

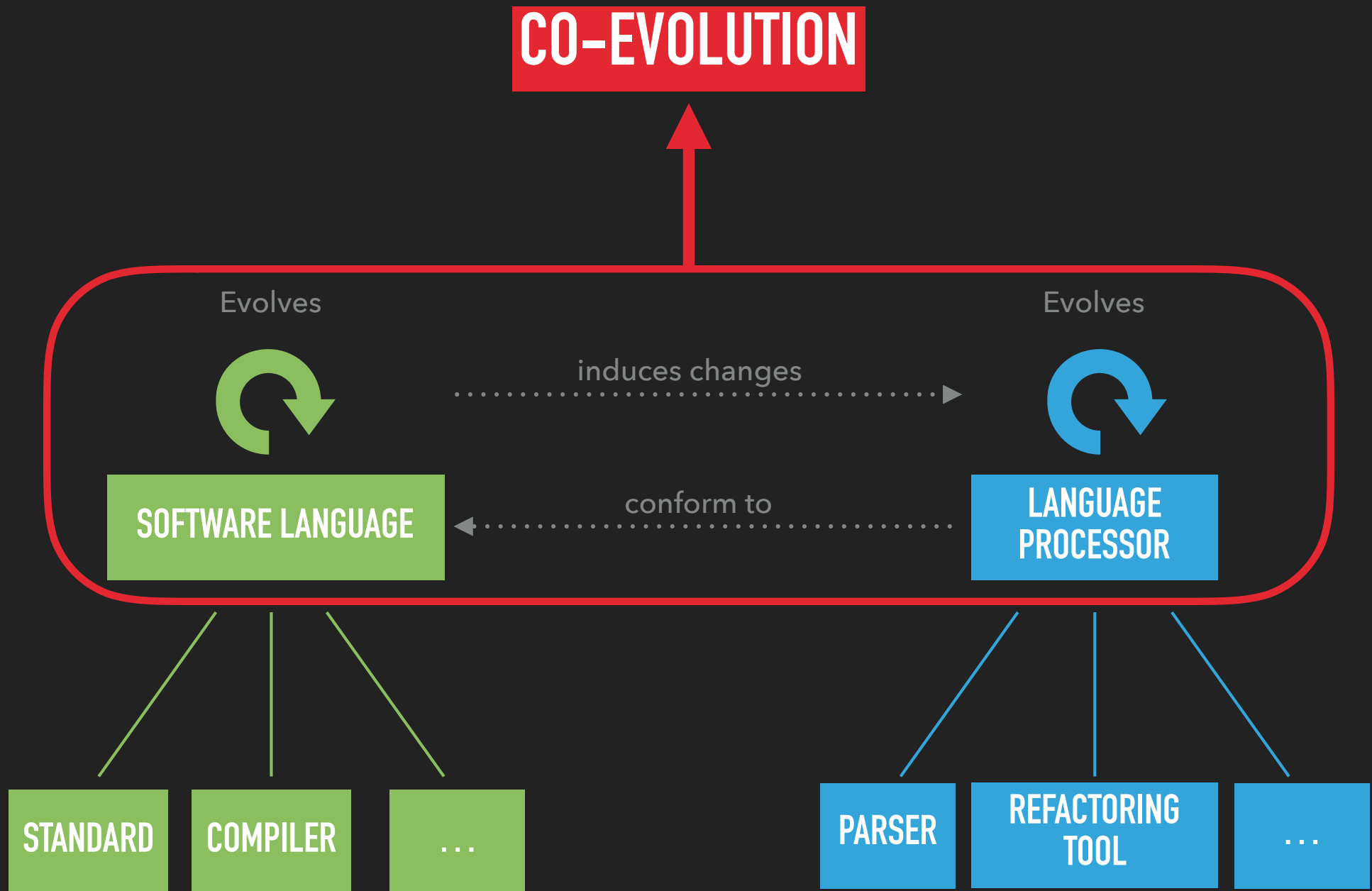
Anna Eilertsen  
Anya Bagge  
University of Bergen,  
Norway

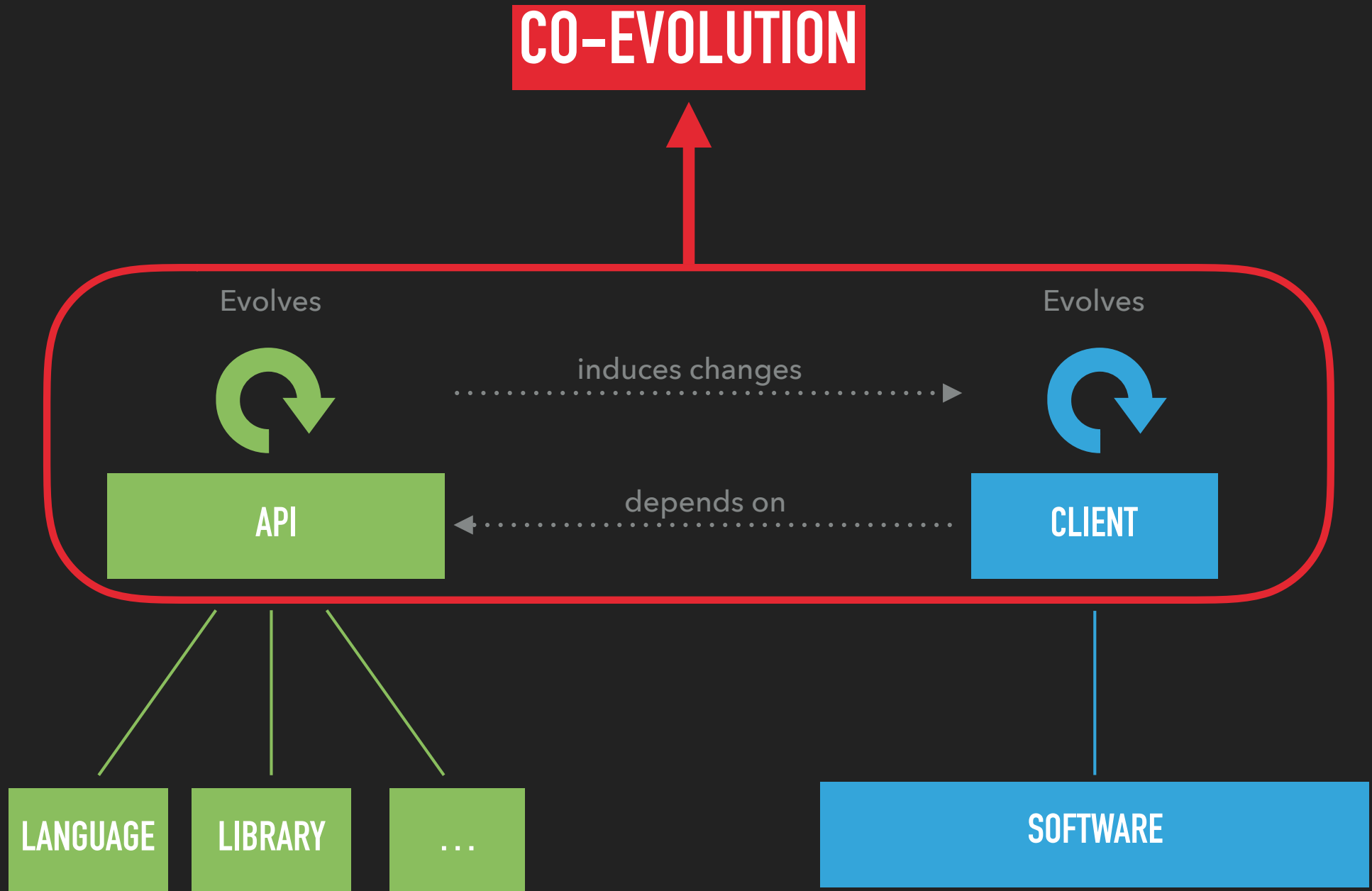


EXPLORING API/CLIENT

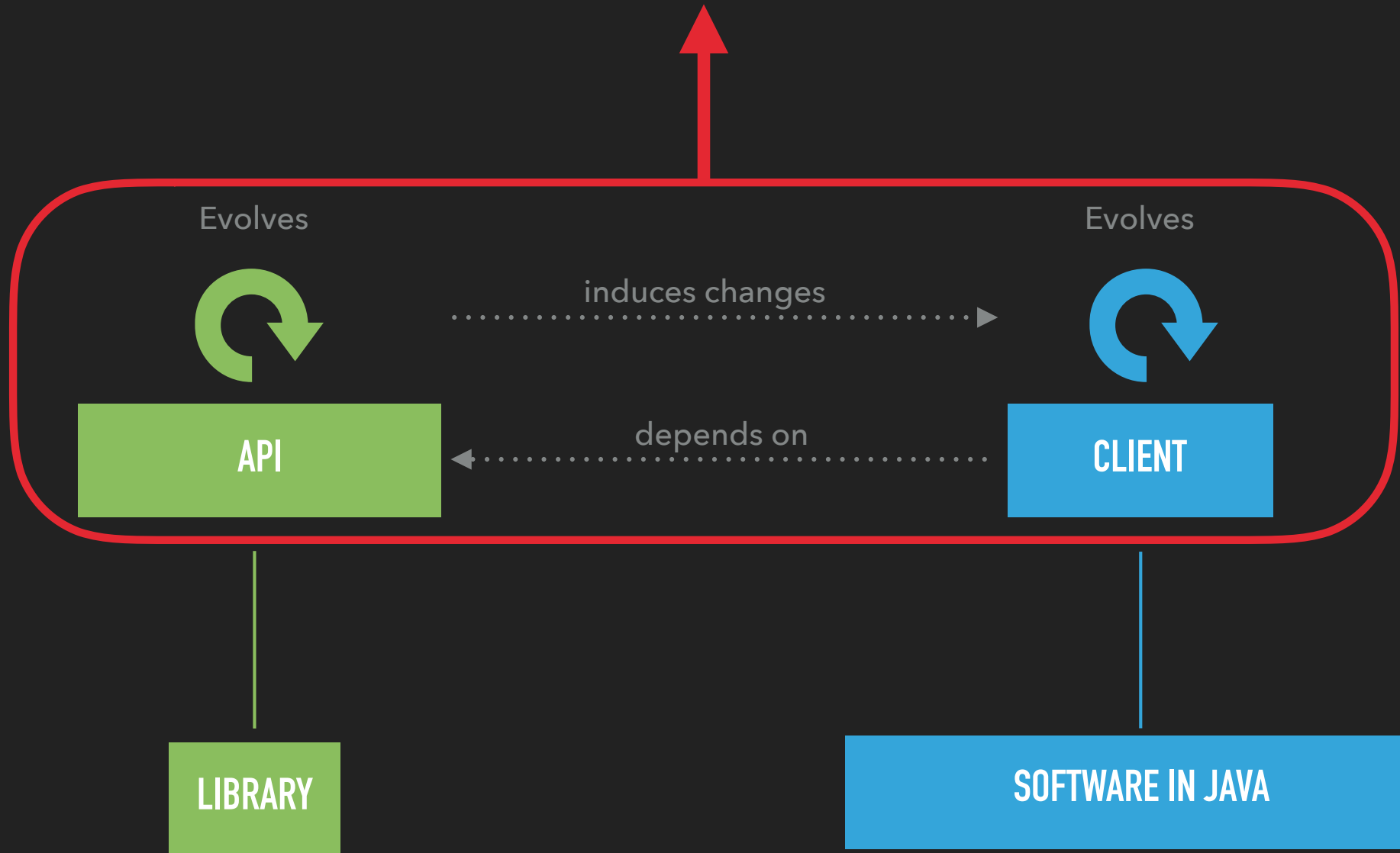
---

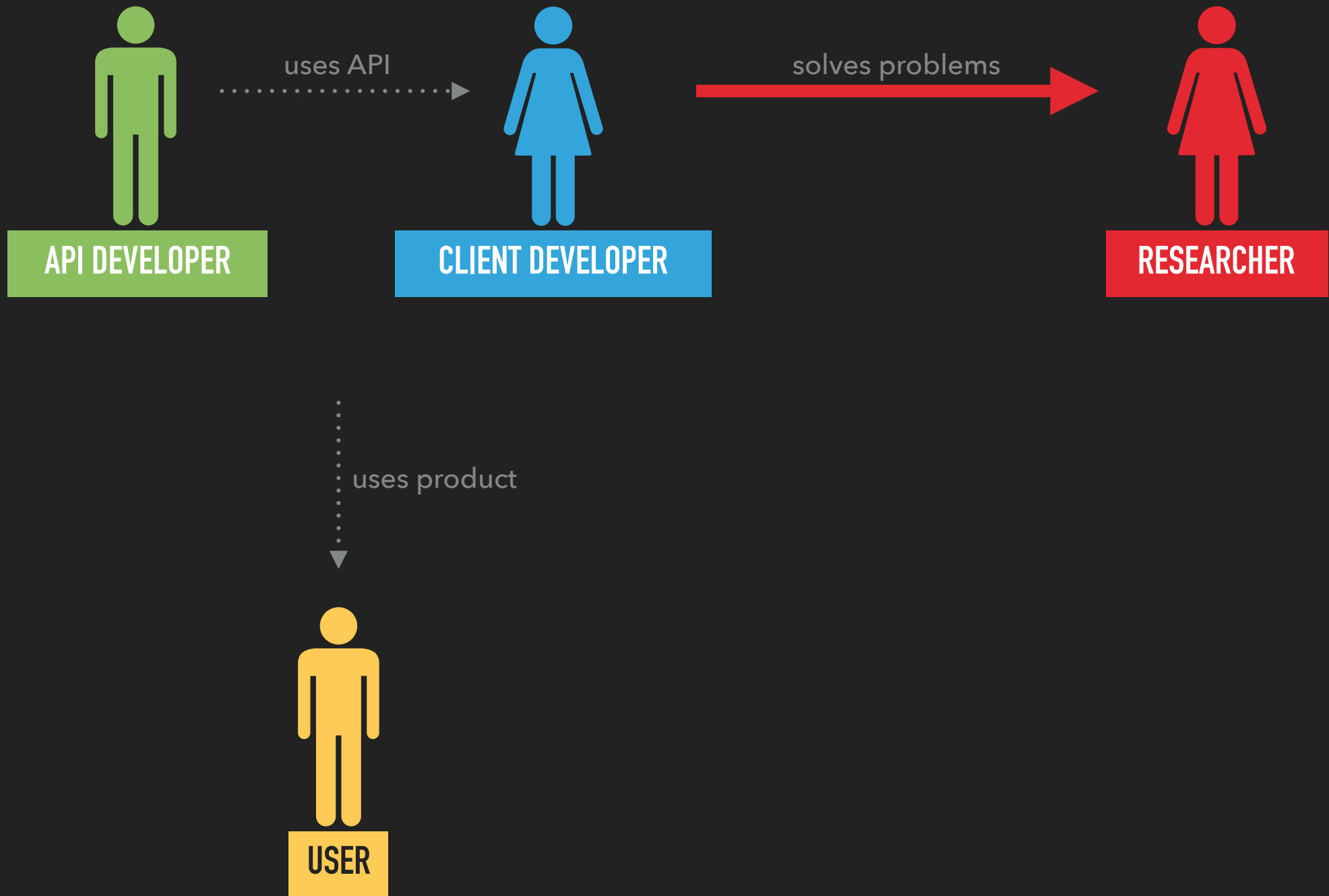
**CO-EVOLUTION**

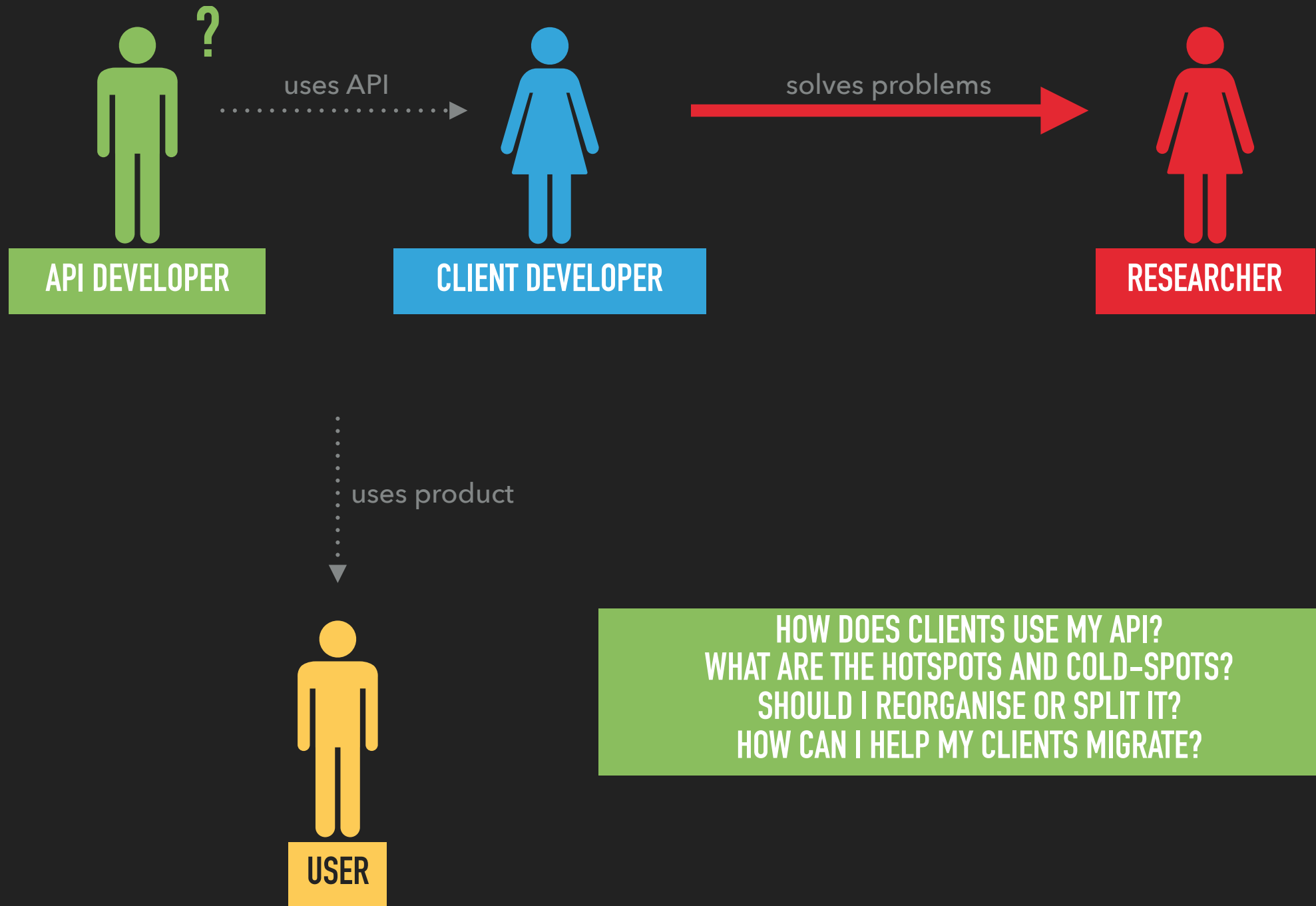


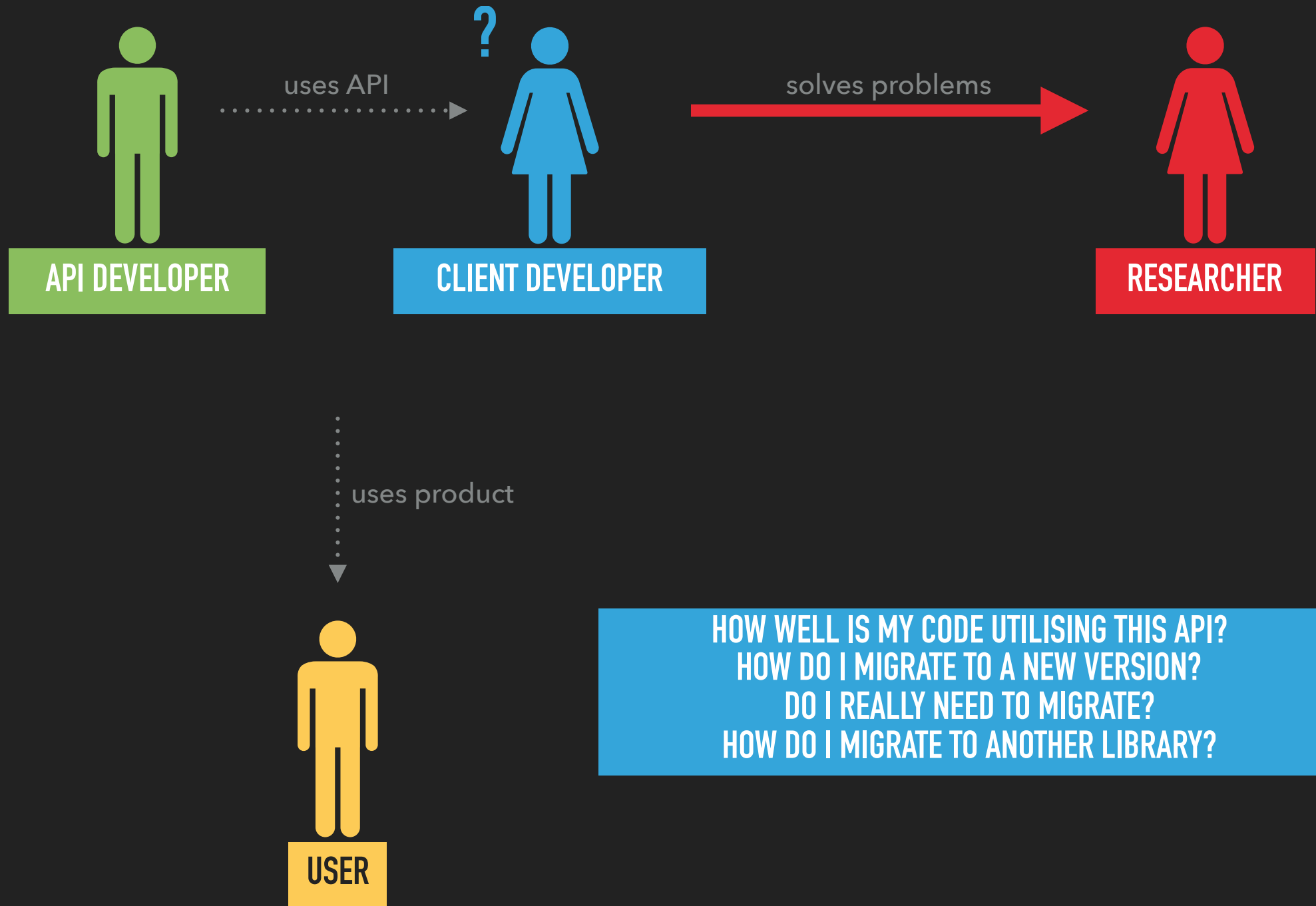


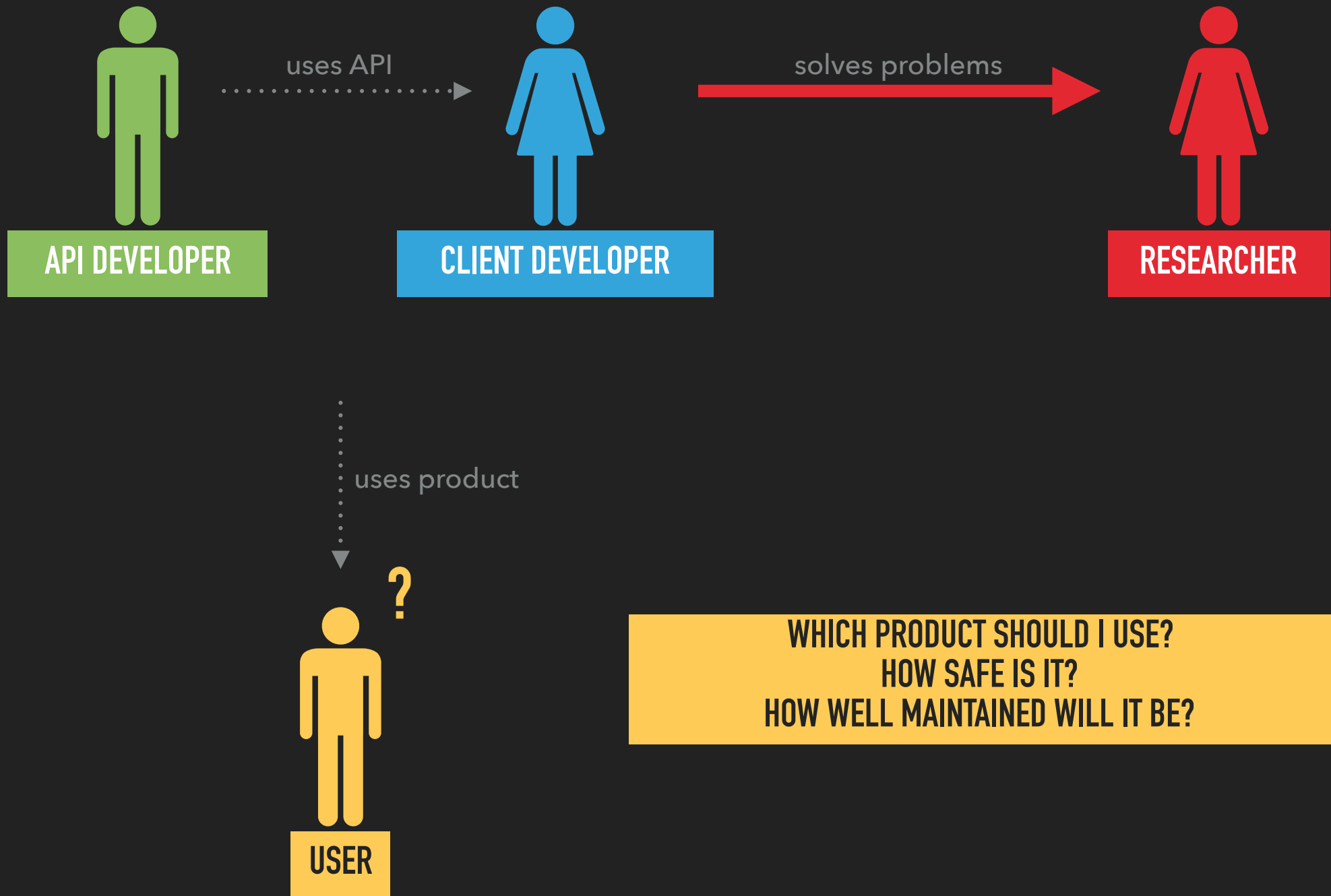
# CO-EVOLUTION



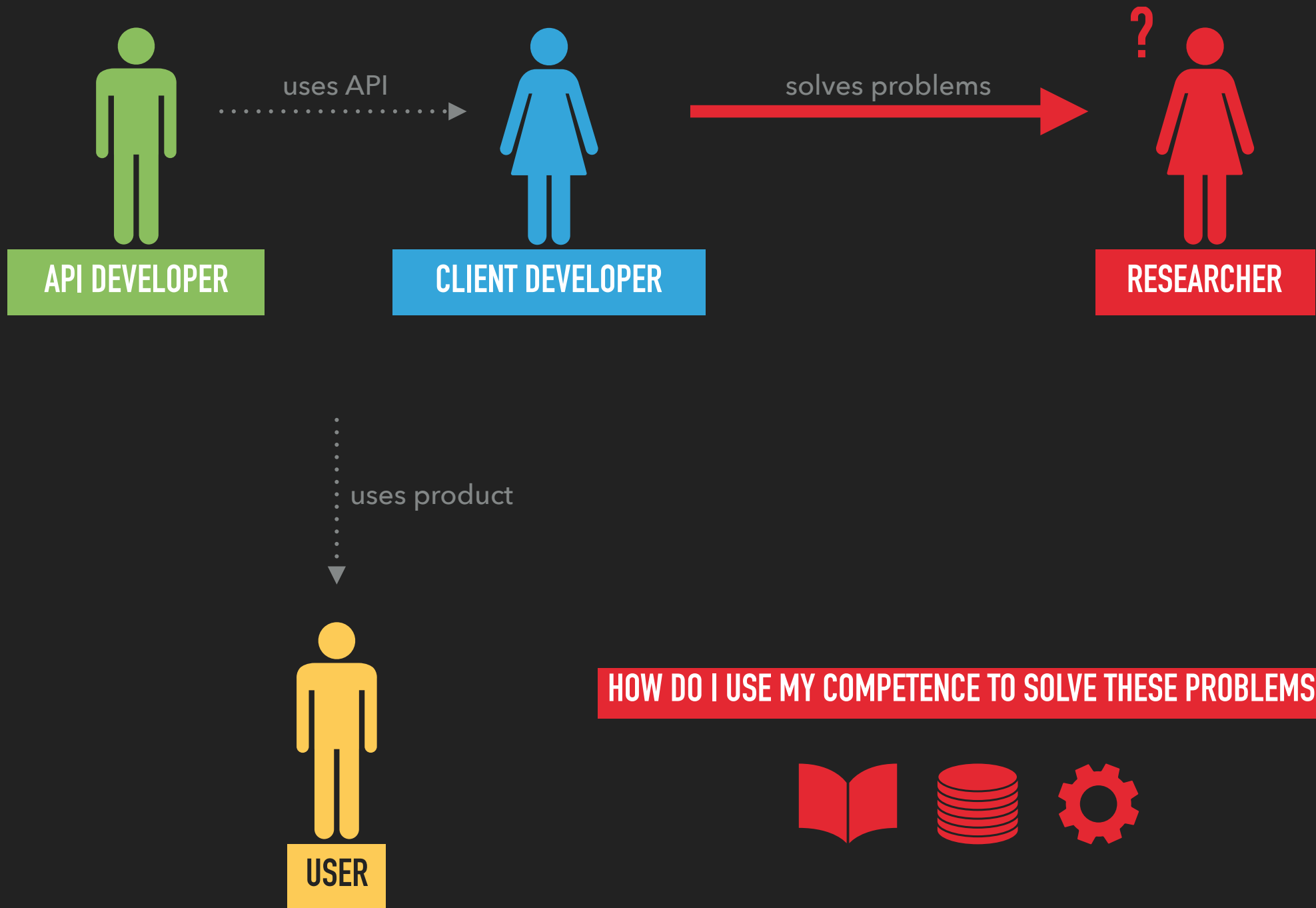












- Are the number of clients of this API increasing or decreasing?
- Do clients of this API usually immediately update their code upon an API release?
- What are hotspots in my API?
- Which methods are usually used together? In what order?

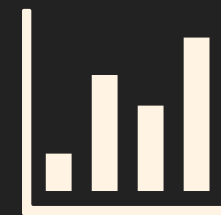
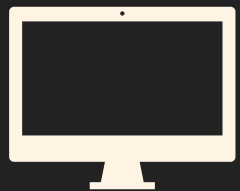
## QUESTIONS

'REAL PROGRAMS'

COLLECT

ABSTRACT

DATASET

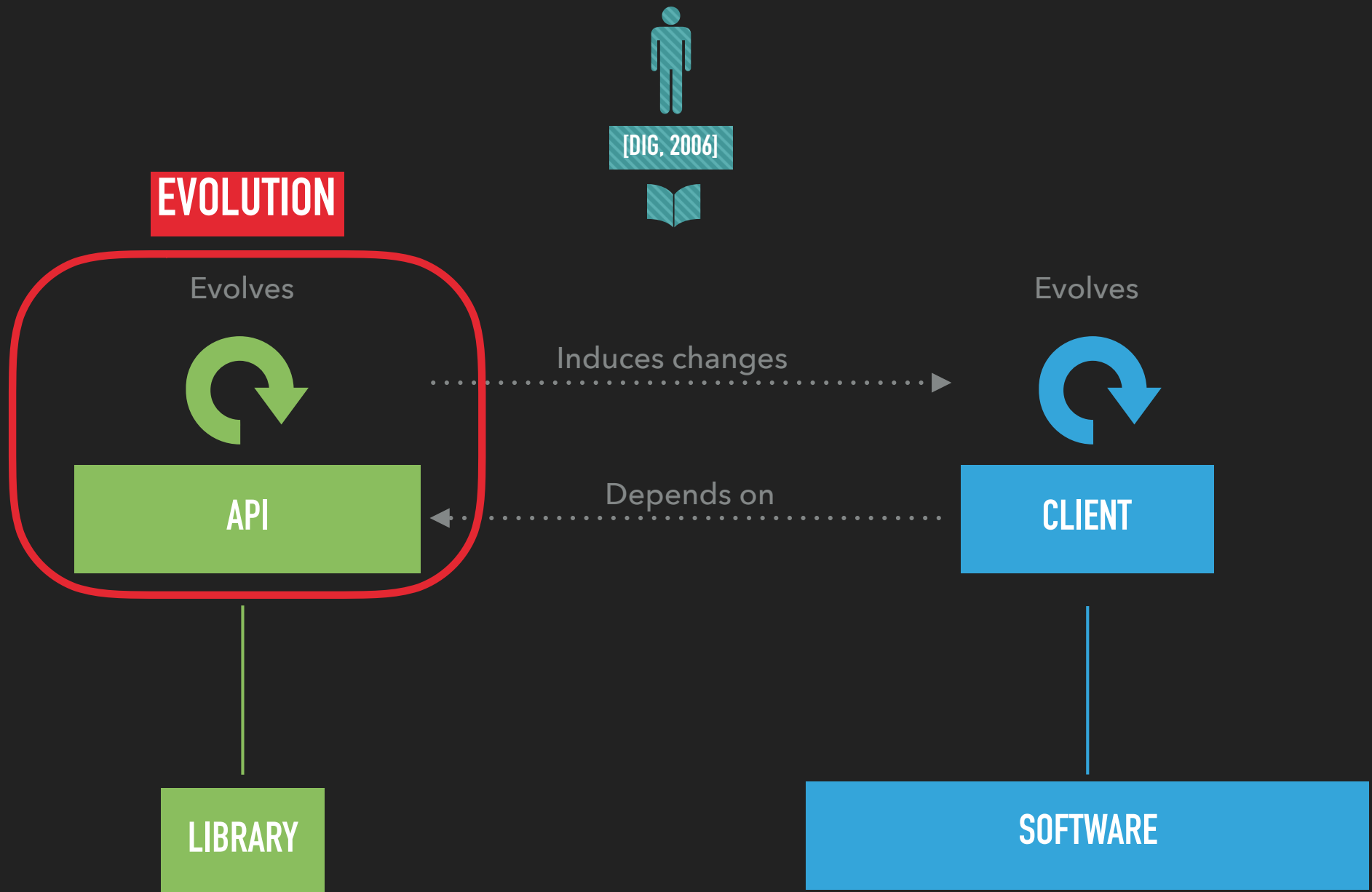


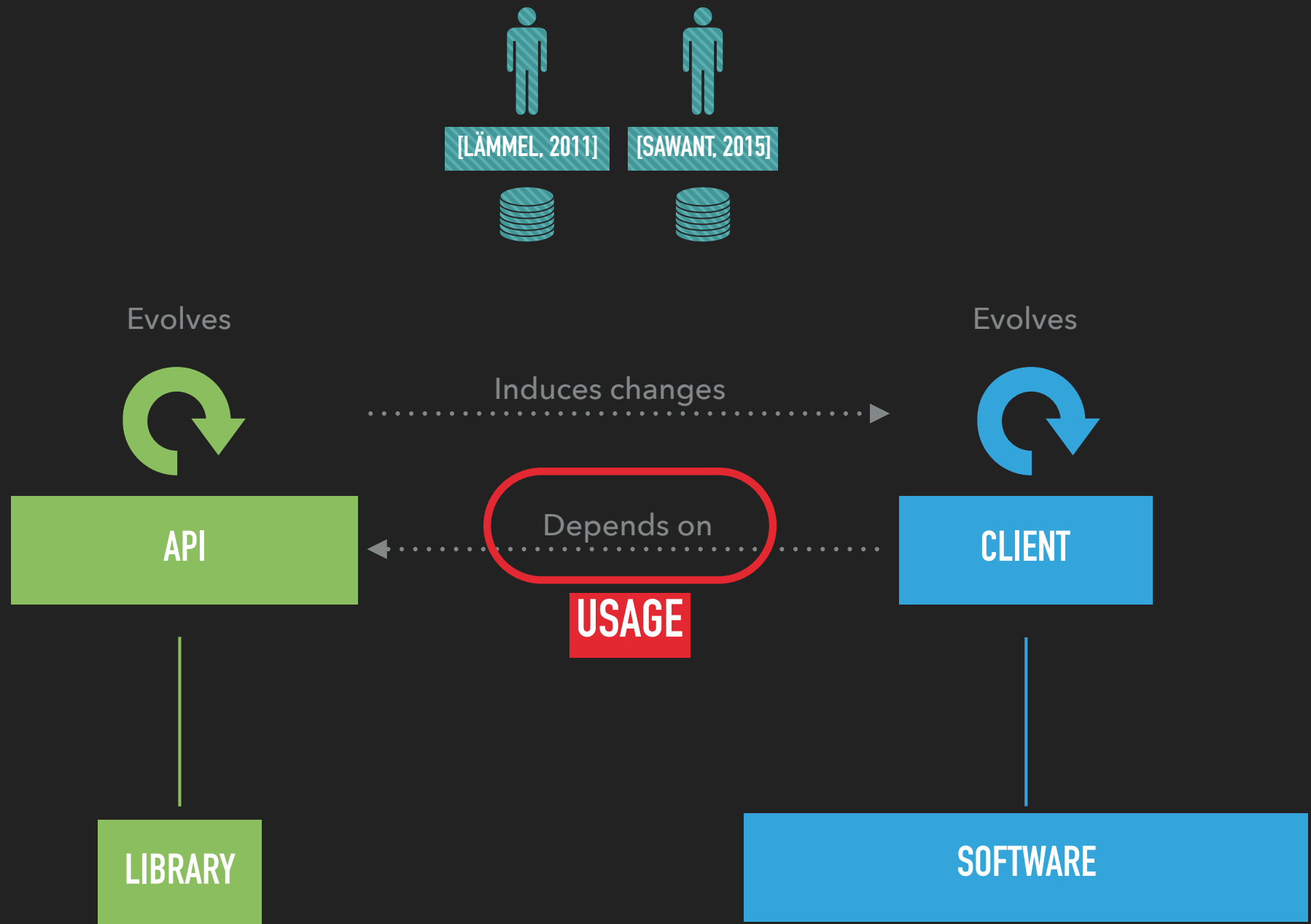
API

CLIENT











[DIG, 2006]



5 handpicked APIs  
2 major releases  
breaking changes  
manual analysis



[LÄMMEL, 2011]



1476 SourceForge projects  
69 APIS  
per commit  
automated analysis



[SAWANT, 2015]



20'263 GitHub projects  
5 popular APIs  
per commit  
automated analysis



[SAWANT, 2017]

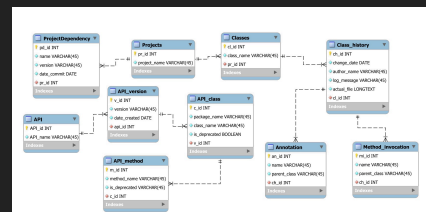


SQL queries

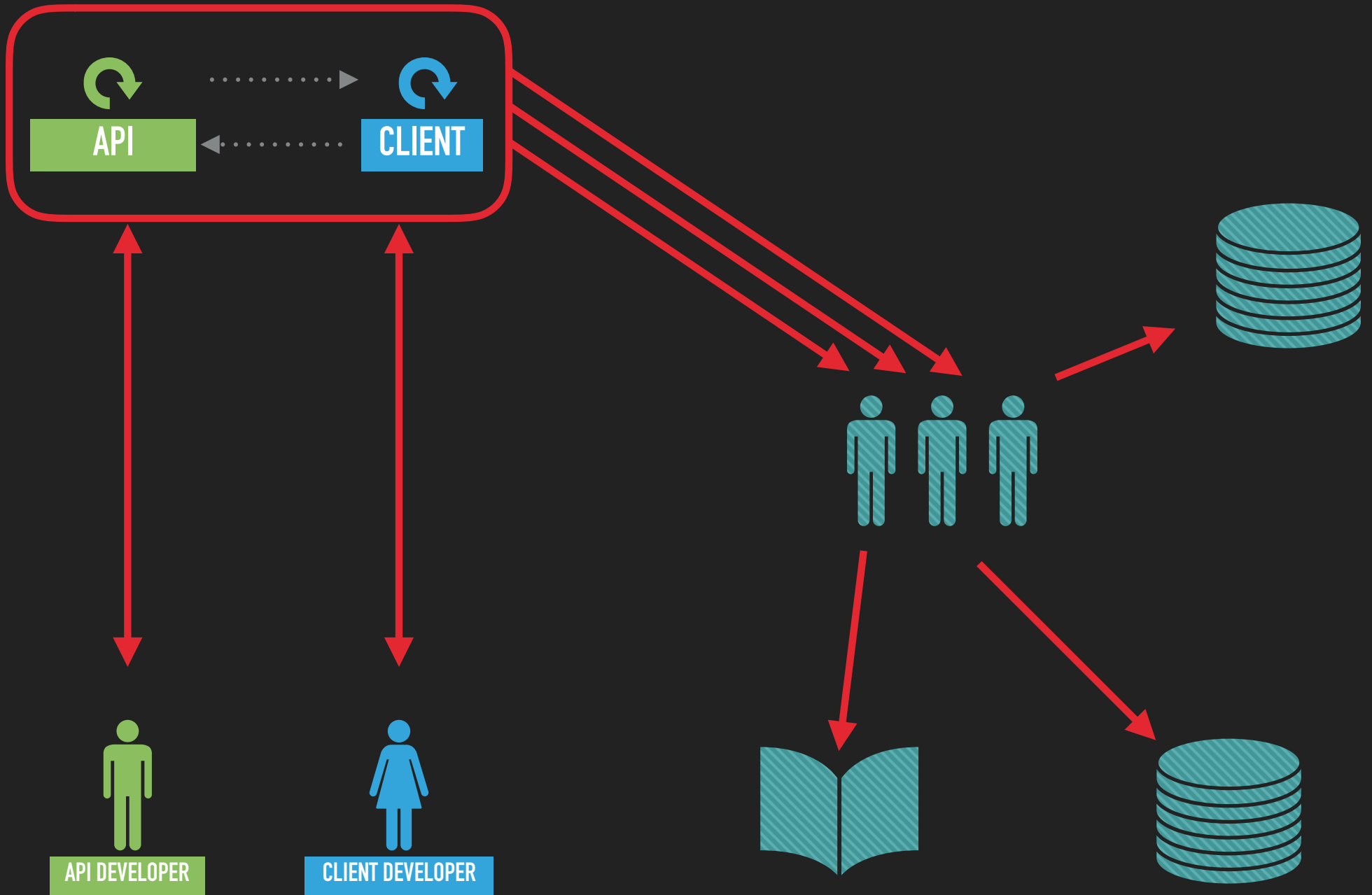


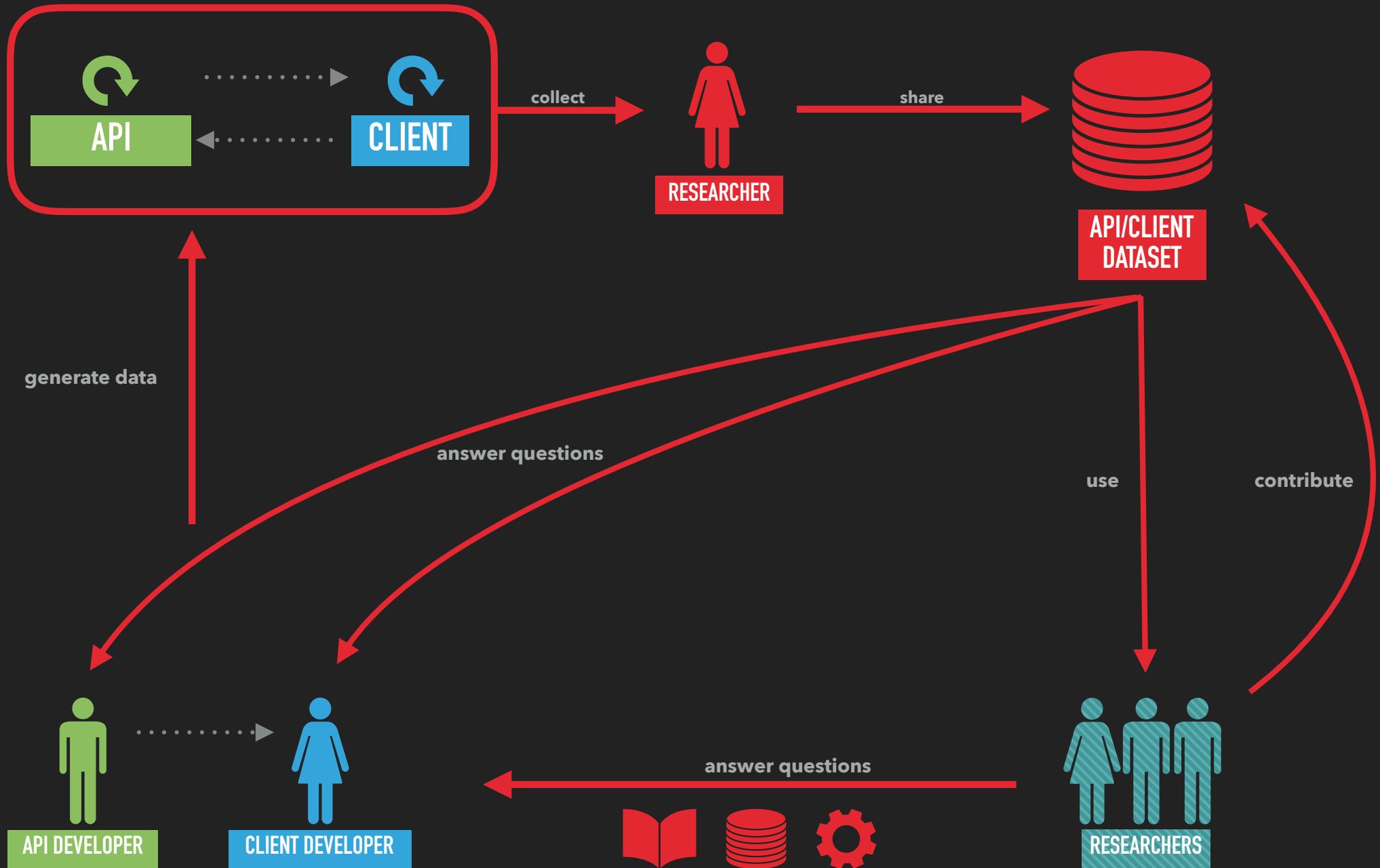
“80% REFACTORINGS”

“API FOOTPRINT”



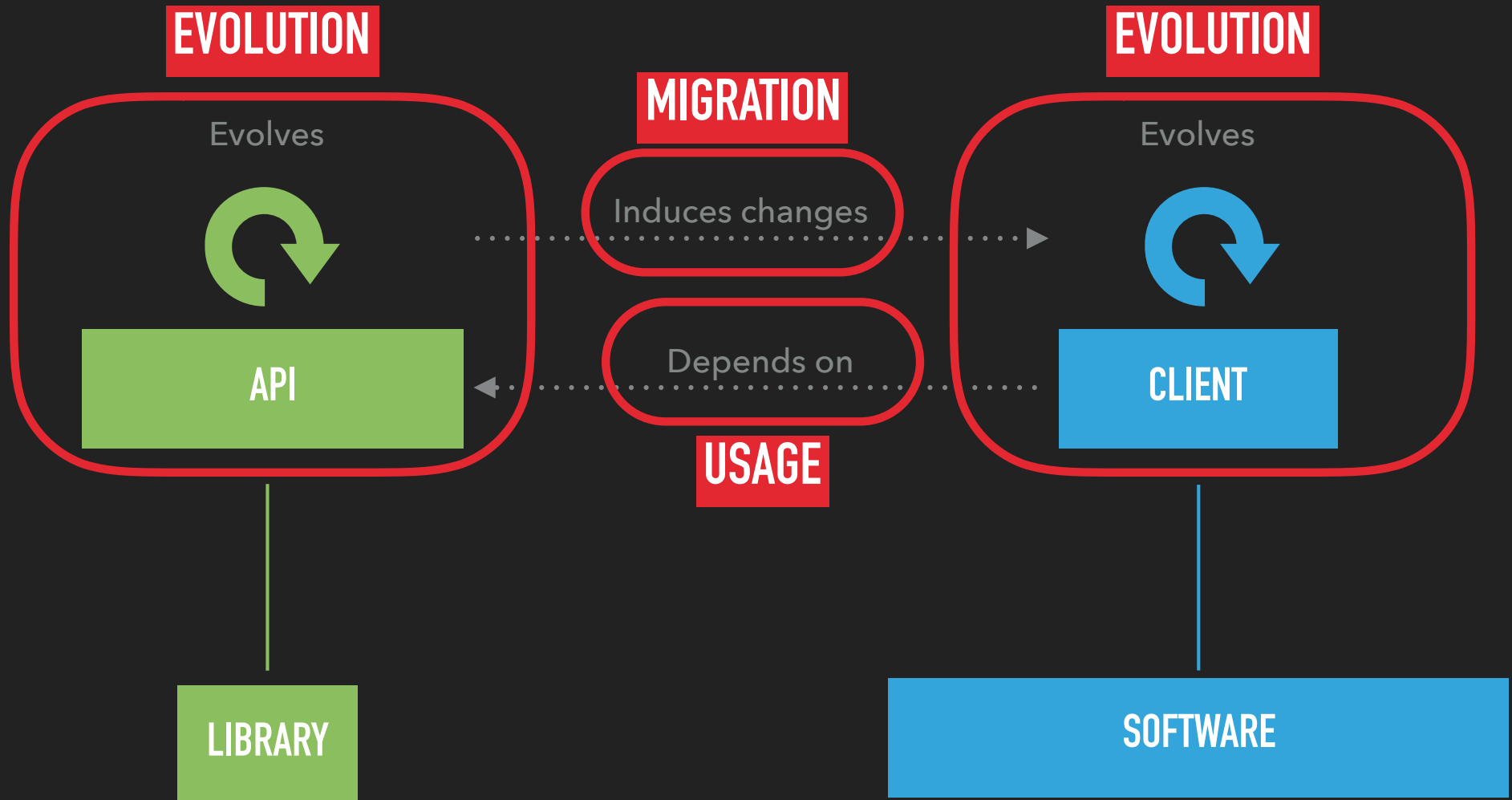
“DO CLIENTS MIGRATE?”  
“API USE COVERAGE”







# CO-EVOLUTION



## PROBLEMS

- ▶ Complex interactions cannot be used automatically
- ▶ Results of previous research are difficult to reuse

## OUR SOLUTION

- ▶ Make extendable dataset:  
    'simple building blocks, complex patterns'
- ▶ Use knowledge base (Jena / Graal)
- ▶ Abstract (and extend) by **inference rules**
- ▶ Extendable by other researchers: **new rules, new results**

