



Time-Traveling Queries for Faster Debugging and Program Comprehension

Maximilian Willembinck, Steven Costiou, Anne Etien, Stéphane Ducasse

► To cite this version:

Maximilian Willembinck, Steven Costiou, Anne Etien, Stéphane Ducasse. Time-Traveling Queries for Faster Debugging and Program Comprehension. Journées Nationales du Génie de la Programmation et du Logiciel 2022, Jun 2022, Vannes, France. hal-03738585

HAL Id: hal-03738585

<https://inria.hal.science/hal-03738585>

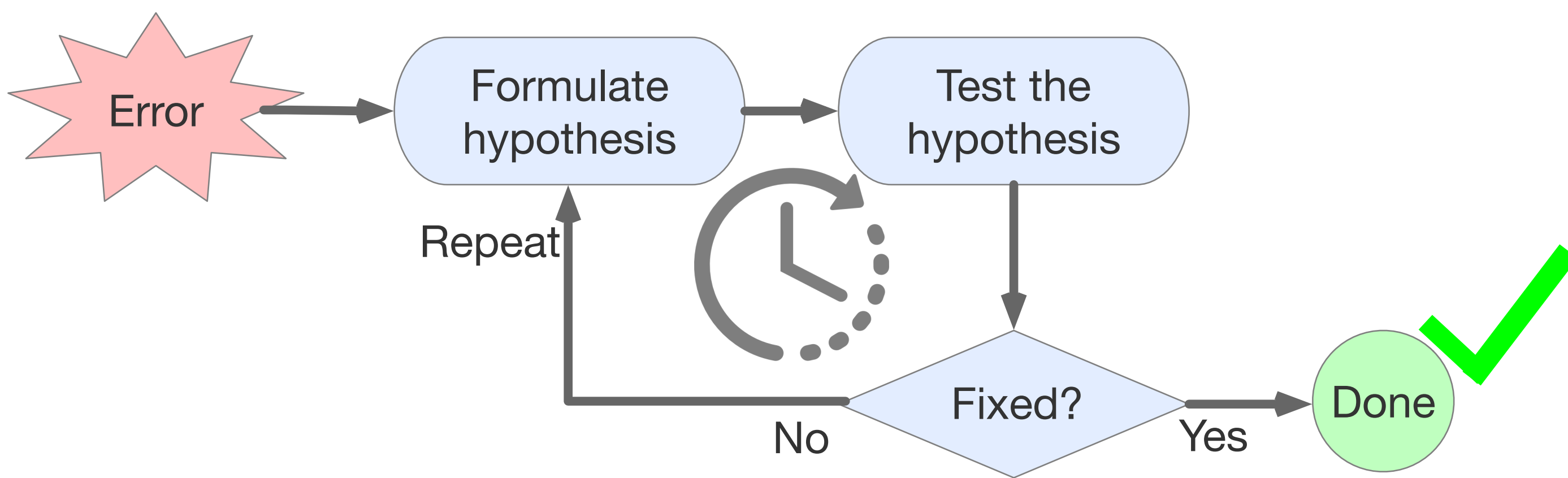
Submitted on 26 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Context: The debugging process

Debugging is a time-consuming iterative process:



To understand the behavior of a program, developers ask program comprehension questions

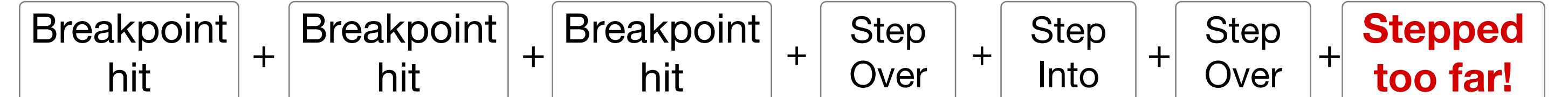
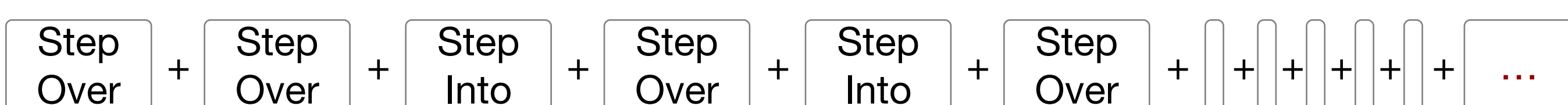
- When during the execution is this method called?
- Where are instances of this class created?
- Where is this variable or data being accessed?
- Etc.

BETTER PROGRAM COMPREHENSION => SHORTER DEBUGGING PROCESS

Problem: Understanding programs for debugging is difficult

To find answers, developers explore their program executions using debugging tools

Debugging question: What is the value of this variable during the execution?



Tedious! 😞

Can we do better?

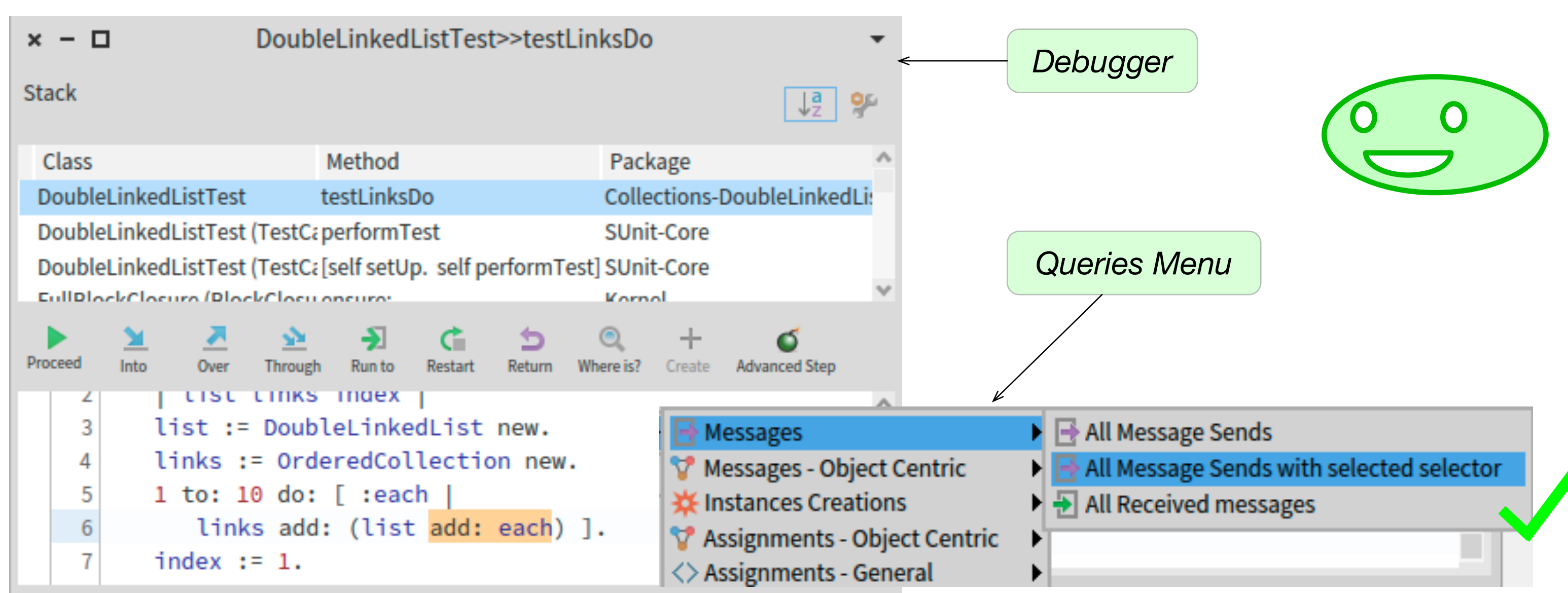
Missed the target! 😞

Solution: Time-Traveling Queries

Time-Traveling Queries (TTQs)

Do you have a debugging question?

Select a Time-Traveling Query from the Queries Menu!



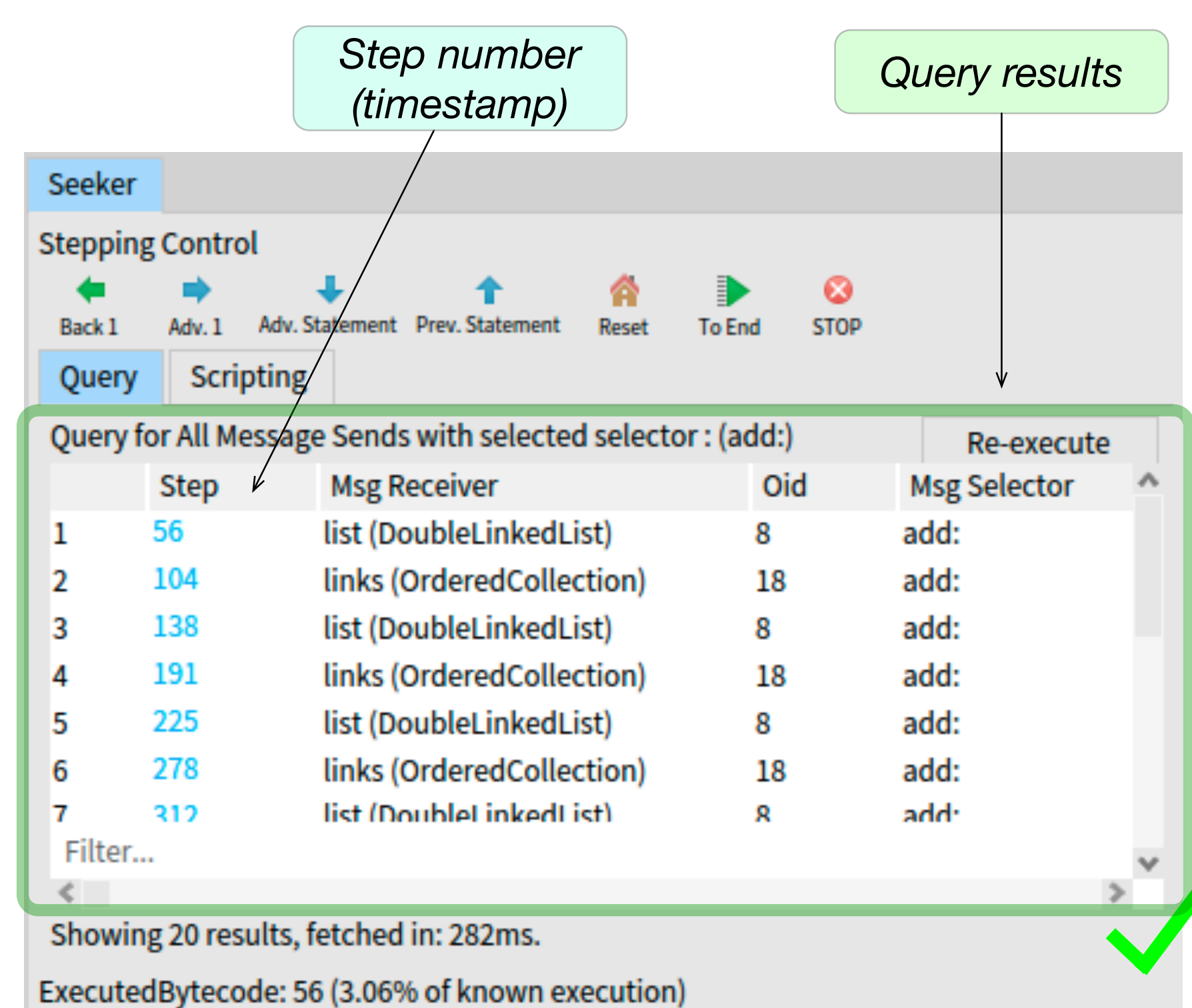
- TTQs request information of an execution related to **common debugging questions**.

- **Find answers directly** in your query result (Don't miss target!).

- **"Click & Time-Travel"**

Reverse or advance the execution **jumping** directly to any of the results timestamp (Less tedious!).

- **Explore your execution states forward or backward.**



Do you have another debugging question? Just select another query!

There is no query for your debugging question? Write your own TTQ!

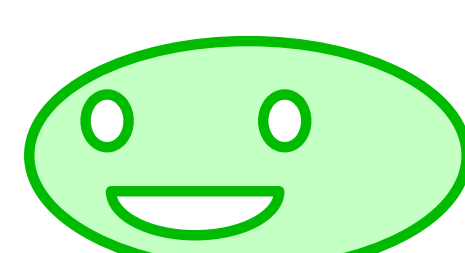
Example:

```
"Lists global variables assignments"
^ UserTTQ from: programStates
  select: [ :state | state isAssignment and: [ state node variable isGlobalVariable ] ]
  collect: [ :state |
    ResultItem new
      bytecodeIndex: state bytecodeIndex;
      variableName: state assignmentVariableName;
      yourself ]
```

From where to extract the data?

What program states are relevant?

What should be included in the results?



Time-Traveling Queries Evaluation

With TTQs, developers perform program comprehension tasks more accurately, faster, and with less effort than with standard debugging tools*.

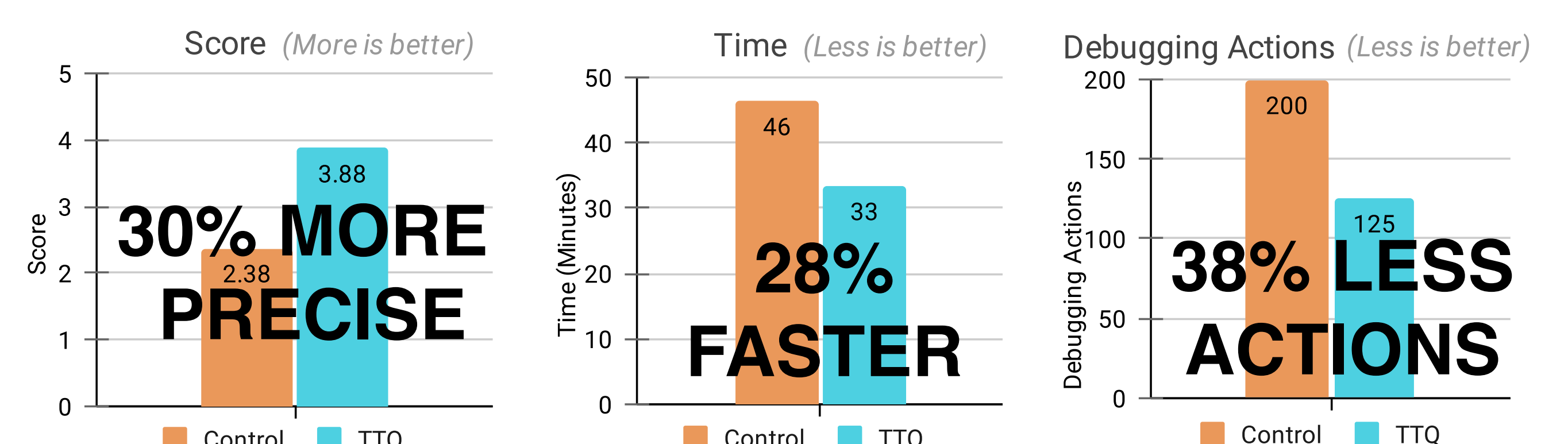
Controlled Experiment

- Repeated Measures Design (Within-subject)
- **34 Participants.**

Research Question

Do TTQs improve program comprehension tasks of participants regarding precision, time spent and efforts? (vs. using standard debugging tools)

Results



Control: Without Time-Traveling Queries

TTQ: Using Time-Traveling Queries

(*) M. Willembrinck, S. Costiou, A. Etien and S. Ducasse, "Time-Traveling Debugging Queries: Faster Program Exploration," *2021 IEEE 21st International Conference on Software Quality, Reliability and Security (QRS)*, 2021, pp. 642-653.

Future work

- Scaling the solution to daily debugging scenarios.
- Studying new relevant queries.
- Research TTQs generalization to different programming languages.
- Designing new TTQs-based debugging tools.