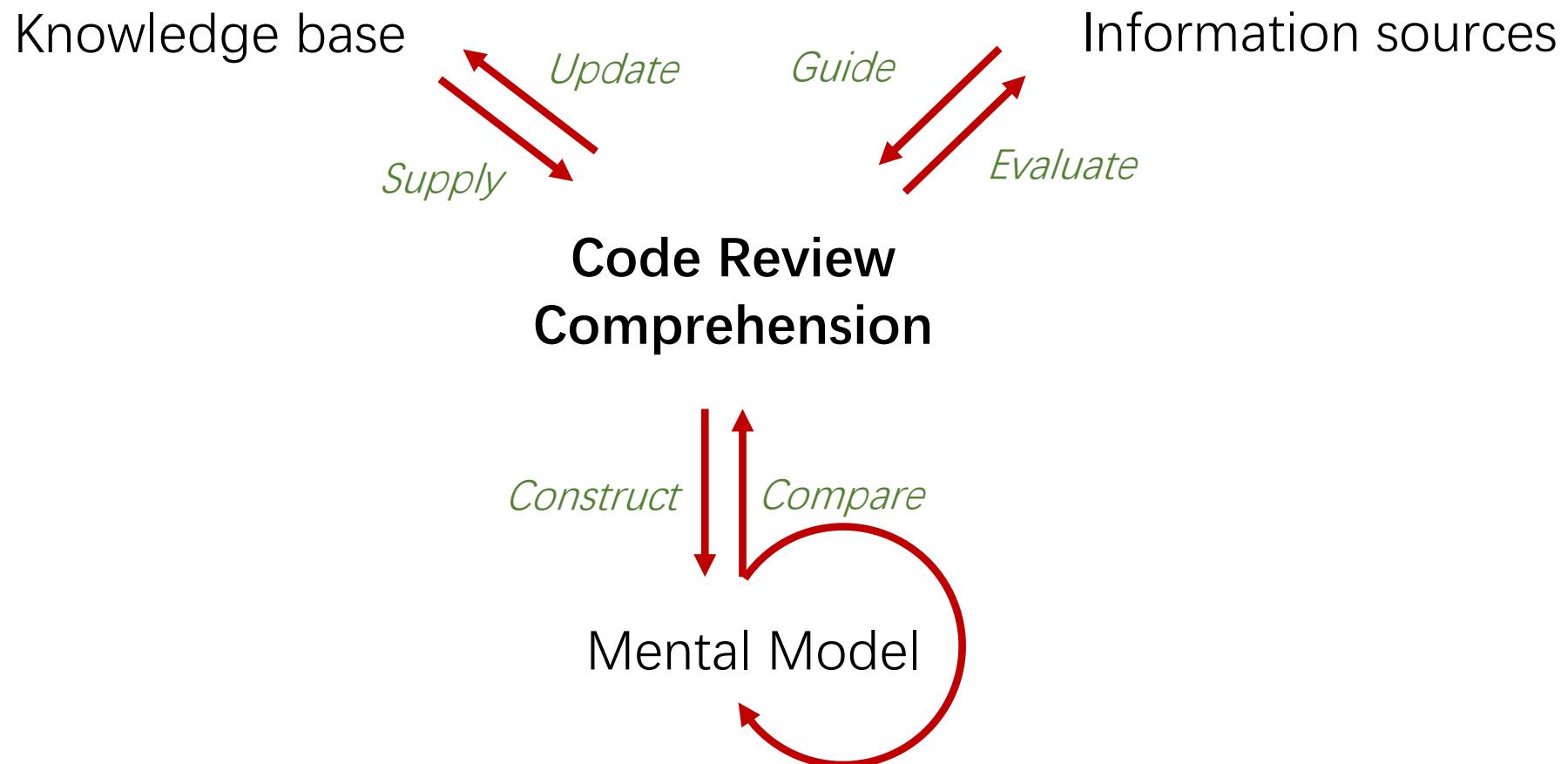
(Letovsky, 1987)



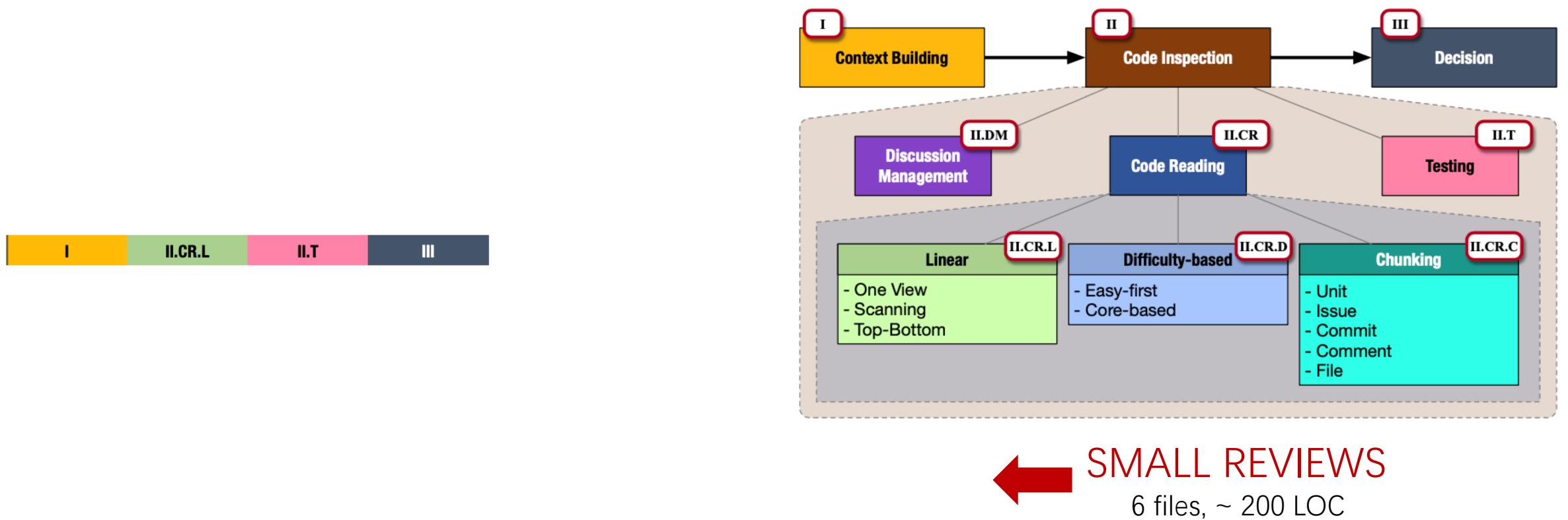
# Code Review Comprehension Model



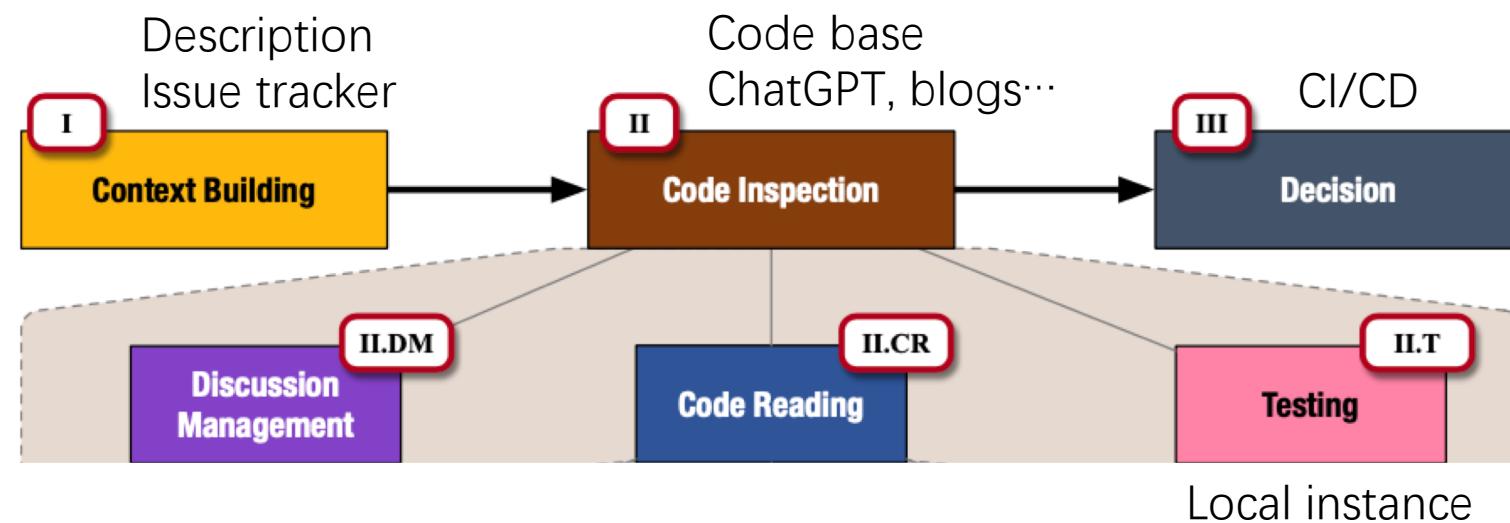
# Code Review Comprehension Model







# Information sources



## Internal code review tools

# Knowledge base

SE and programming knowledge

Own experience

- Previously used solutions
- Team mates expertise

Existing mental models

- Software system
- PR

RECOGNITION OF CORRECT  
IMPLEMENTATION

COMPARISON TO THE  
ACTUAL PR



# Mental Model Ecosystem



Existing mental models

- Software system
- PR

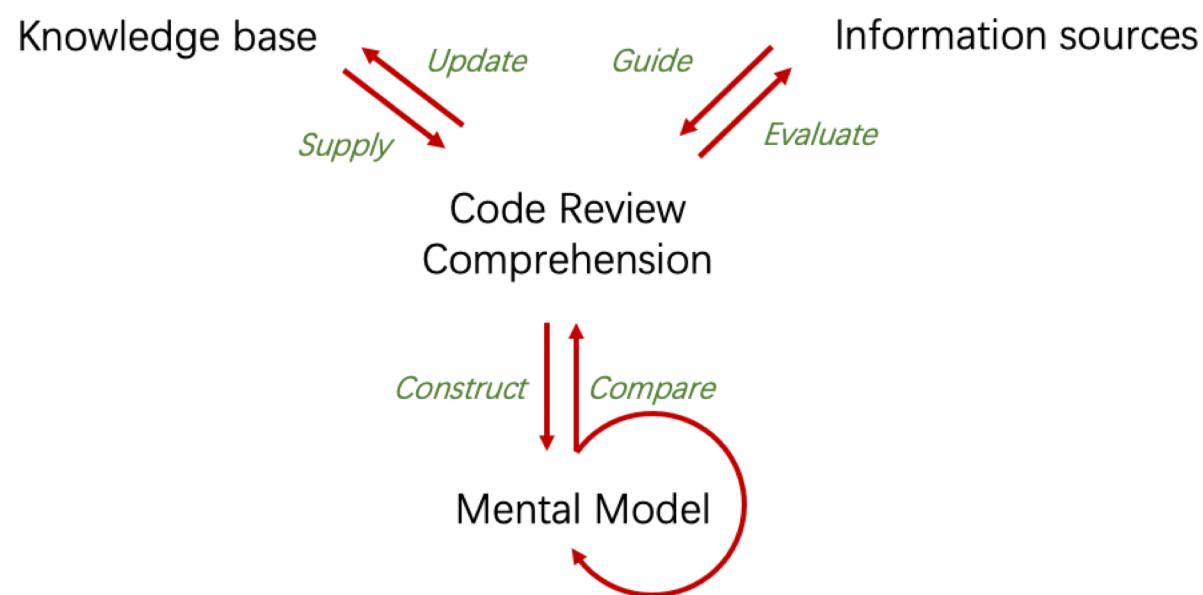


**Mental Model  
of the Actual PR**

Alternatives

- Ideals
- Expectations

# Code Review Comprehension Model



Interpreting and contextualizing reviewer behavior

Creating tools aligning with natural reviewing strategies

Teaching and reflecting on reviewing strategies