Automated test generation for Web-based APIs

SeTTS: Session Type Test Synthesis

*joint work with **Adrian Francalanza, Emilio Tuosto, and Alceste Scalas***

Christian Bartolo Burlo · 23/06/23

Outline of tool and approach

Tool demo

Results + Conclusion

Outline of tool and approach

Tool demo

Results + Conclusion

Outline of tool and approach

Tool demo

Results + Conclusion

Why should we test APIs?

Web-based APIs

Why we should test them

- Over **80%** of all web traffic is API usage
- ***2018 —*** Facebook API bug
- ***2019 —*** Wells Fargo API outage
- ***2022 —*** FreeHour API bug

These incidents highlight the need for comprehensive testing to identify and mitigate vulnerabilities in API endpoints.

We focus on REST APIs. Why?

70% of all public APIs are REST APIs.

Existing documentation frameworks, such as OpenAPI, GraphQL.

Understanding REST

A brief introduction

Method URL

- **POST** /api/owner.....*Creates a new pet owner.*
- **GET** /api/owner/{id}.....*Retrieve owner with ID.*
- **PUT** /api/owner/{id}.....*Update owner.*
- **DELETE** /api/owner/{id}.....*Delete the user.*

Web-based APIs

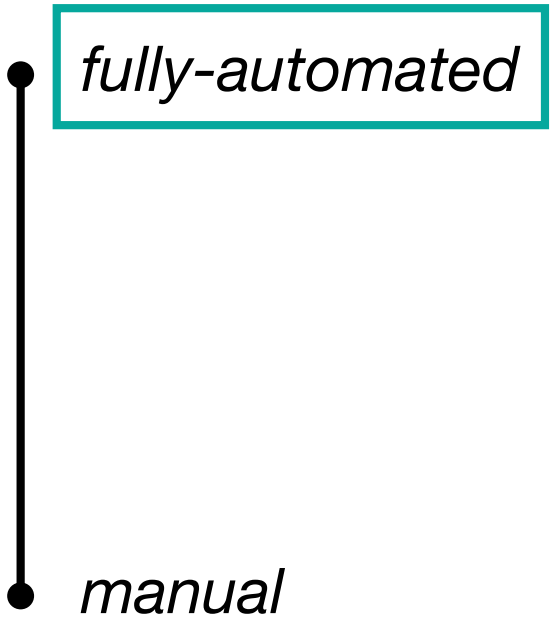
Challenges of testing REST APIs

- Request parameter generation and dependencies
- Validating responses: response code and body
- API call dependencies
- Request/response formatting
- Error reporting/reproducibility

- **POST** /api/owner.....*Creates a new pet owner.*
- **GET** /api/owner/{id}.....*Retrieve owner with ID.*
- **PUT** /api/owner/{id}.....*Update owner.*
- **DELETE** /api/owner/{id}.....*Delete the user.*

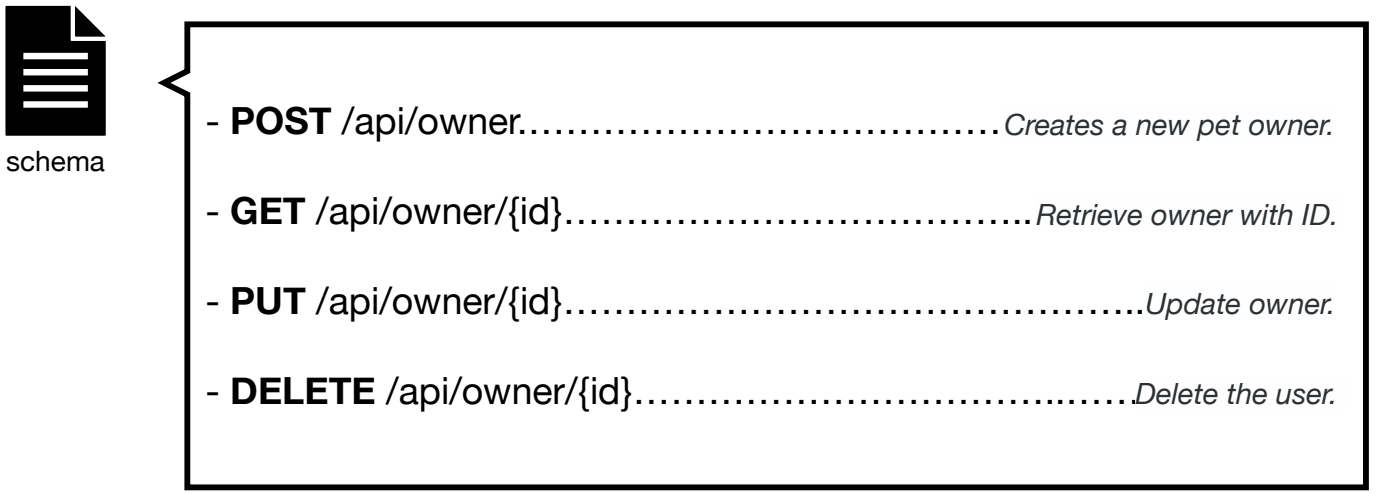
Current testing approaches

The state-of-the-art



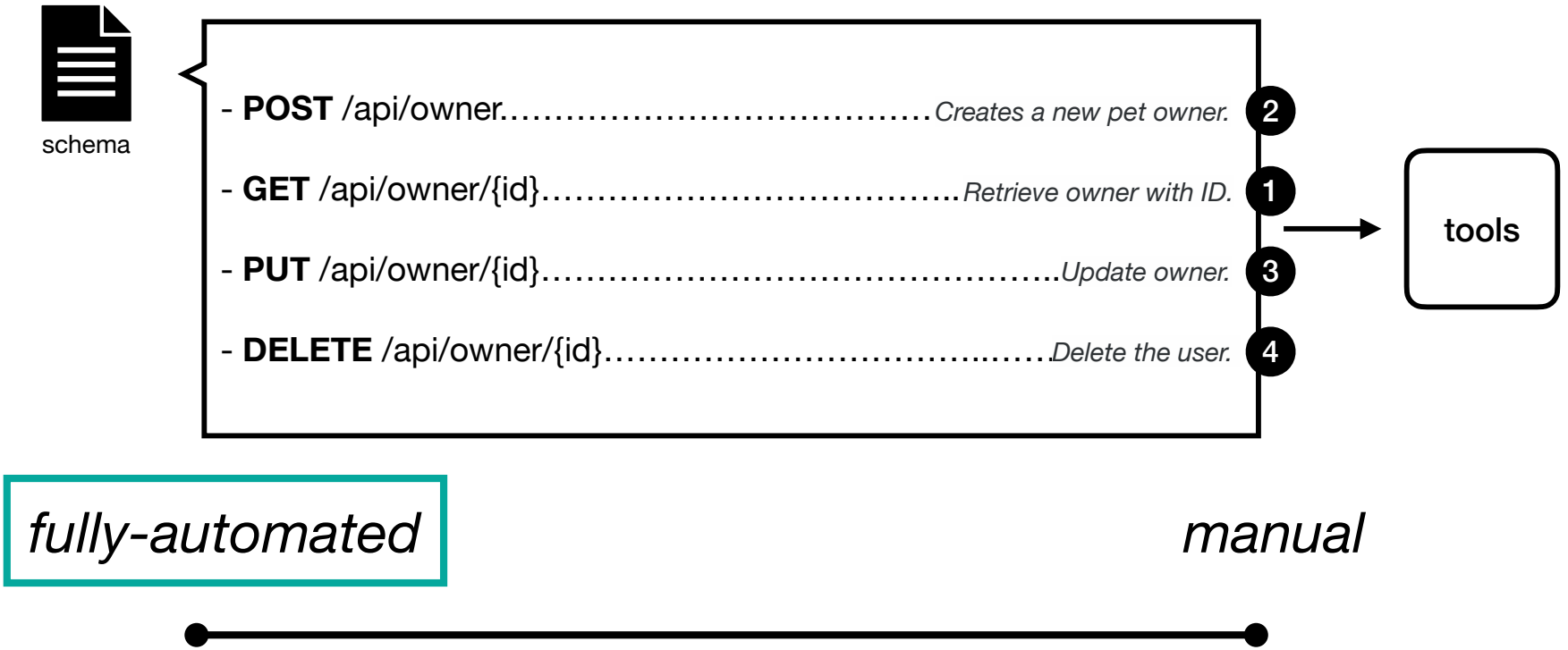
Current testing approaches

Fully-automated REST API testing



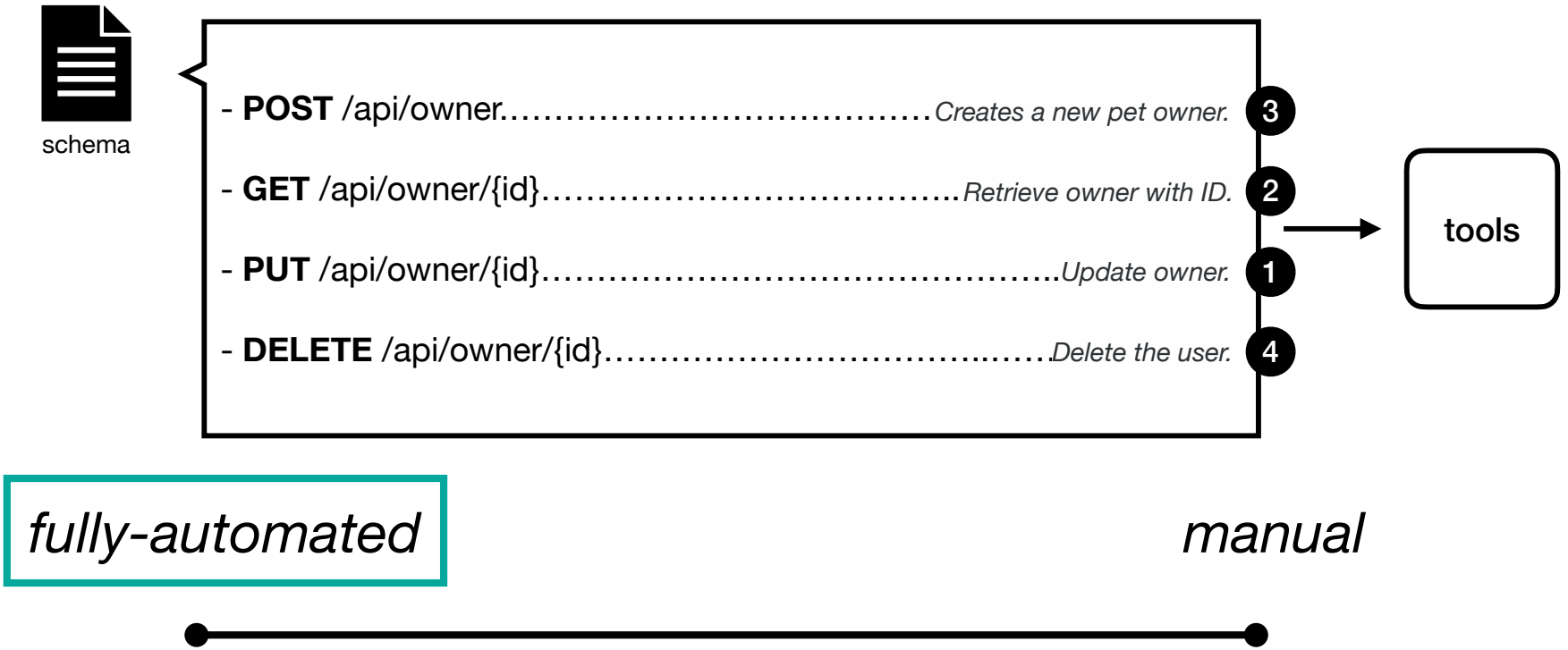
Current testing approaches

Fully-automated REST API testing



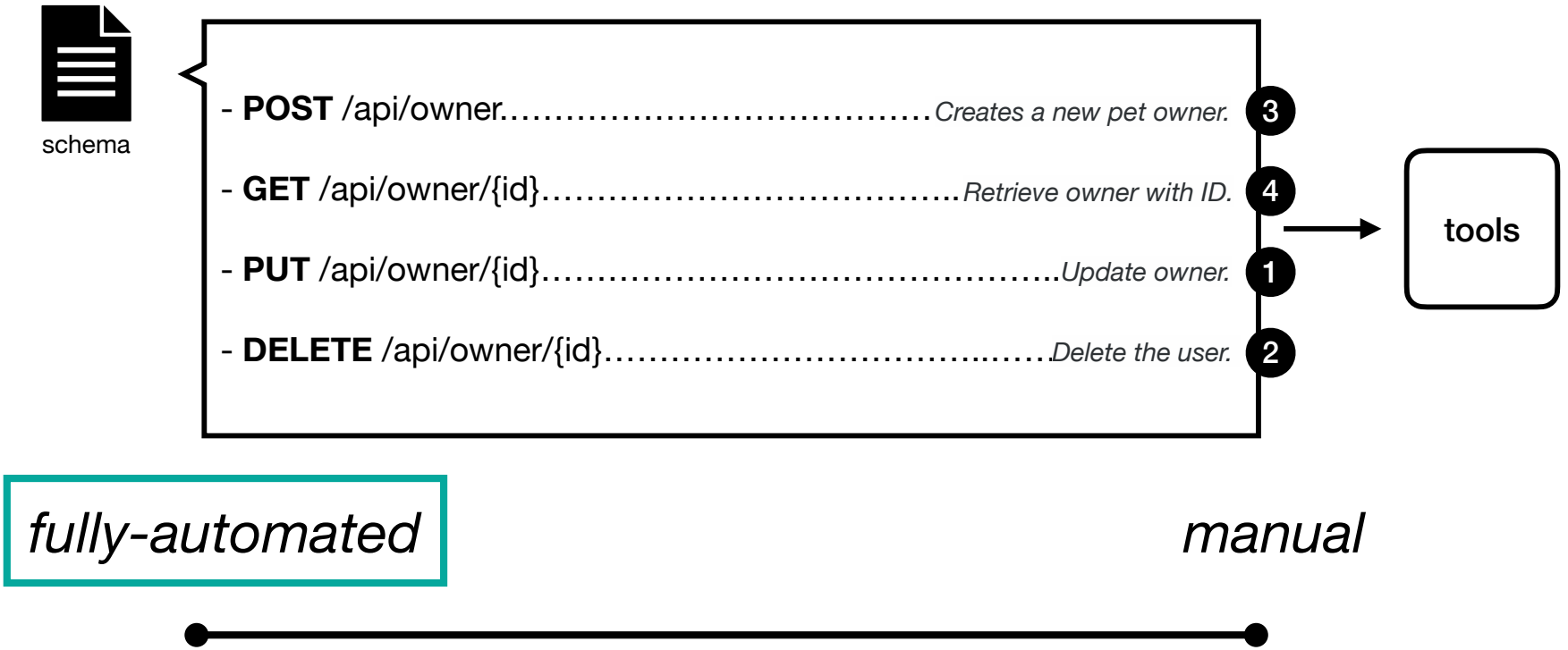
Current testing approaches

Fully-automated REST API testing



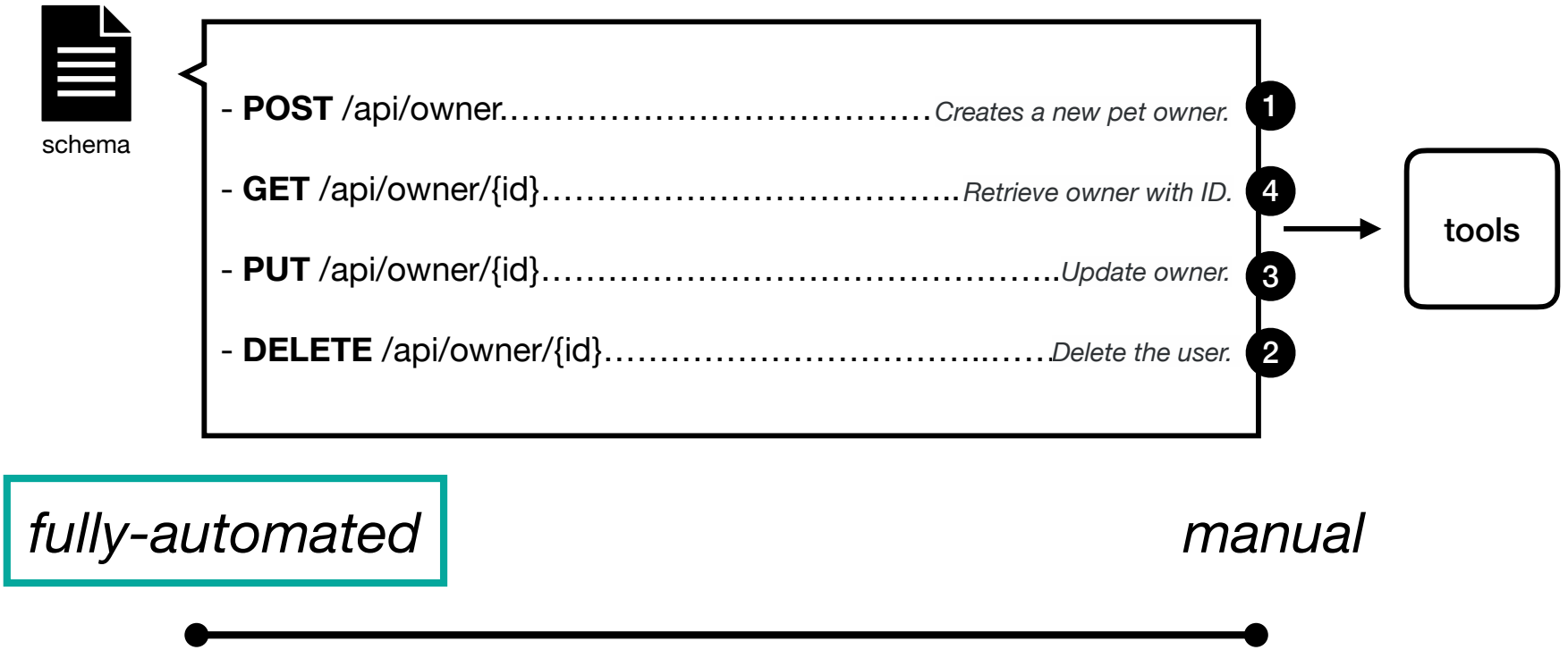
Current testing approaches

Fully-automated REST API testing



Current testing approaches

Fully-automated REST API testing



Current testing approaches

Fully-automated REST API testing

- Very inefficient use of resources, a single test run can take a couple of hours.
- Shooting in the dark, as one cannot tailor tests to specific needs.
- Limited in what sort of errors they are able to detect.
- However, it's fully automated, i.e., push button technology.

Current testing approaches

Manual tests

- Well, they're manual tests.
- Time consuming, error-prone and difficult to maintain.
- However, they can be tailored for each particular case.
- This ensures coverage of particular cases, as needed by tester.

Current testing approaches

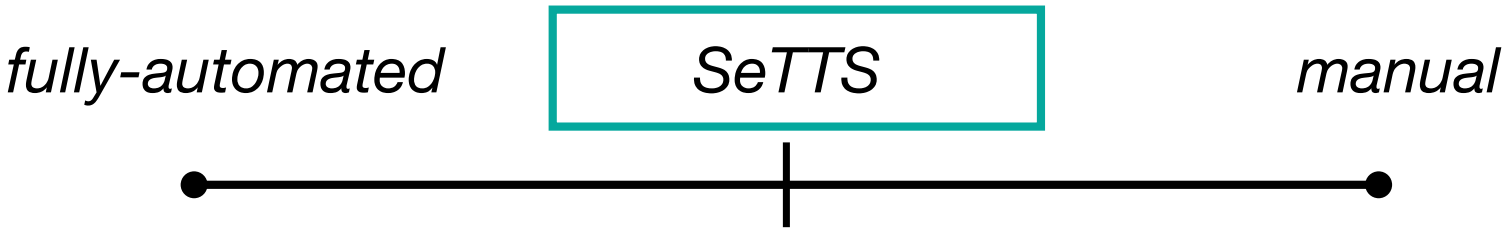
Manual tests

What if we want the efficiency of automated tests
but the coverage of manually-written tests?

Session Type Test Synthesis

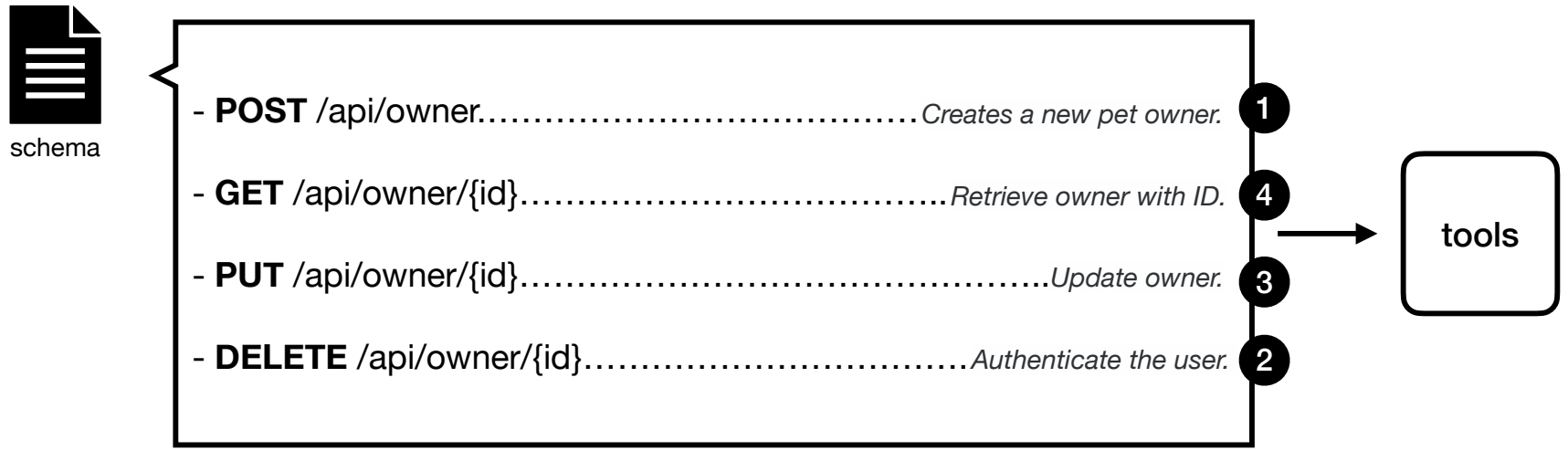
A model-based testing approach

- We attempted to strike a balance between fully-automated and manually written tests.
- By building models of the intended communication protocol, we can direct automated tests.



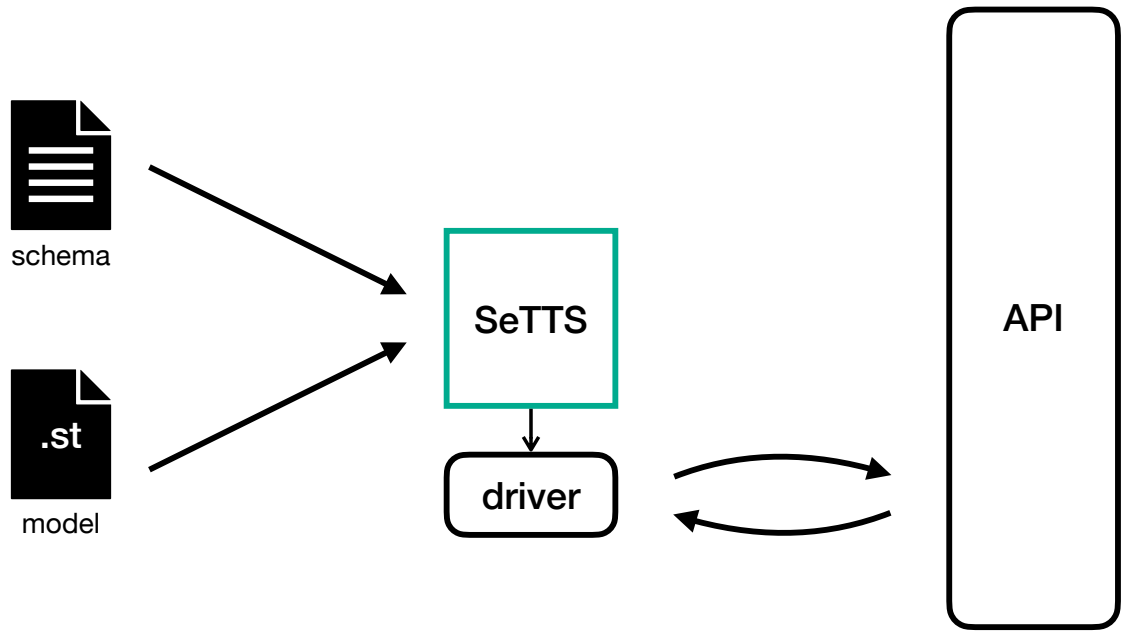
Session Type Test Synthesis

A model-based testing approach



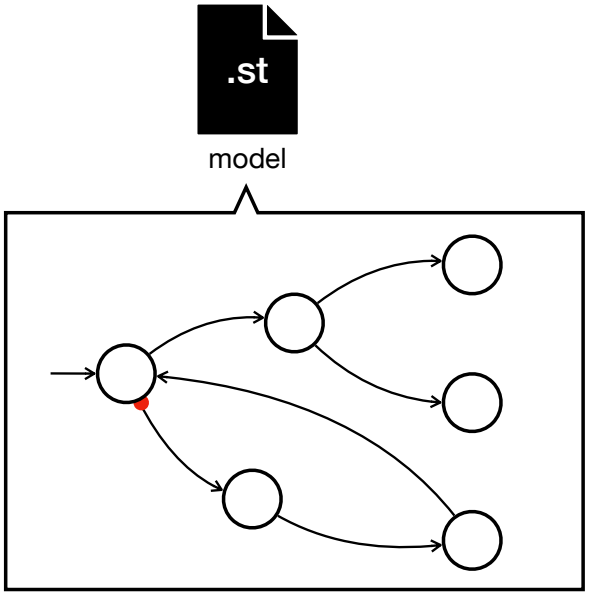
Our approach

Model-based testing



Our approach

Model-based testing

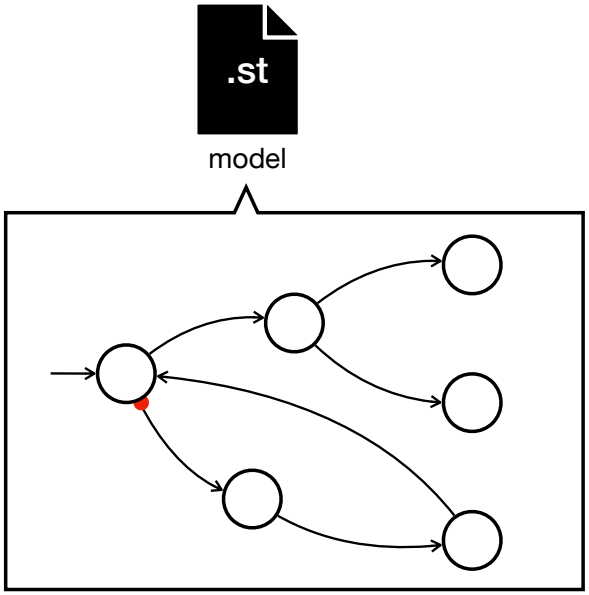


driver

- **POST** /api/owner..... *Creates a new pet owner.*
- **GET** /api/owner/{id}..... *Retrieve owner with ID.*
- **PUT** /api/owner/{id}..... *Update owner.*
- **DELETE** /api/owner/{id}..... *Authenticate the user.*

Our approach

Model-based testing

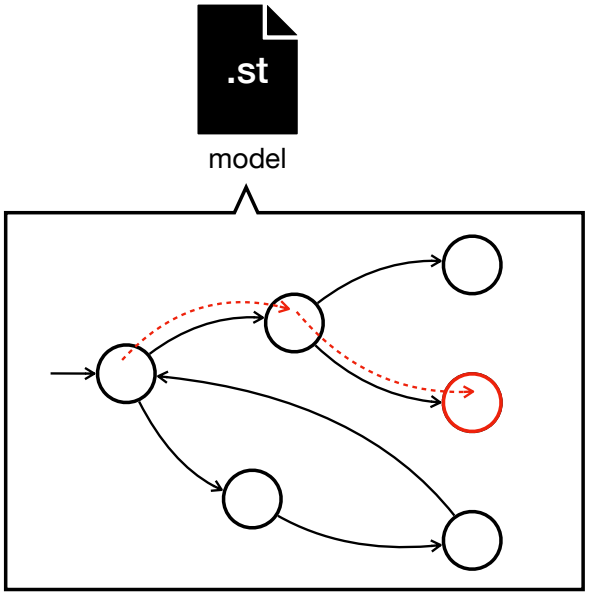


driver

- **POST** /api/owner..... *Creates a new pet owner.*
- **GET** /api/owner/{id}..... *Retrieve owner with ID.*
- **PUT** /api/owner/{id}..... *Update owner.*
- **DELETE** /api/owner/{id}..... *Authenticate the user.*

Our approach

Model-based testing



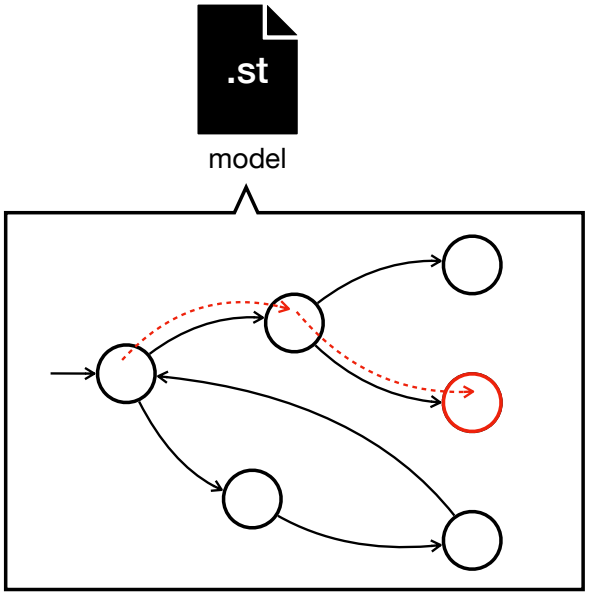
driver

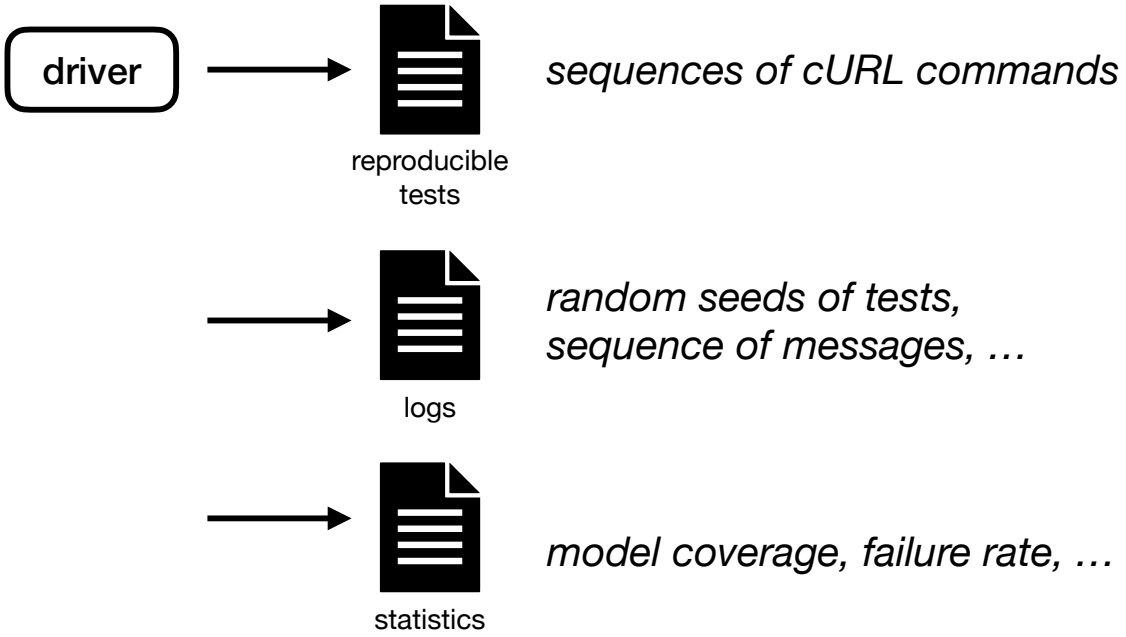
- response ***code***
- response ***body***
- ***assertions*** on payload data

- **POST** /api/owner.....*Creates a new pet owner.*
- **GET** /api/owner/{id}.....*Retrieve owner with ID.*
- **PUT** /api/owner/{id}.....*Update owner.*
- **DELETE** /api/owner/{id}.....*Authenticate the user.*

Our approach

Model-based testing





Questions

Tool demo

https://chrisbartoloburlo.github.io/cots/

Conclusion

SeTTS Analysis

Quantitative aspects

Application	*SeTTS line cov.*	*Manual line cov.*	*Fully-auto line cov.*	*App LOC*
RestCountries	**1722**	896		2409
GestaoHospital	**2857**	2532		4427
PetClinic	3099	**3127**		10,416
UsersRegisty	**2035**	1906		5452
FeaturesService	**1626**	1576	360	2026
LanguageTool	**4999**		935	18,053
PetStore	**1987**		763	3693

SeTTS Analysis

Qualitative aspects

- Request parameter generation and dependencies
- Validating responses: response code and body
- API call dependencies
- Request/response formatting
- Error reporting/reproducibility

```
!CollectionApi.getCollection(id: Int(genId)).
  ?C200(collections: Seq[Collection])
    <util.checkCollections(collections)>
```

SeTTS Analysis

Qualitative aspects

- Request parameter generation and dependencies
- Validating responses: response code and body
- API call dependencies
- Request/response formatting
- Error reporting/reproducibility

```
!CollectionApi.getCollection(id: Int(genId)).
  ?C200(collections: Seq[Collection])
    <util.checkCollections(collections)>
```

SeTTS Analysis

Qualitative aspects

- Request parameter generation and dependencies
- Validating responses: response code and body
- API call dependencies
- Request/response formatting
- Error reporting/reproducibility

```
!CollectionApi.getCollection(id: Int(genId)).
  ?C200(collections: Seq[Collection])
    <util.checkCollections(collections)>
```

SeTTS Analysis

Qualitative aspects

- Request parameter generation and dependencies
- Validating responses: response code and body
- API call dependencies
- Request/response formatting
- Error reporting/reproducibility

```
!CollectionApi.getCollection(id: Int(genId)).
  ?C200(collections: Seq[Collection])
    <util.checkCollections(collections)>
```

SeTTS Analysis

Qualitative aspects

- Request parameter generation and dependencies
- Validating responses: response code and body
- API call dependencies
- Request/response formatting
- Error reporting/reproducibility

```
!CollectionApi.getCollection(id: Int(genId)).
  ?C200(collections: Seq[Collection])
    <util.checkCollections(collections)>
```