Are the Methods in Your DAOs in the Right Place?

A Preliminary Study

Maurício Aniche
Gustavo Oliva
Marco Aurélio Gerosa

What are

DAOs ?

```
public class InvoiceDAO {

    @Inject private EntityManager em;

    public List<Invoice> getAll() {
        String sql = "select …
                     from Invoice
                     where ...";

        return em.createQuery(sql).getResultList();
    }

     public void save(Invoice inv) {
        em.save(inv);
     }
}
```

They can get complicated!

```
SELECT
    p.name as project,
    c.id as commitId,
    a.name as artifactName,
    a.path as artifactPath
FROM
    Projects p
JOIN
    Commits c ON c.project_id = p.id
JOIN
    Artifacts a on a.commit_id = c.id
WHERE
    p.repository = ’Apache’;
```

They can get complicated!

```
SELECT
    p.name as project,
    c.id as commitId,
    a.name as artifactName,
    a.path as artifactPath
FROM
    Projects p
JOIN
    Commits c ON c.project_id = p.id
JOIN
    Artifacts a on a.commit_id = c.id
WHERE
    p.repository = 'Apache';
```

Belongs to ProjectDAO?

... and more complicated!

```
SELECT
    p.name as project,
    c.id as commitId,
    a.name as artifactName,
    a.path as artifactPath
FROM
    Commits c
JOIN
    Projects p ON c.project_id = p.id
JOIN
    Artifacts a on a.commit_id = c.id
WHERE
    p.repository = 'Apache';
```

or CommitDAO?

... and more complicated!

```
SELECT
    p.name as project,
    c.id as commitId,
    a.name as artifactName,
    a.path as artifactPath
FROM
    Commits c
JOIN
    Projects p ON c.project_id = p.id
JOIN
    Artifacts a on a.commit_id = c.id
WHERE
    p.repository = 'Apache';
```

Where

to put this query?

finding the right place is

very problematic!

duplicated code

modularity violation = bugs

spend time searching

RQ:

How can one automatically identify methods that may have been placed in the wrong or in ambiguous DAOs

Research Design

Heuristic
Implement a Tool
Select Projects
Qualitative Analysis

the heuristic

basically looks to

the method signature!

```
public class InvoiceDAO {
    public Invoice findById(int id) { … }
    public ShoppingCart find(User u) { … })
    public List<Invoice> getAll() { … }
    public void save(Invoice inv) { ...}
}
```

the projects were...

Project	# of classes	# of commits	# of DAOs	# of methods
Gnarus	924	10451	39	233
Caelumweb	1321	12077	81	590
Codesheriff	56	339	10	70

java VRaptor unit tested

web MVC Hibernate

and the heuristic found out...

Project	# of DAO methods	# of right methods	# of wrong methods	% of wrong methods
Caelumweb	590	511	79	13.38%
Codesheriff	70	57	13	18.57%
Gnarus	233	200	33	14.16%

the developers agreed!

Project	# of inspected methods	# of agreement	% of agreement
Caelumweb	79	59	74.68%
Codesheriff	13	8	61.53%
Gnarus	33	16	48.48%

What

have we learned?

A quick and cheap approach to identify methods in wrong DAOs

Filters 13% to 18% of all methods and
it is correct 50% to 75% of the cases

Threats to Validity

- Only three projects
- Point of view of a single developer
- We only validated the ones the heuristic has selected

What

are the next steps?

- Understand the cost of this TD
- Improve the heuristic

maurício aniche (aniche@ime.usp.br)
gustavo oliva (goliva@ime.usp.br)
marco gerosa (gerosa@ime.usp.br)