Migration to Microservices: A Comparative Study of Decomposition Strategies and Analysis Metrics

SERVICE COMPUTATION 2023

June 26th-30th

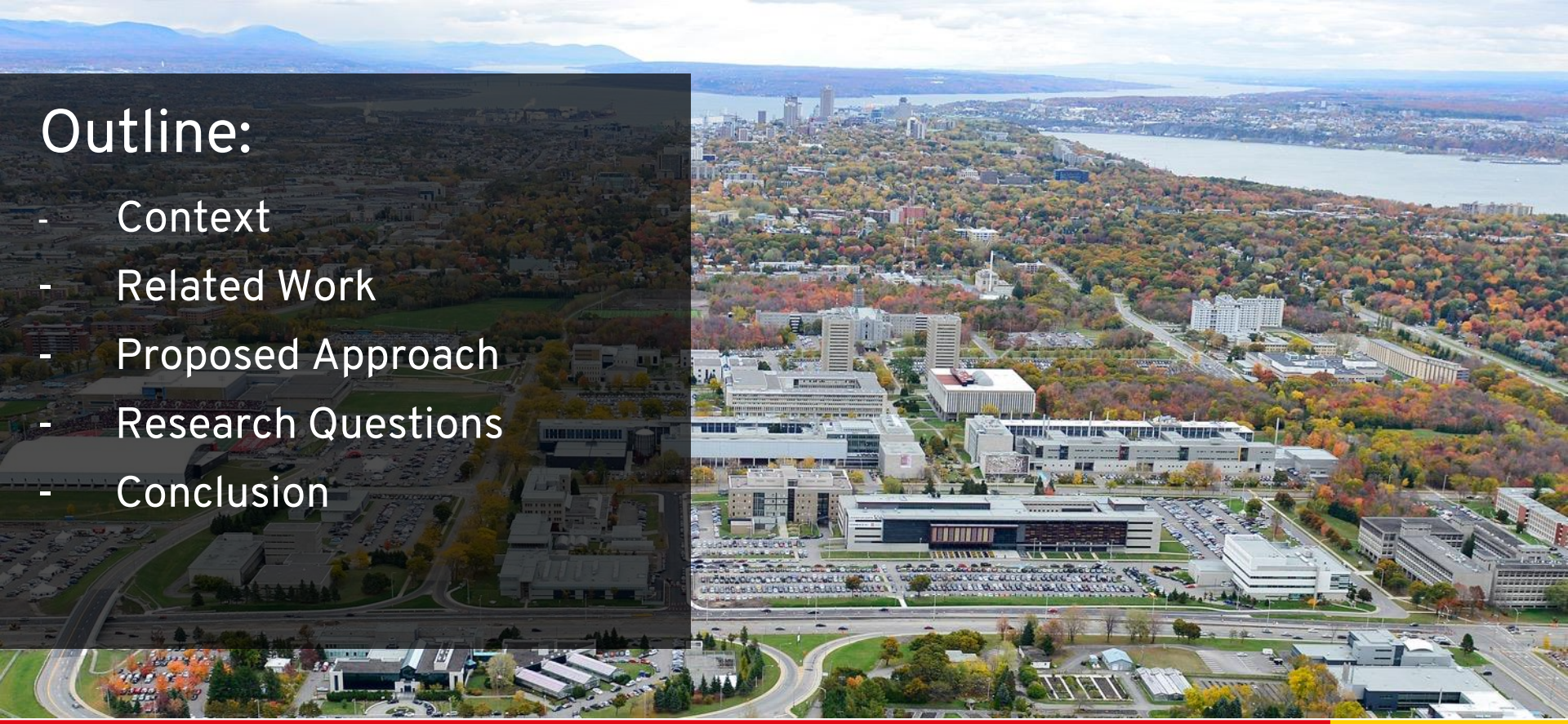
Nice, France

Meryam Chaieb *
Laval University

Khaled Sellami
Laval University

Mohamed Aymen Saied
Laval University

UNIVERSITÉ LAVAL



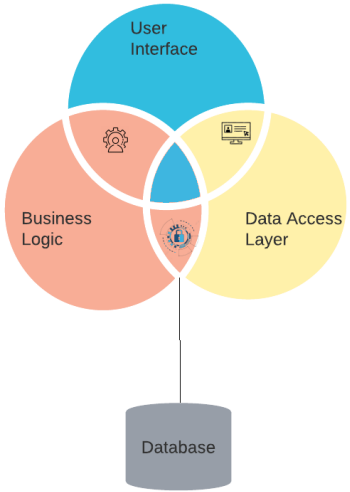
Outline:

- Context
- Related Work
- Proposed Approach
- Research Questions
- Conclusion

UNIVERSITÉ LAVAL

Context 1/2

Monolith architecture



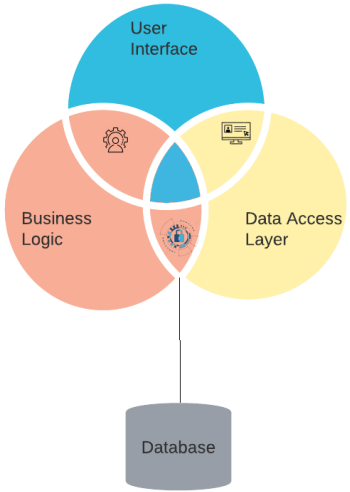
- ✗ Lack of Scalability
- ✗ Decreased Fault Isolation
- ✗ Limited Development Team Autonomy

Microservice architecture

UNIVERSITÉ LAVAL

Context 1/2

Monolith architecture

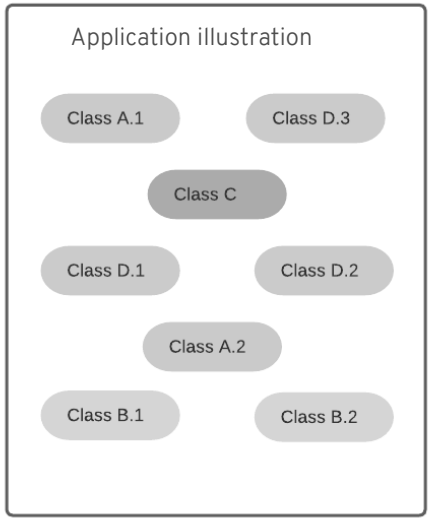


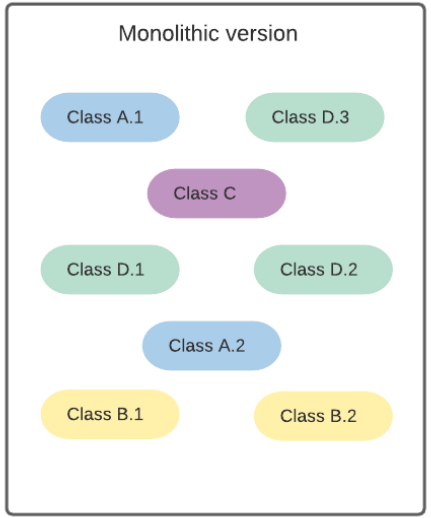
Microservice architecture

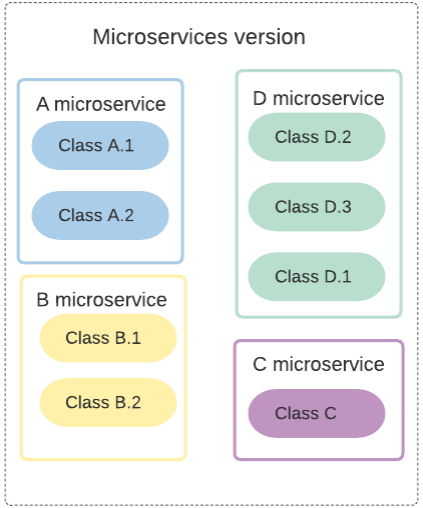
- ✓ Scalability and flexibility
- ✓ Fault Isolation and Resilience
- ✓ Team Autonomy

UNIVERSITÉ LAVAL

Context 2/2







Monolithic application

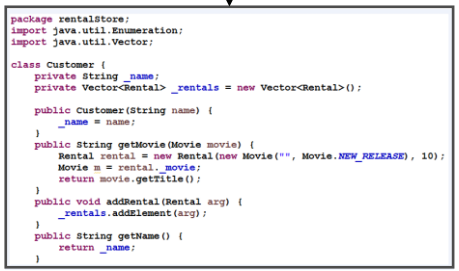
Decomposing process

Microservice application

Related Work 1/2

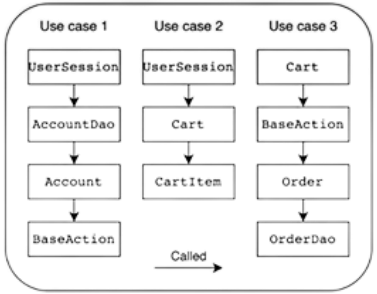
Input Type

Source code

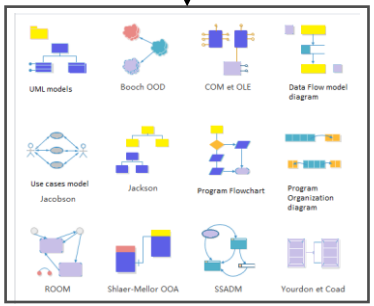


```
package rentalStore;
import java.util.Enumeration;
import java.util.Vector;

class Customer {
    private String _name;
    private Vector<Rental> _rentals = new Vector<Rental>();

    public Customer(String name) {
        _name = name;
    }
    public String getMovie(Movie movie) {
        Rental rental = new Rental(new Movie("", Movie.NEW_RELEASE), 10);
        Movie m = rental._movie;
        return movie.getTitle();
    }
    public void addRental(Rental arg) {
        _rentals.addElement(arg);
    }
    public String getName() {
        return _name;
    }
```

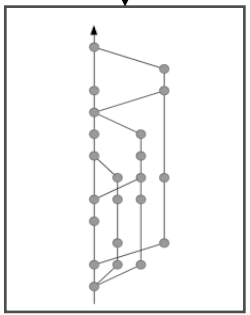
Execution trace



Diagrams



Git Commit History



Bunch B. S. Mitchell and S. Mancoridis, "On the evaluation of the bunch search-based software modularization algorithm," Soft Computing 2008

HierDecomp K. Sellami, M. A. Saied, and A. Ouni, "A hierarchical dbscan method for extracting microservices from monolithic applications," in The International Conference on Evaluation and Assessment in Software Engineering 2022

Mono2Micro K. Kalia, X. Jin, K. Rahul, S. Saurabh, V. Maja, and B. Debasish, "Mono2micro: A practical and effective tool for decomposing monolithic java applications to microservices," Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, 2021.

FOSCI W. Jin, T. Liu, Y. Cai, R. Kazman, R. Mo, and Q. Zheng, "Service candidate identification from monolithic systems based on execution traces," IEEE Transactions on Software Engineering2021

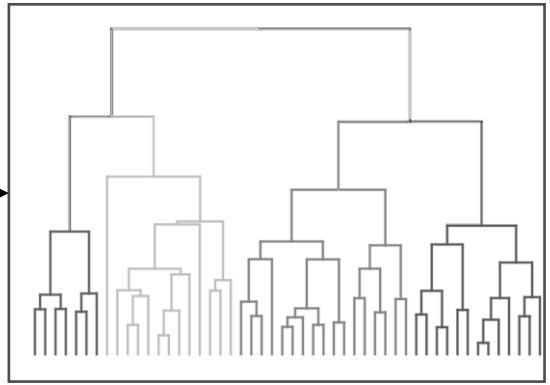
Service cutter
Michael Gysel, Lukas Kolbener, Wolfgang Giershe, Olaf Zimmermann, "Service Cutter: A Systematic Approach to Service Decomposition" LNPSE,2016

MEM G. Mazlami, J. Cito, and P. Leitner, "Extraction of microservices from monolithic software architectures," in 2017 IEEE International Conference on Web Services (ICWS)

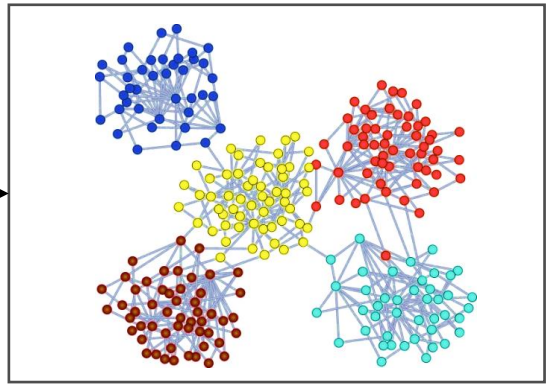
UNIVERSITÉ LAVAL

Related Work 2/2

Output type



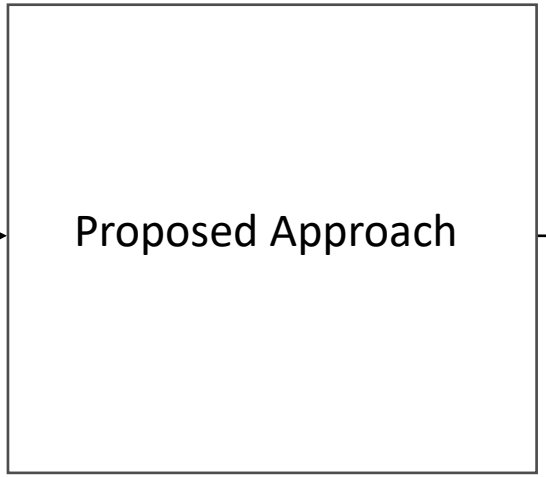
- HierDecomp
- HyDecomp
- FOSCI

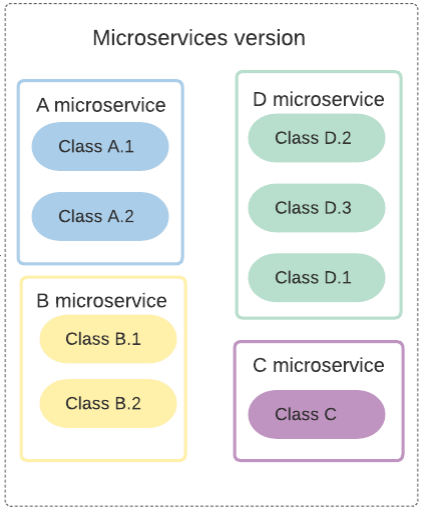


- Mono2Micro
- Service Cutter
- Bunch

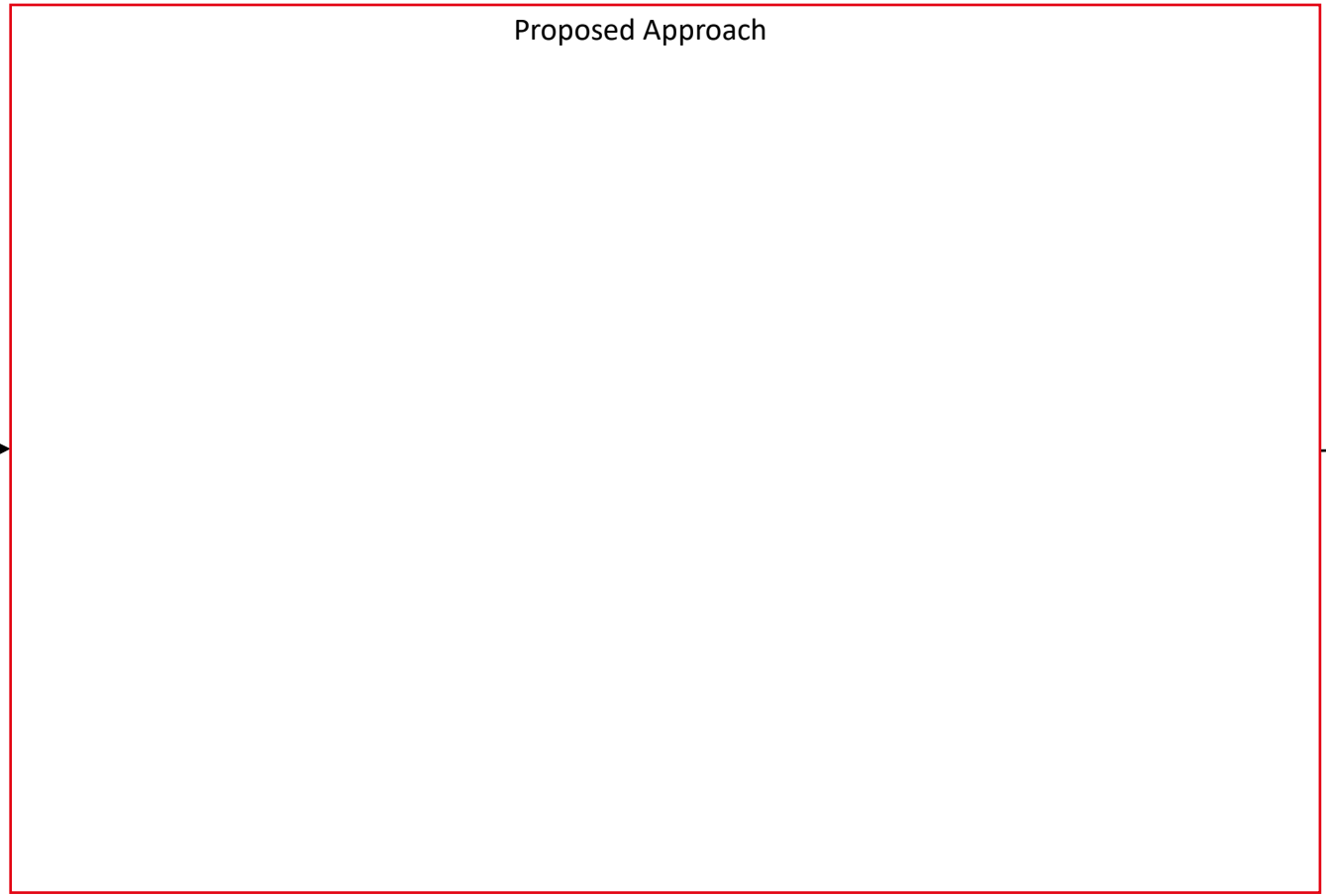
Proposed Approach 1/10

```
package rentalStore;
import java.util.Enumeration;
import java.util.Vector;

class Customer {
    private String _name;
    private Vector<Rental> _rentals = new Vector<Rental>();

    public Customer(String name) {
        _name = name;
    }
    public String getMovie(Movie movie) {
        Rental rental = new Rental(new Movie("", Movie.NEW_RELEASE), 10);
        Movie m = rental._movie;
        return movie.getTitle();
    }
    public void addRental(Rental arg) {
        _rentals.addElement(arg);
    }
    public String getName() {
        return _name;
    }
```

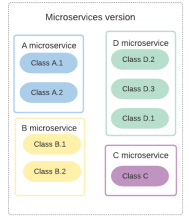




Proposed Approach 2/10

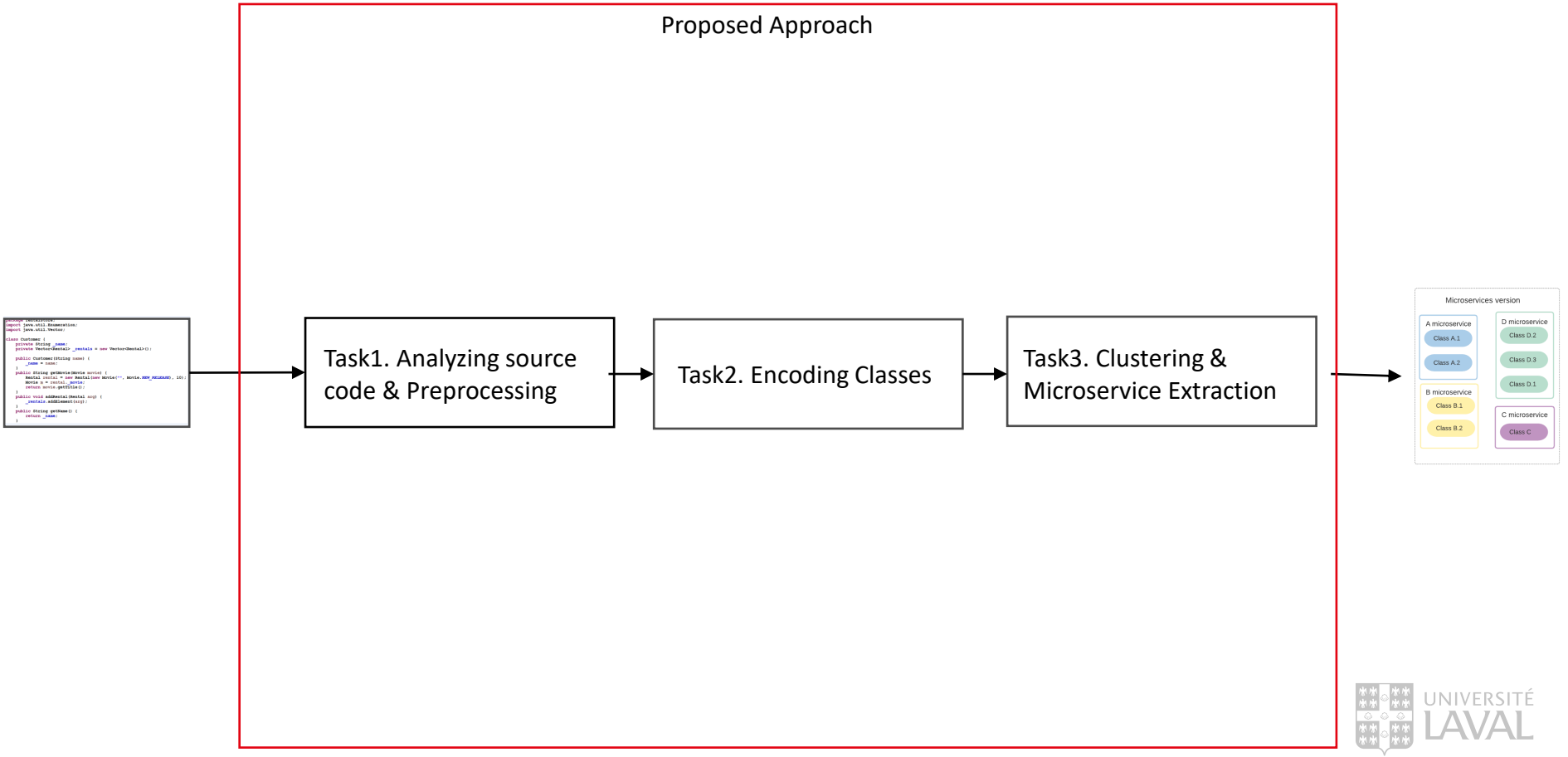



```
package rentalStore;
import java.util.Enumeration;
import java.util.Vector;

class Customer {
    private String _name;
    private Vector<Rental> _rentals = new Vector<Rental>();

    public Customer(String name) {
        _name = name;
    }
    public String getMovie(Movie movie) {
        Rental rental = new Rental(new Movie("", Movie.NEW_RELEASE) , 10);
        Movie m = rental._movie;
        return movie.getTitle();
    }
    public void addRental(Rental arg) {
        _rentals.addElement(arg);
    }
    public String getName() {
        return _name;
    }
```



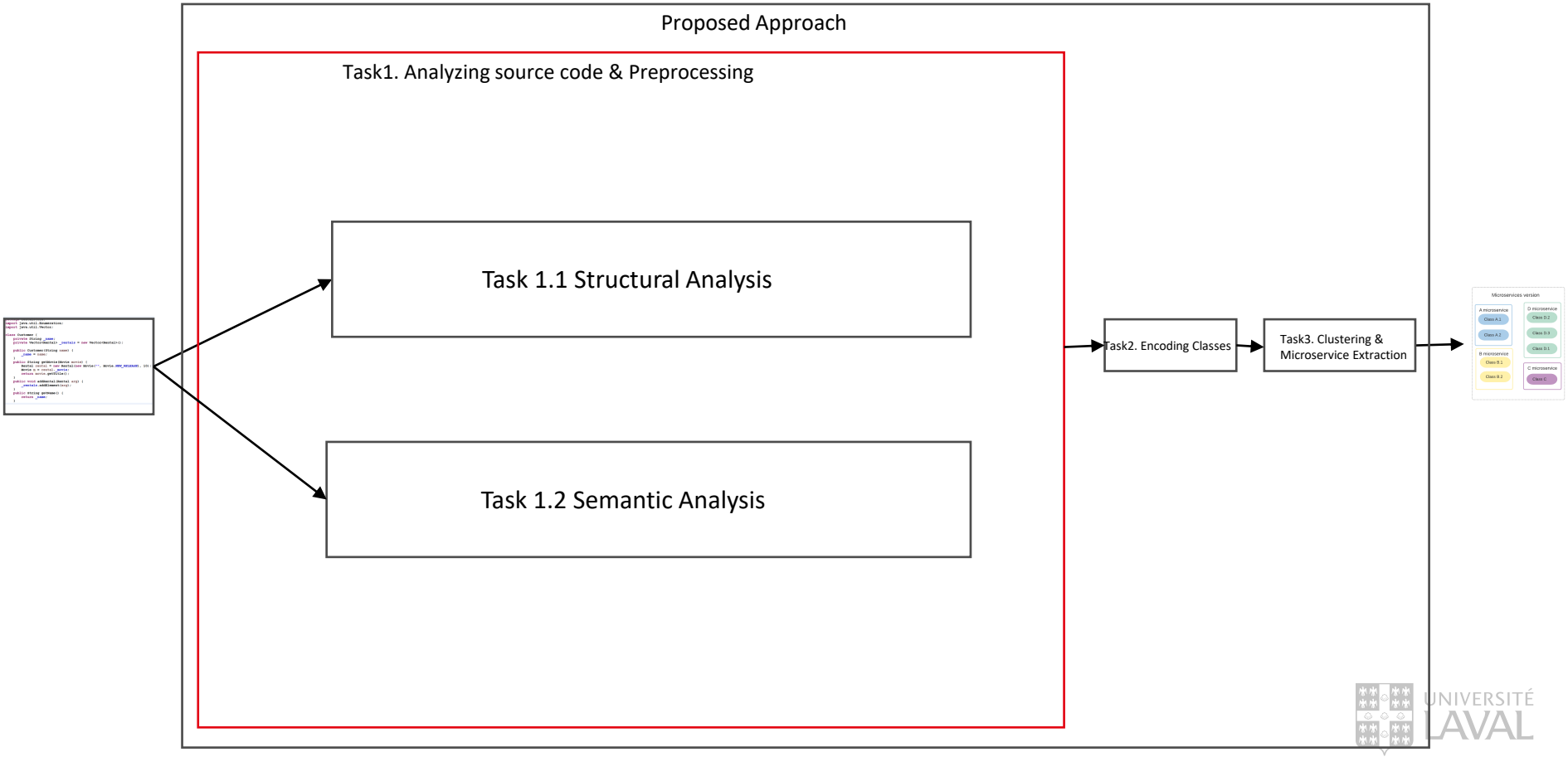
UNIVERSITÉ LAVAL

Proposed Approach 2/10

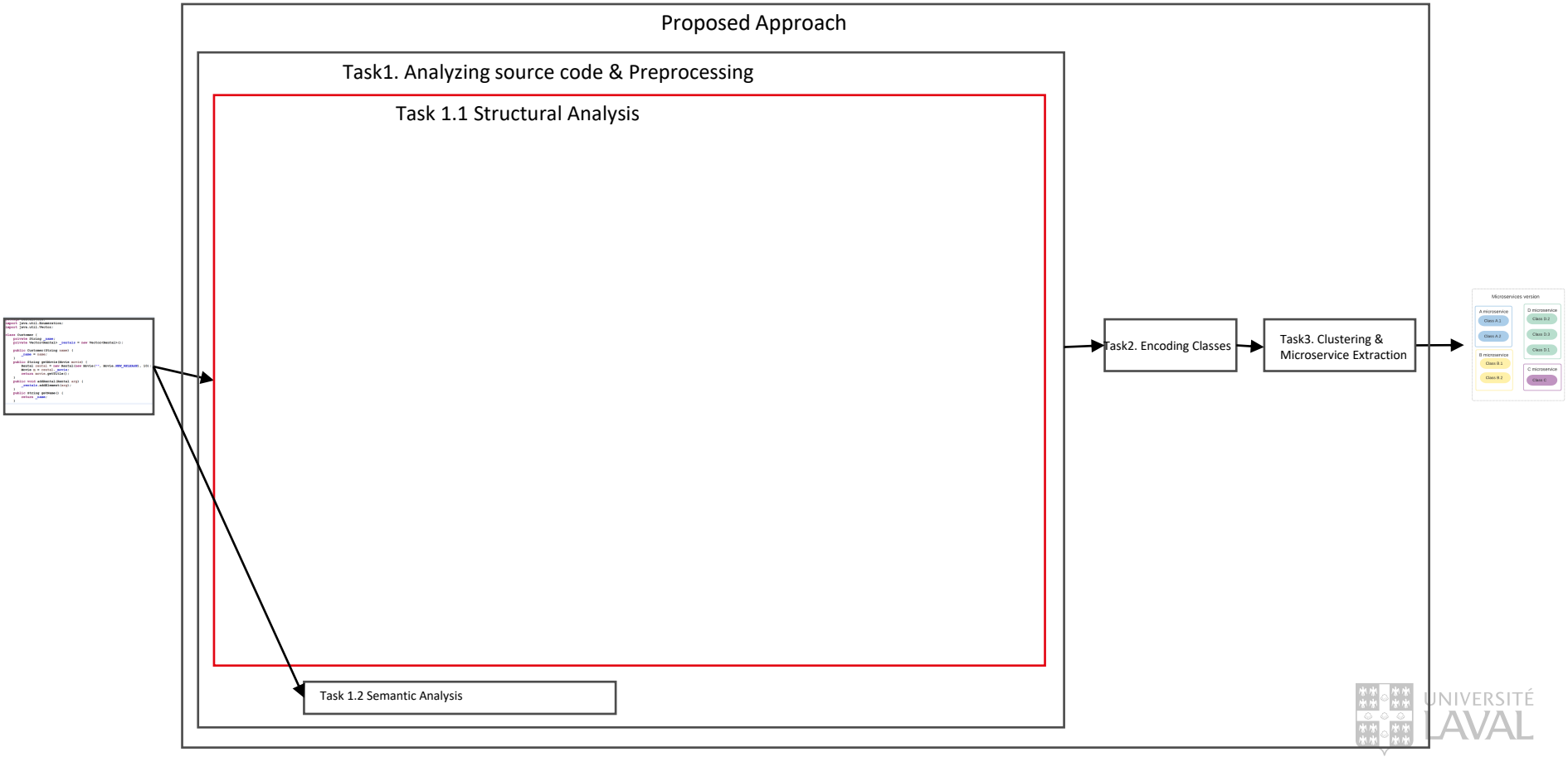


UNIVERSITÉ LAVAL

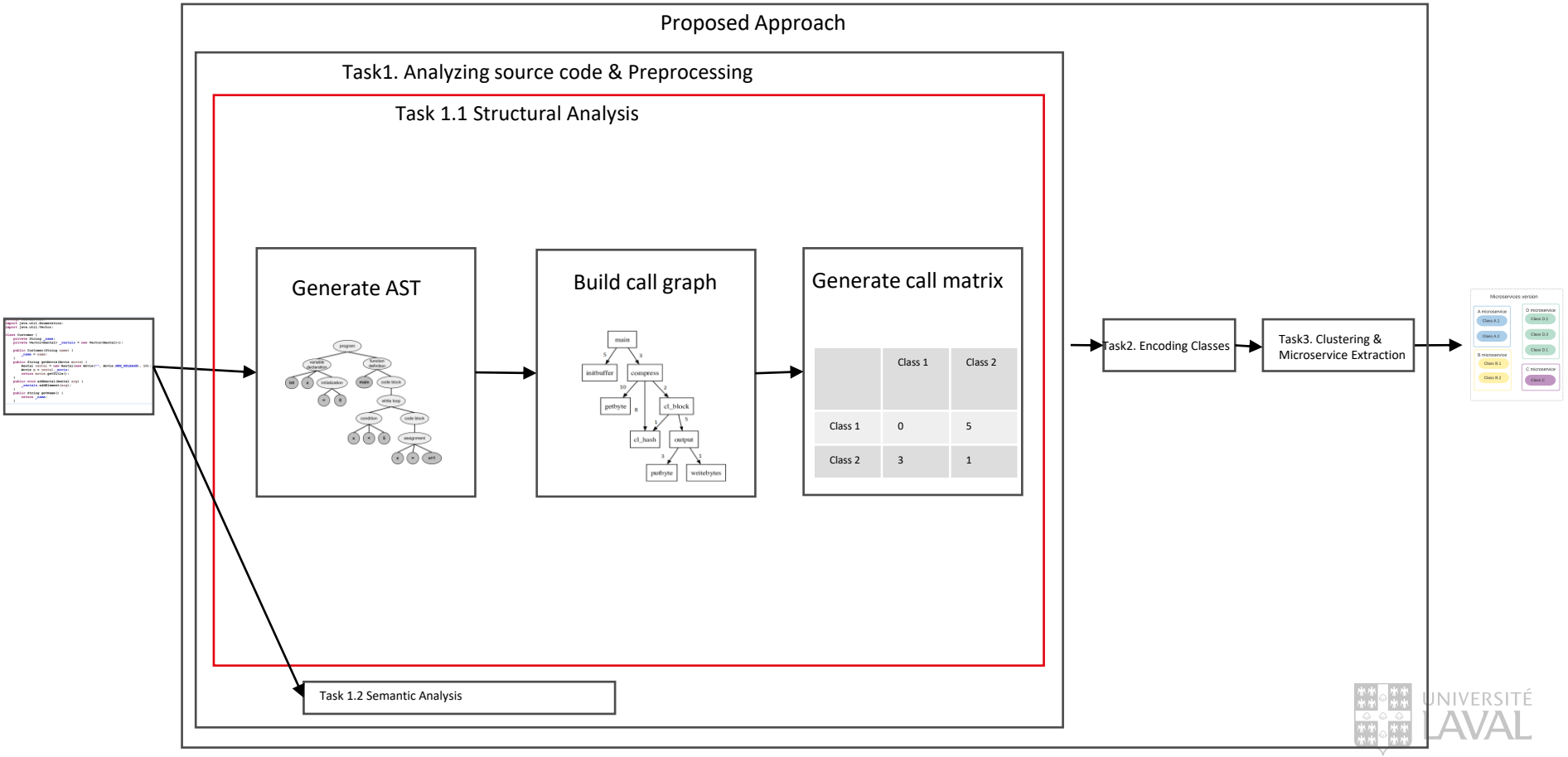
Proposed Approach 3/10



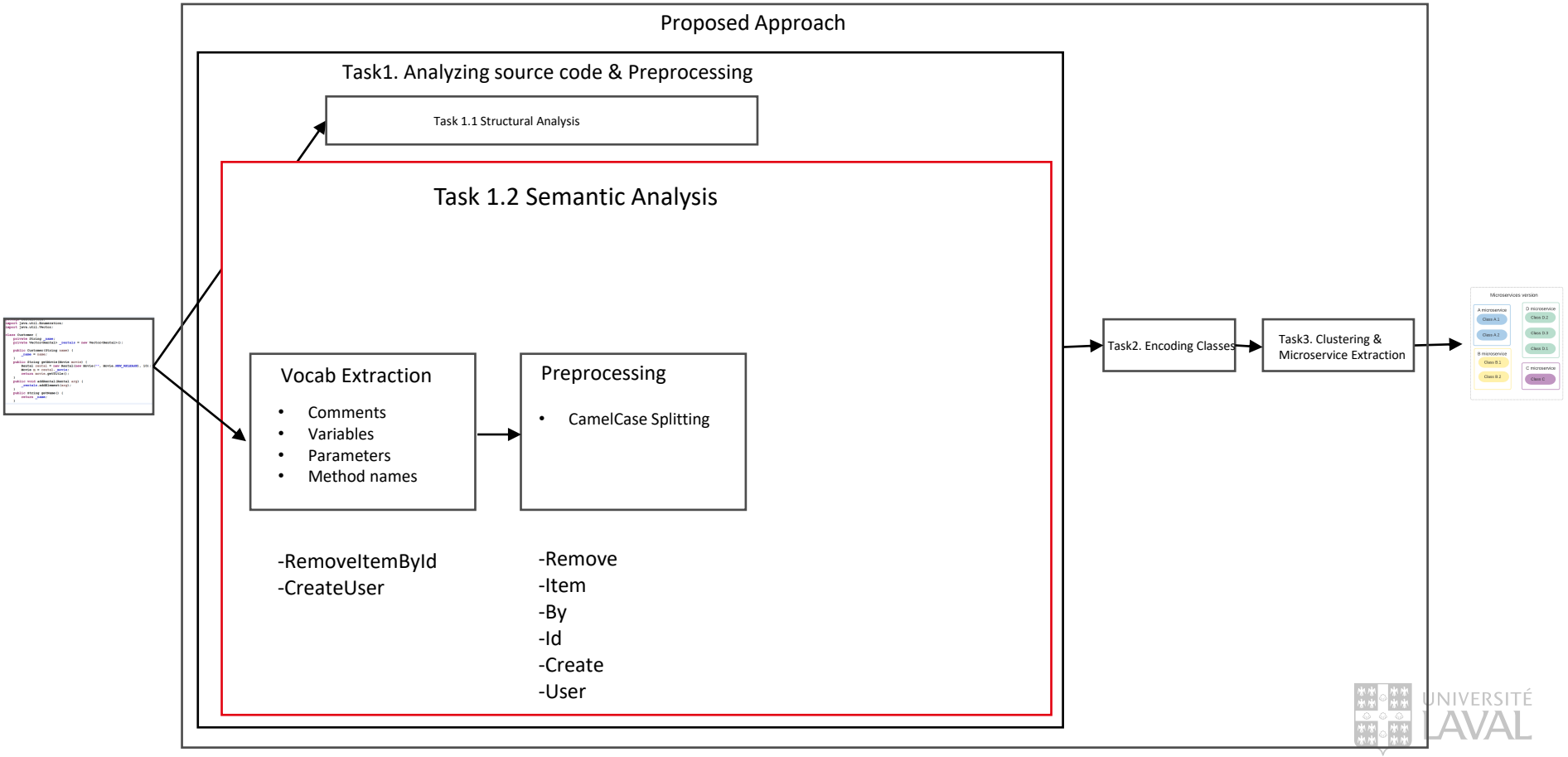
Proposed Approach 3/10



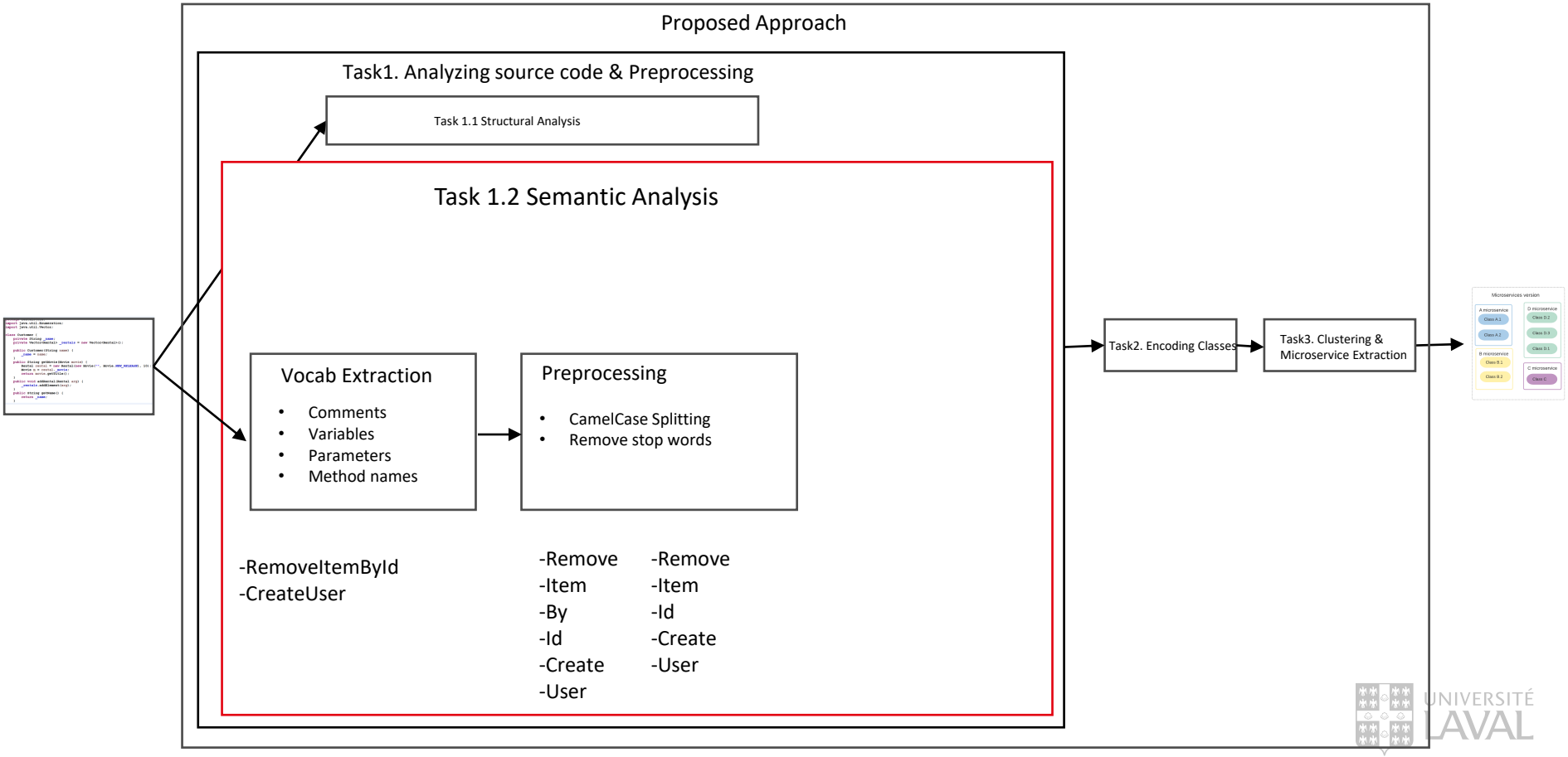
Proposed Approach 4/10



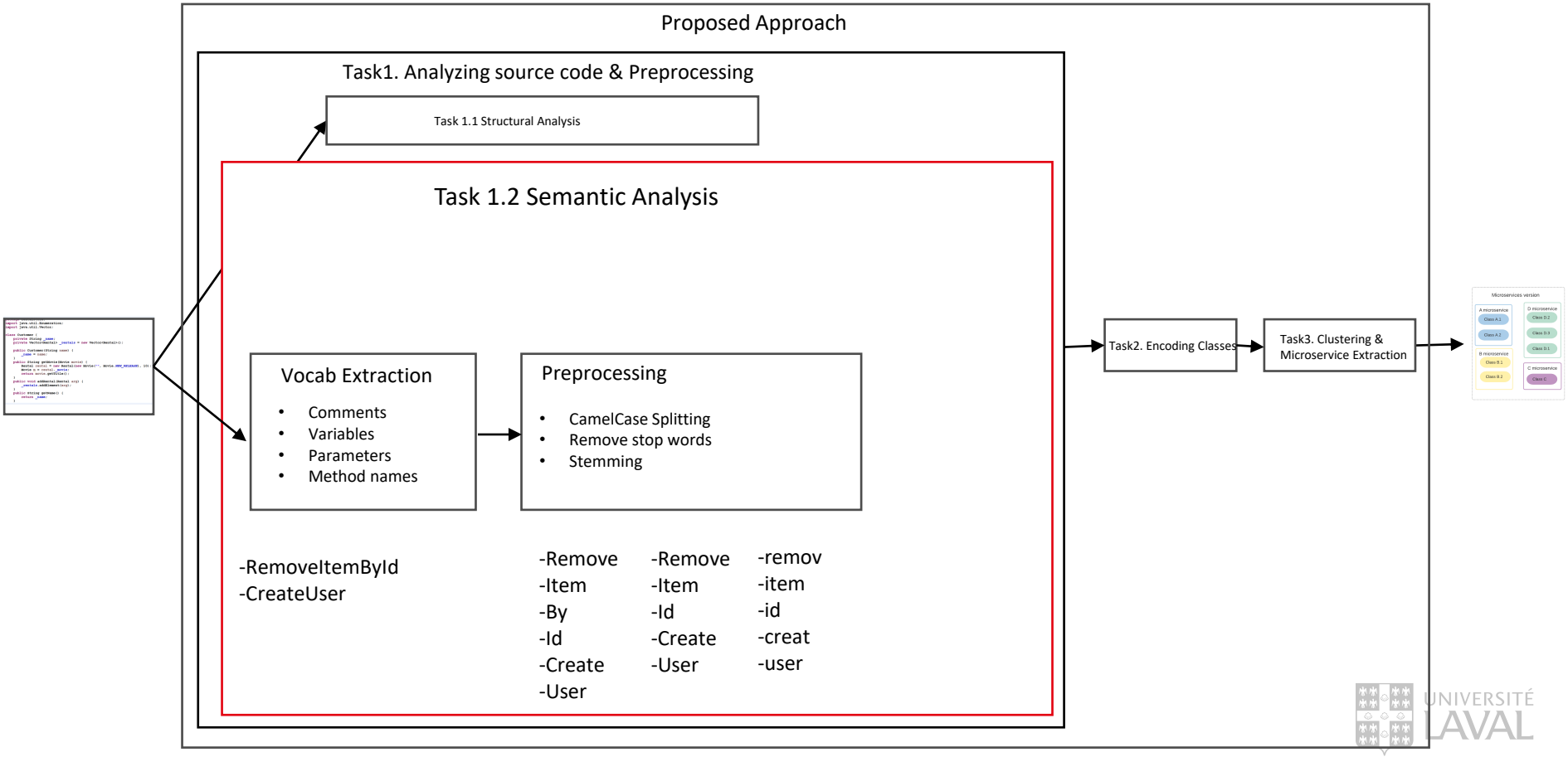
Proposed Approach 5/10



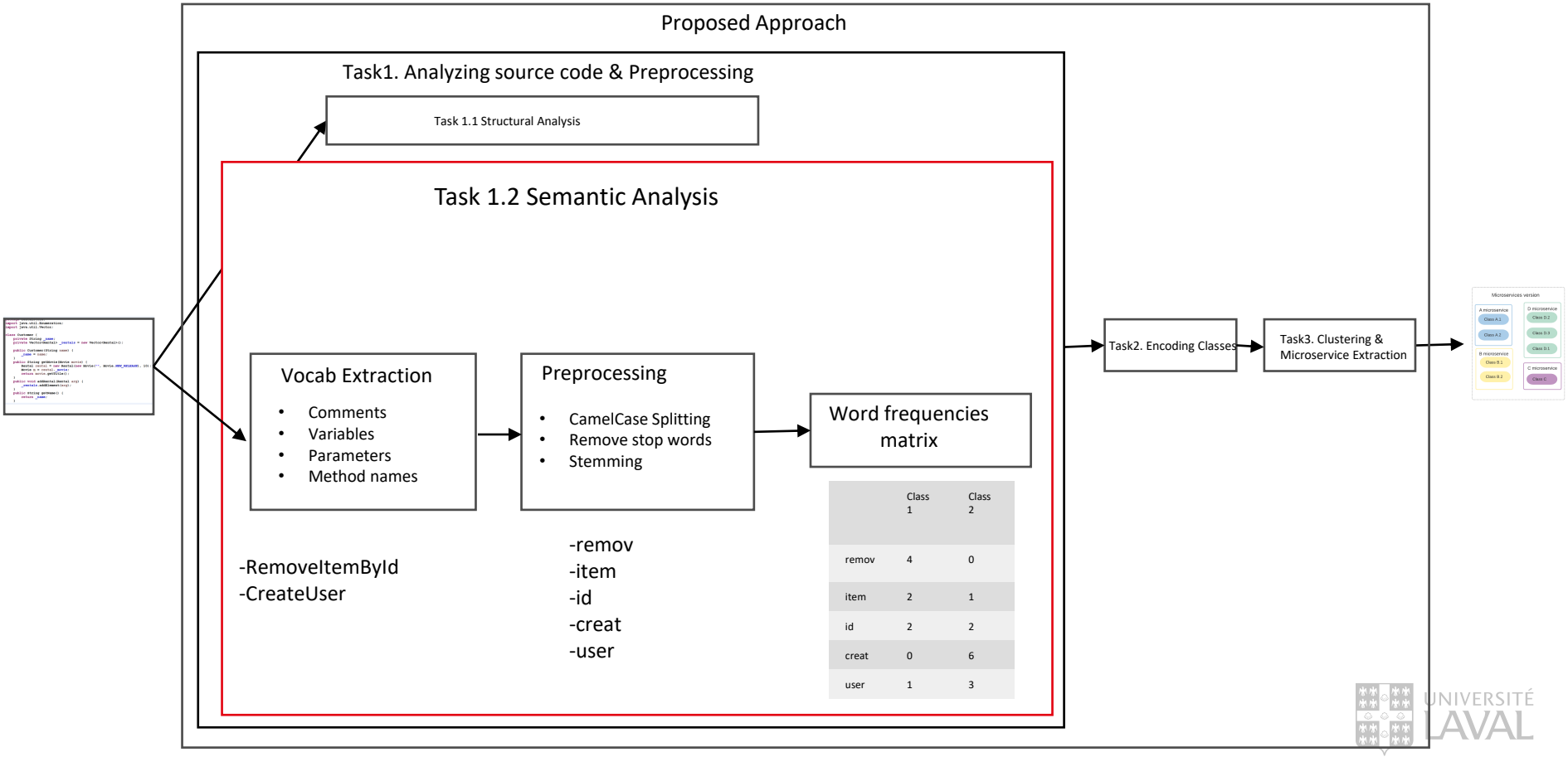
Proposed Approach 5/10



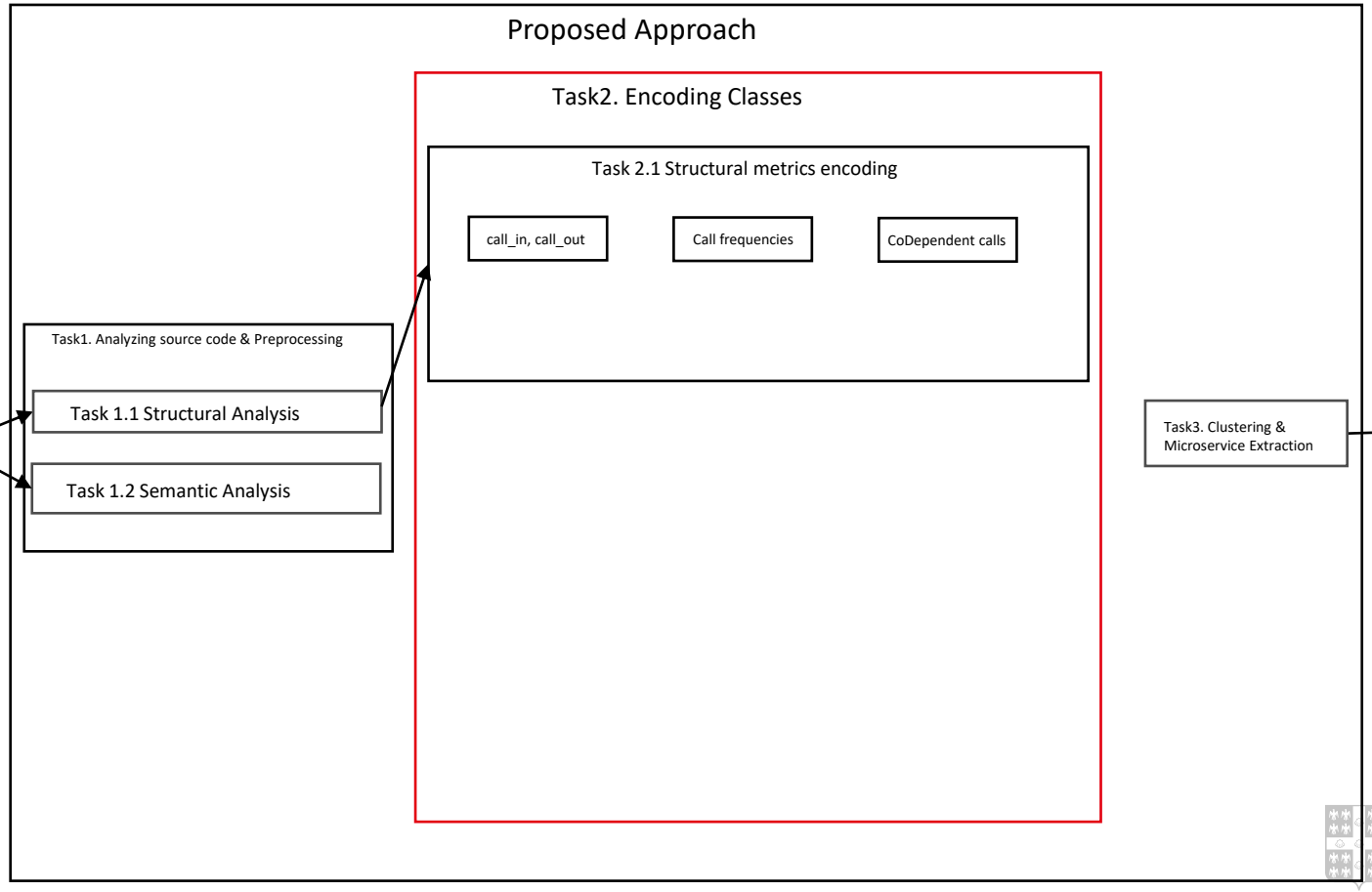
Proposed Approach 5/10



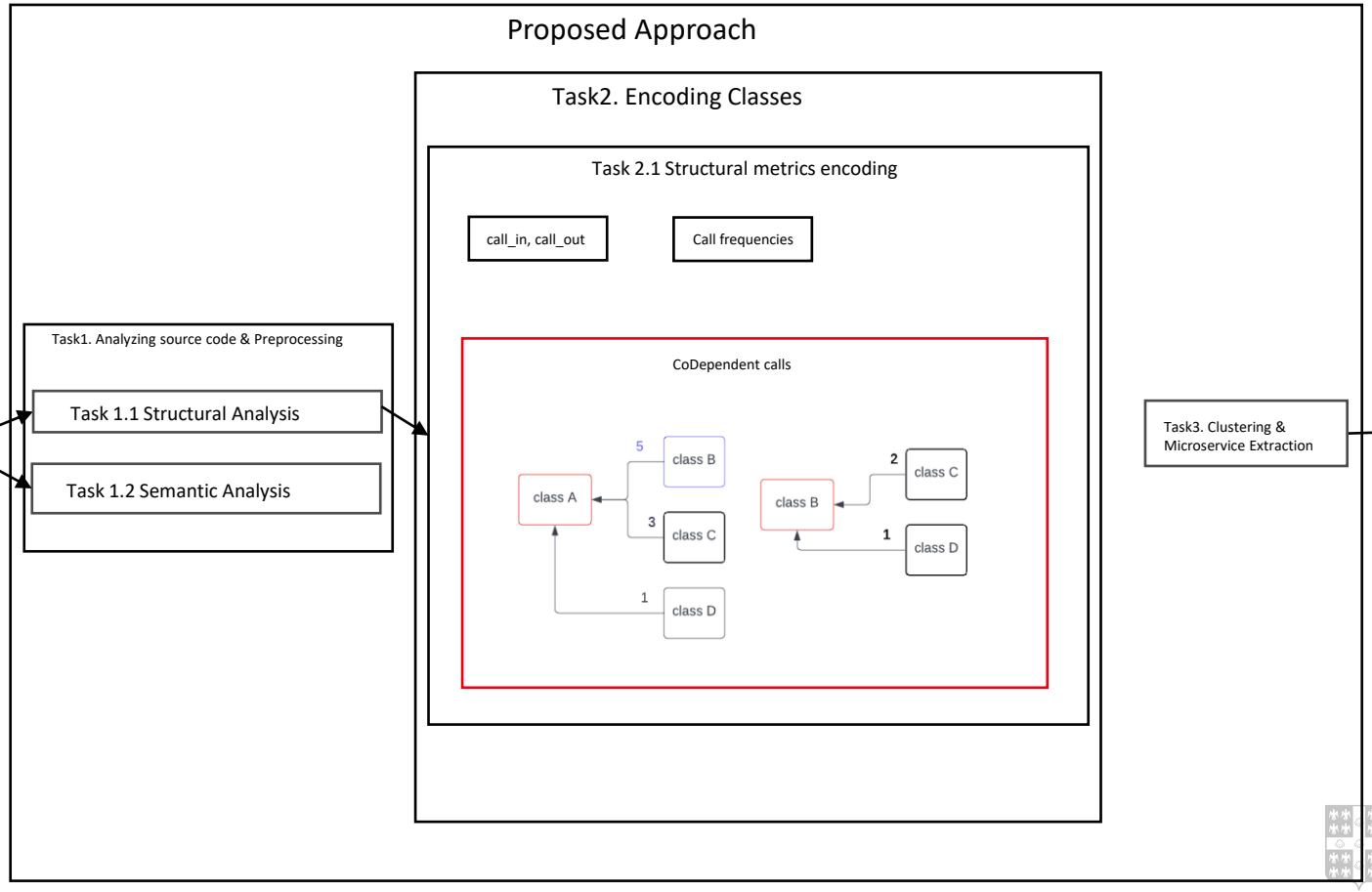
Proposed Approach 5/10



Proposed Approach 6/10

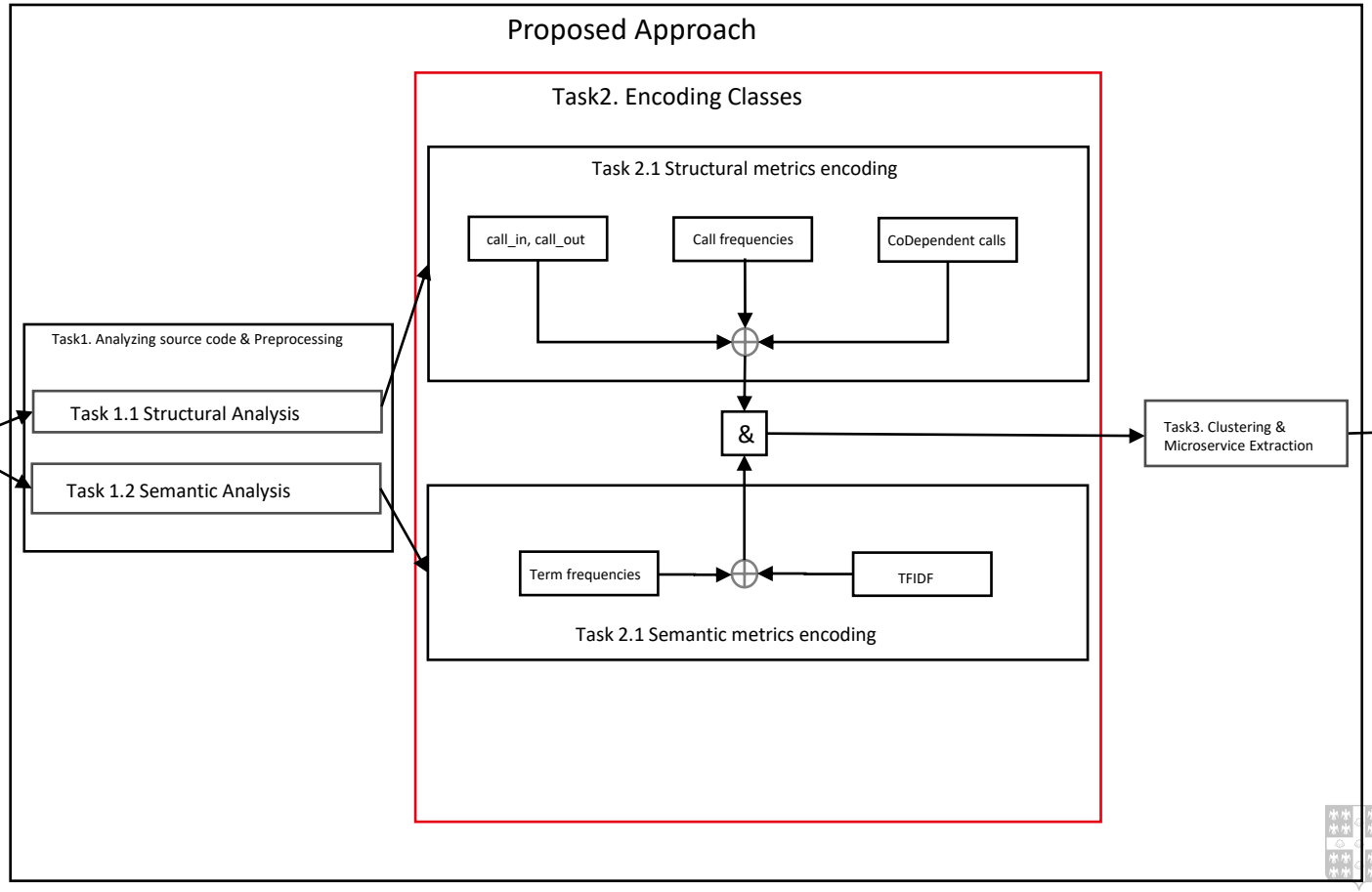


Proposed Approach 6/10

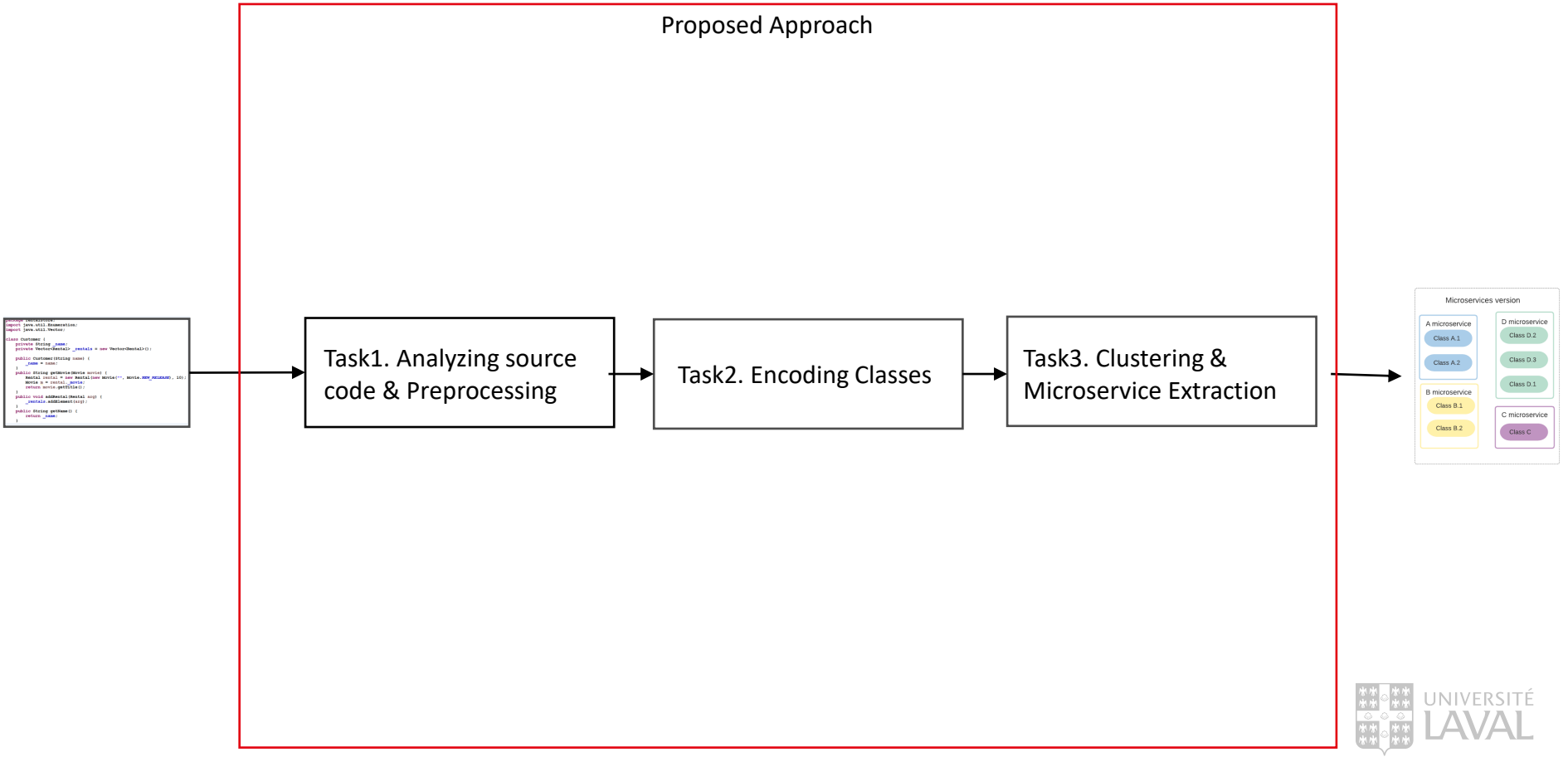


UNIVERSITÉ LAVAL

Proposed Approach 1/10

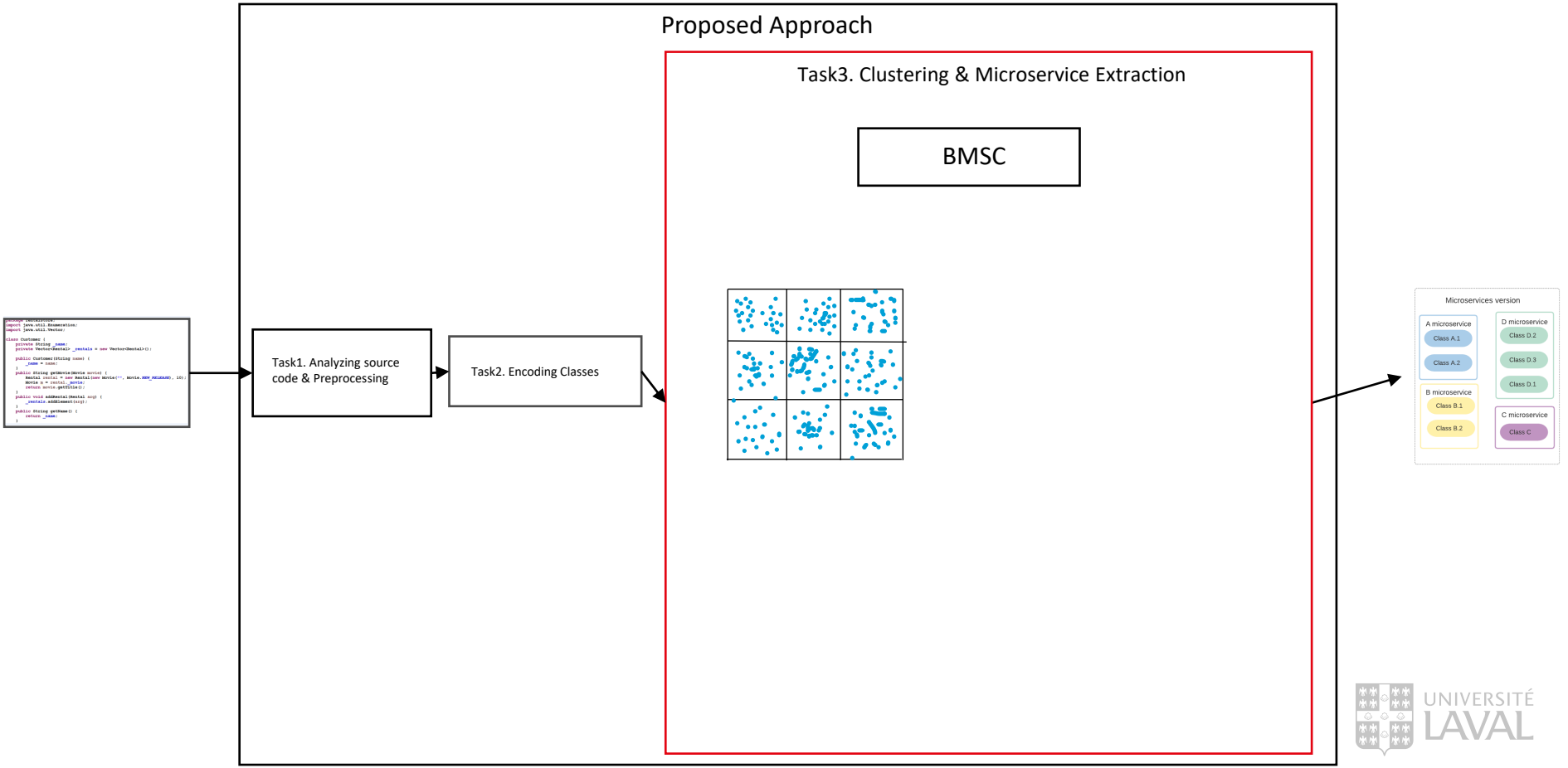


Proposed Approach 7/10



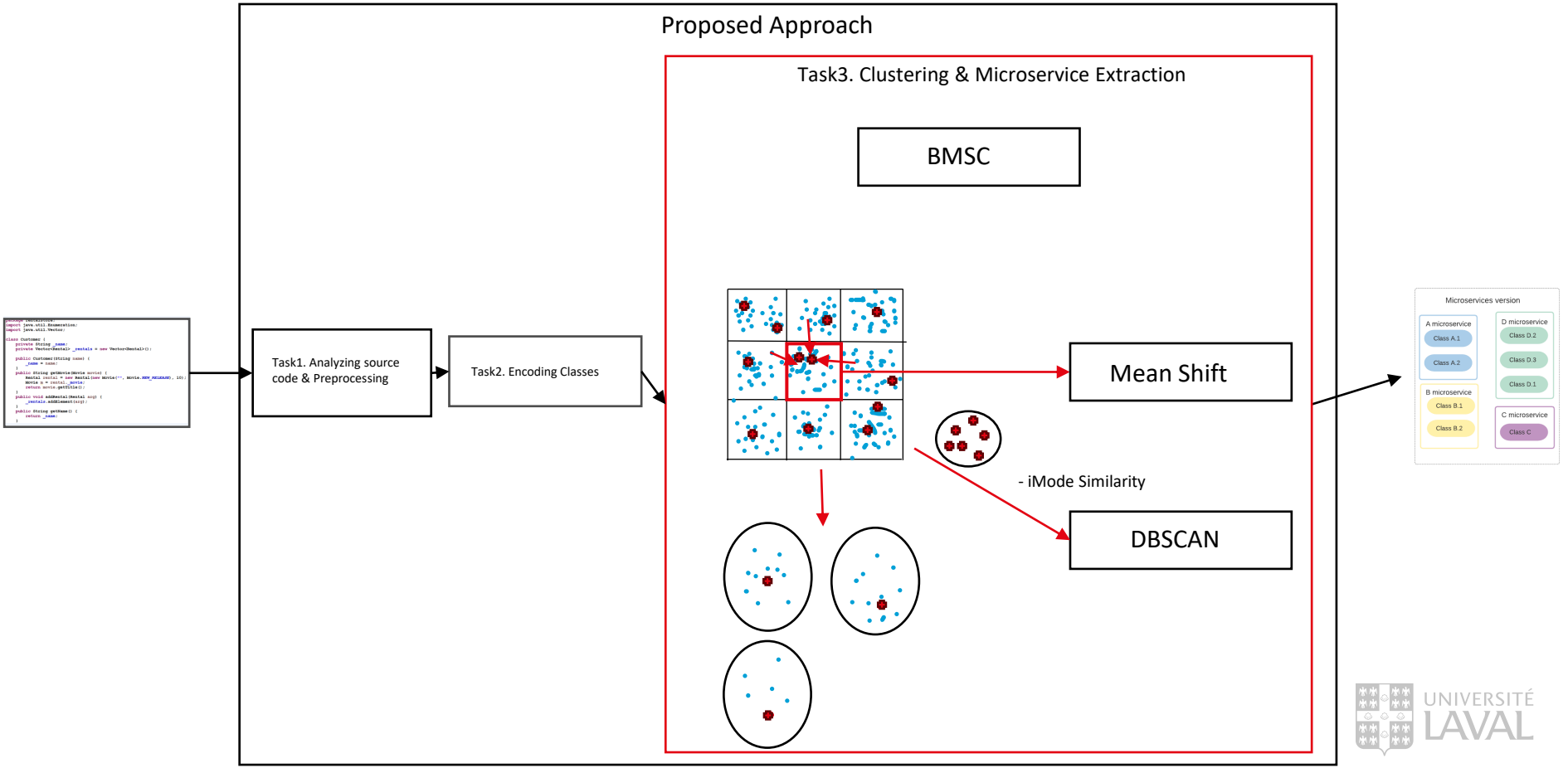
UNIVERSITÉ LAVAL

Proposed Approach 8/10



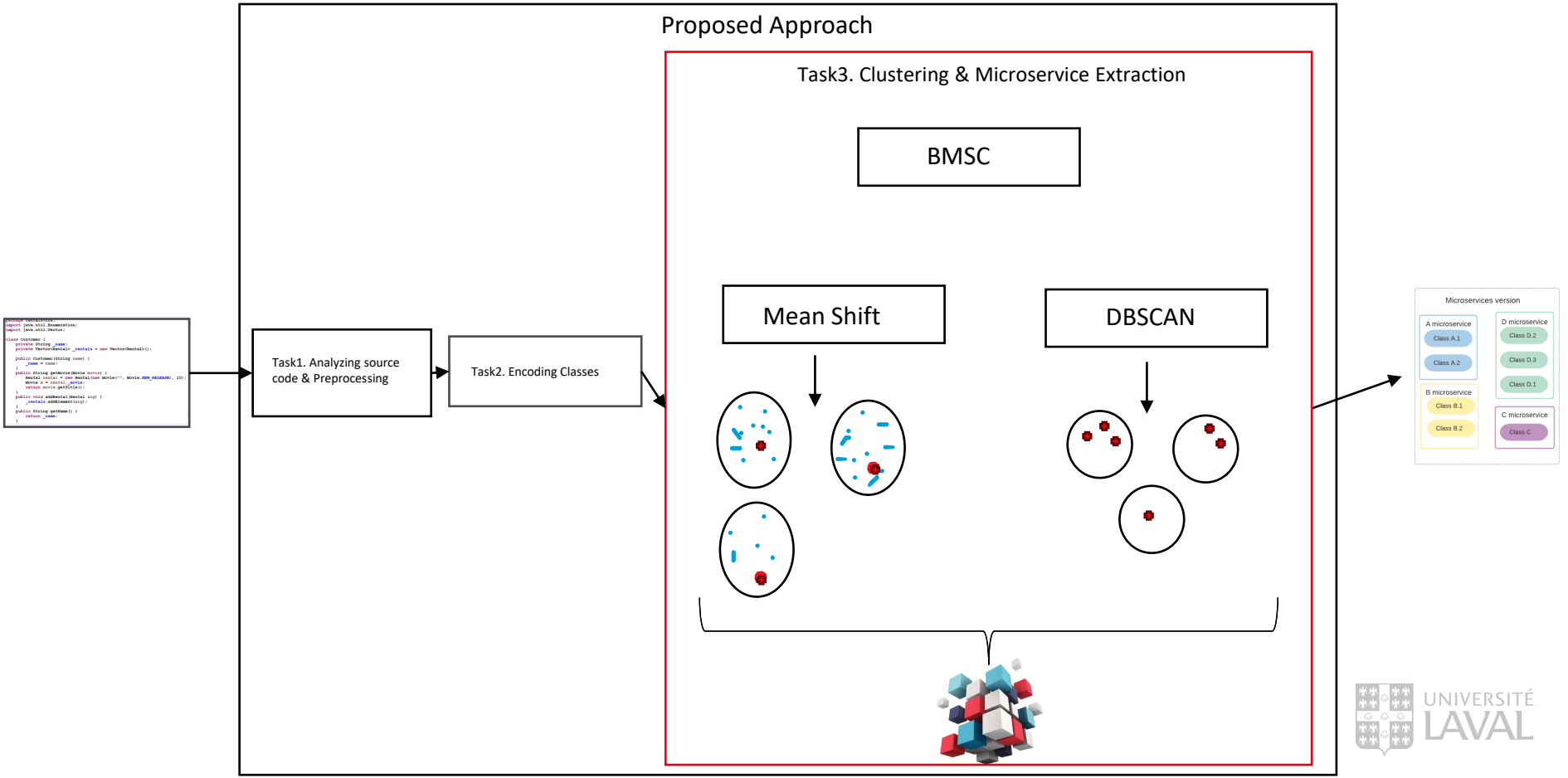
UNIVERSITÉ LAVAL

Proposed Approach 8/10



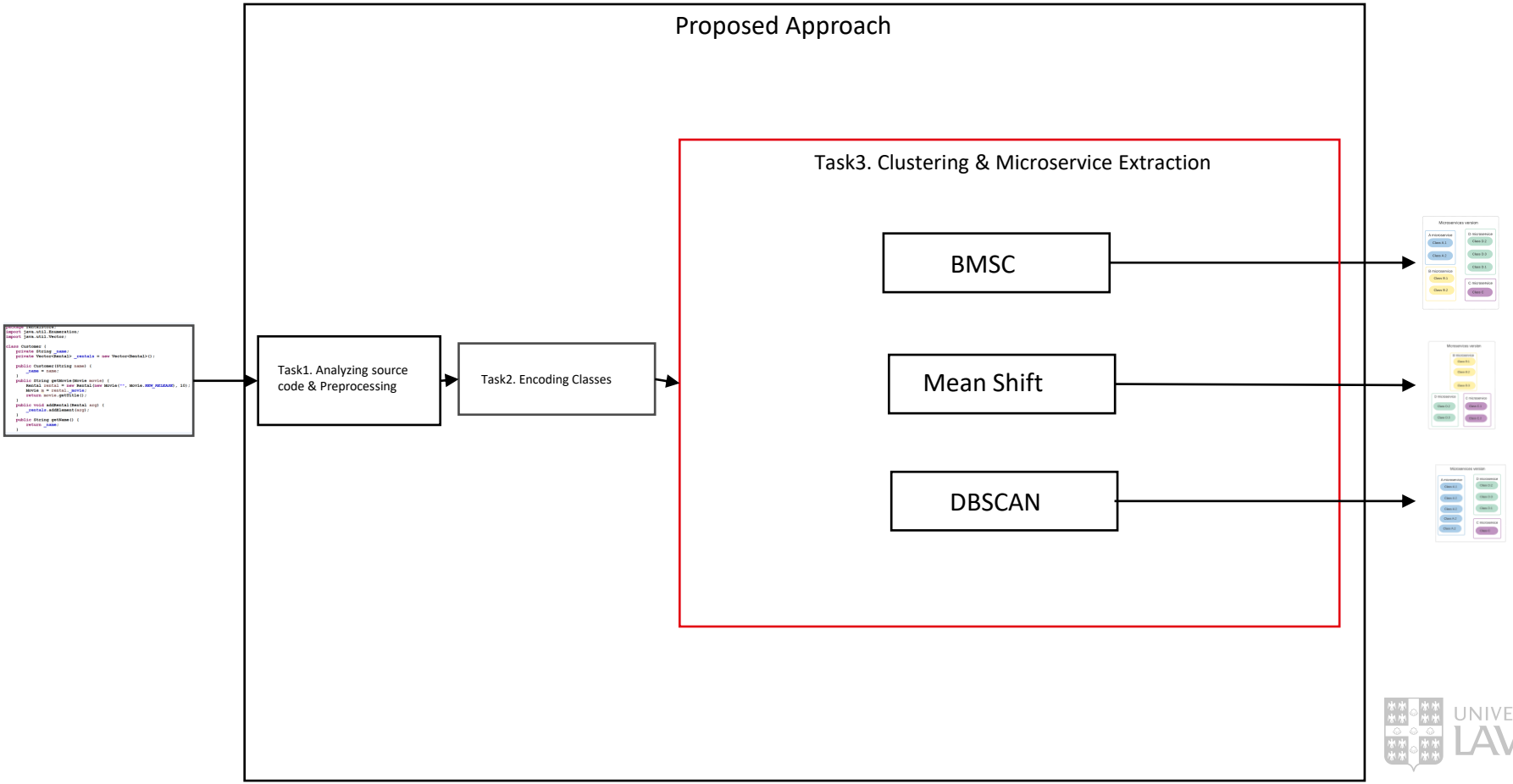
UNIVERSITÉ LAVAL

Proposed Approach 9/10



UNIVERSITÉ LAVAL

Proposed Approach 10/10



UNIVERSITÉ LAVAL

What is the most effective and promising option among the various choices in our approach?

Project:

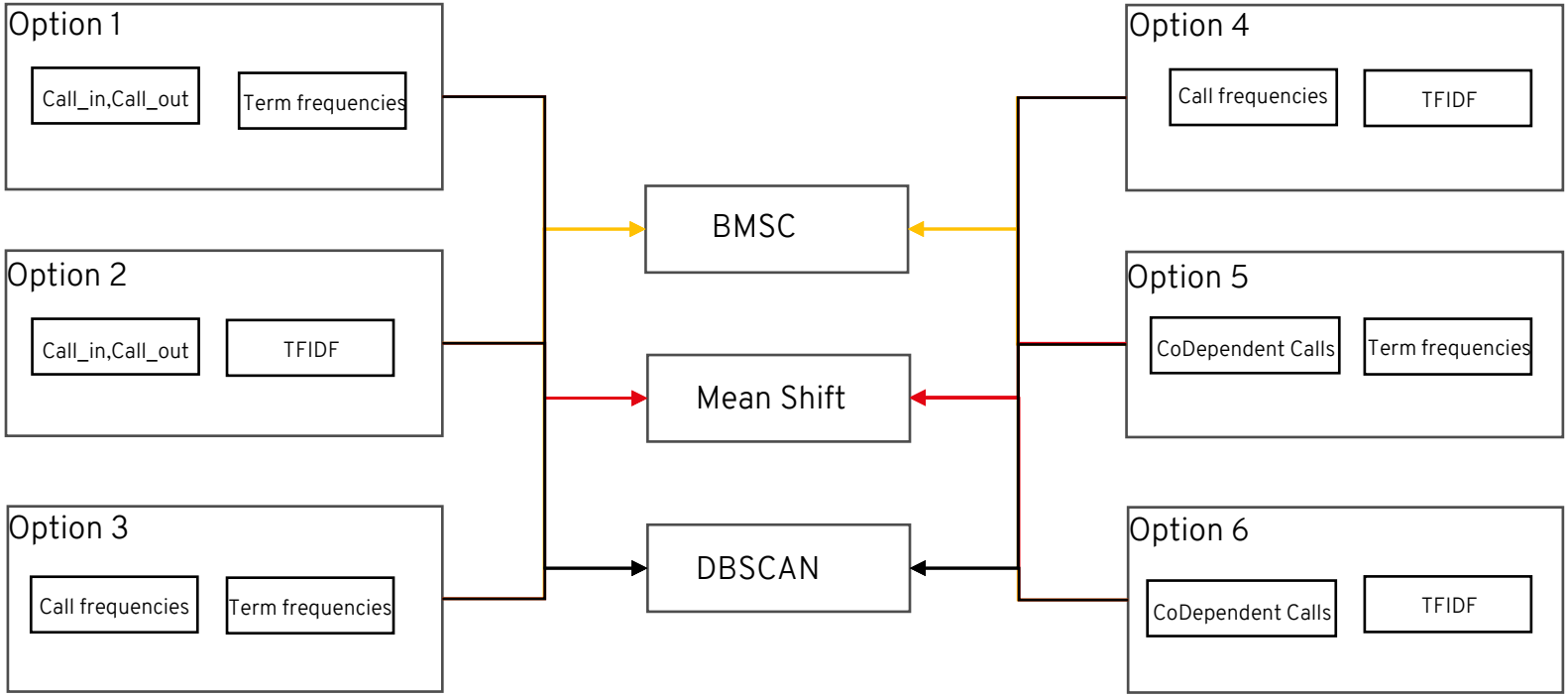
DayTrader: 118 classes

UNIVERSITÉ LAVAL

How can we evaluate the performance of the strategies?: Metrics

- Structural Modularity (SM)
- Inter Call Percentage (ICP)
- InterFace Number (IFN)
- Non-Extreme Distribution (NED)

What is the most effective and promising option among the choices of our approach?: Protocol



What is the most effective and promising option among the choices of our approach?: Results

Evaluation Results using Mean Shift Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.8526	0.7853	0.7944	0.8614	0.8575	0.8742
IFN	1.235	1.8	1.277	1.0454	1.0	1.214
ICP	1.0	0.9	1.0	1.0	1.0	1.0
NED	1.0	0.9	1.0	1.0	1.0	1.0
# microservices	17	10	18	22	21	14
size of the largest micro	98	102	97	92	97	104

Evaluation Results using DBSCAN Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
IFN	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
ICP	0.3244	0.1426	0.1591	0.2859	0.3482	0.0079
NED	0.5	0.666	1.0	0.5	0.5	0.333
# microservices	2	3	2	2	2	3
size of the largest micro	108	86	116	113	113	104

Evaluation Results using BMSC Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.3696	0.3435	0.3887	0.4697	0.40545	0.4052
IFN	1.0344	1.250	1.0370	0.9677	1.0769	1.318
ICP	0.6500	0.591	0.618	0.6432	0.6257	0.639
NED	0.7241	0.6666	0.7037	0.7419	0.6538	0.636
# microservices	29	24	27	31	26	22
size of the largest micro	13	13	13	13	16	17

What is the most effective and promising option among the choices of our approach?: Results

Evaluation Results using Mean Shift Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.8526	0.7853	0.7944	0.8614	0.8575	0.8742
IFN	1.235	1.8	1.277	1.0454	1.0	1.214
ICP	1.0	0.9	1.0	1.0	1.0	1.0
NED	1.0	0.9	1.0	1.0	1.0	1.0
# microservices	17	10	18	22	21	14
size of the largest micro	98	102	97	92	97	104

/ 118

Evaluation Results using BMSC Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.3696	0.3435	0.3887	0.4697	0.40545	0.4052
IFN	1.0344	1.250	1.0370	0.9677	1.0769	1.318
ICP	0.6500	0.591	0.618	0.6432	0.6257	0.639
NED	0.7241	0.6666	0.7037	0.7419	0.6538	0.636
# microservices	29	24	27	31	26	22
size of the largest micro	13	13	13	13	16	17

Evaluation Results using DBSCAN Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
IFN	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
ICP	0.3244	0.1426	0.1591	0.2859	0.3482	0.0079
NED	0.5	0.666	1.0	0.5	0.5	0.333
# microservices	2	3	2	2	2	3
size of the largest micro	108	86	116	113	113	104

UNIVERSITÉ LAVAL

What is the most effective and promising option among the choices of our approach?: Results

Evaluation Results using DBSCAN Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
IFN	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
ICP	0.3244	0.1426	0.1591	0.2859	0.3482	0.0079
NED	0.5	0.666	1.0	0.5	0.5	0.333
# microservices	2	3	2	2	2	3
size of the largest micro	108	86	116	113	113	104

Evaluation Results using BMSC Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.3696	0.3435	0.3887	0.4697	0.40545	0.4052
IFN	1.0344	1.250	1.0370	0.9677	1.0769	1.318
ICP	0.6500	0.591	0.618	0.6432	0.6257	0.639
NED	0.7241	0.6666	0.7037	0.7419	0.6538	0.636
# microservices	29	24	27	31	26	22
size of the largest micro	13	13	13	13	16	17

Evaluation Results using Mean Shift Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.8526	0.7853	0.7944	0.8614	0.8575	0.8742
IFN	1.235	1.8	1.277	1.0454	1.0	1.214
ICP	1.0	0.9	1.0	1.0	1.0	1.0
NED	1.0	0.9	1.0	1.0	1.0	1.0
# microservices	17	10	18	22	21	14
size of the largest micro	98	102	97	92	97	104

What is the most effective and promising option among the choices of our approach?: Results

Evaluation Results using BMSC Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.3696	0.3435	0.3887	0.4697	0.40545	0.4052
IFN	1.0344	1.250	1.0370	0.9677	1.0769	1.318
ICP	0.6500	0.591	0.618	0.6432	0.6257	0.639
NED	0.7241	0.6666	0.7037	0.7419	0.6538	0.636
# microservices	29	24	27	31	26	22
size of the largest micro	13	13	13	13	16	17

Evaluation Results using DBSCAN Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
IFN	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
ICP	0.3244	0.1426	0.1591	0.2859	0.3482	0.0079
NED	0.5	0.666	1.0	0.5	0.5	0.333
# microservices	2	3	2	2	2	3
size of the largest micro	108	86	116	113	113	104

Evaluation Results using Mean Shift Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.8526	0.7853	0.7944	0.8614	0.8575	0.8742
IFN	1.235	1.8	1.277	1.0454	1.0	1.214
ICP	1.0	0.9	1.0	1.0	1.0	1.0
NED	1.0	0.9	1.0	1.0	1.0	1.0
# microservices	17	10	18	22	21	14
size of the largest micro	98	102	97	92	97	104

What is the most effective and promising option among the choices of our approach?: Results

Evaluation Results using BMSC Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.3696	0.3435	0.3887	0.4697	0.40545	0.4052
IFN	1.0344	1.250	1.0370	0.9677	1.0769	1.318
ICP	0.6500	0.591	0.618	0.6432	0.6257	0.639
NED	0.7241	0.6666	0.7037	0.7419	0.6538	0.636
# microservices	29	24	27	31	26	22
size of the largest micro	13	13	13	13	16	17

Option 6 is the optimal approach for all three clustering algorithms. It uses co-dependent calls metric as structural information and TF-IDF vector as semantic information. Consequently, our work will continue to focus on this strategy.

Evaluation Results using Mean Shift Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.8526	0.7853	0.7944	0.8614	0.8575	0.8742
IFN	1.235	1.8	1.277	1.0454	1.0	1.214
ICP	1.0	0.9	1.0	1.0	1.0	1.0
NED	1.0	0.9	1.0	1.0	1.0	1.0
# microservices	17	10	18	22	21	14
size of the largest micro	98	102	97	92	97	104

Evaluation Results using DBSCAN Algorithm

Metrics	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6
SM	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
IFN	0.120	0.1085	0.2702	0.2718	0.2487	0.1116
ICP	0.3244	0.1426	0.1591	0.2859	0.3482	0.0079
NED	0.5	0.666	1.0	0.5	0.5	0.333
# microservices	2	3	2	2	2	3
size of the largest micro	108	86	116	113	113	104

How does the stability and robustness of BMSC algorithms compare to that of Mean Shift and DBSCAN?

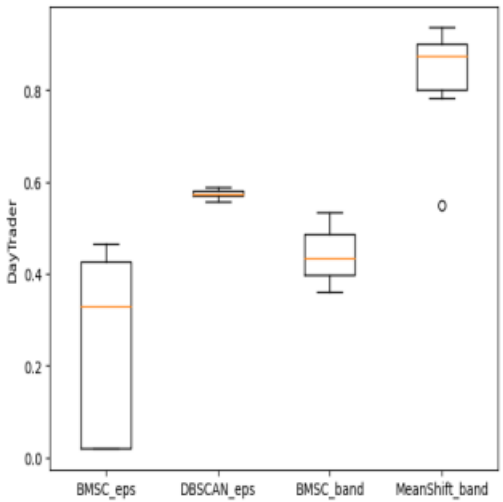
Hyperparams:

- Bandwidth
- Epsilon
- MinPts =5

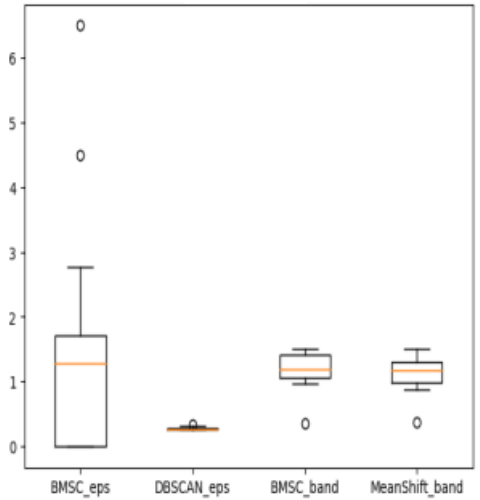
UNIVERSITÉ LAVAL

How does the stability and robustness of BMSC algorithms compare to that of Mean Shift and DBSCAN?: Results

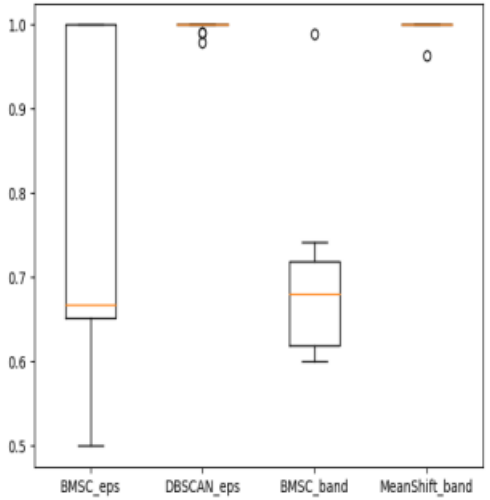
Structural Modularity (SM)



InterFace Number (IFN)



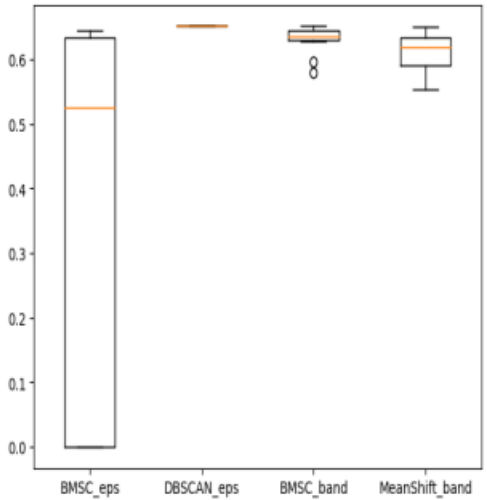
Inter Call Percentage (ICP)



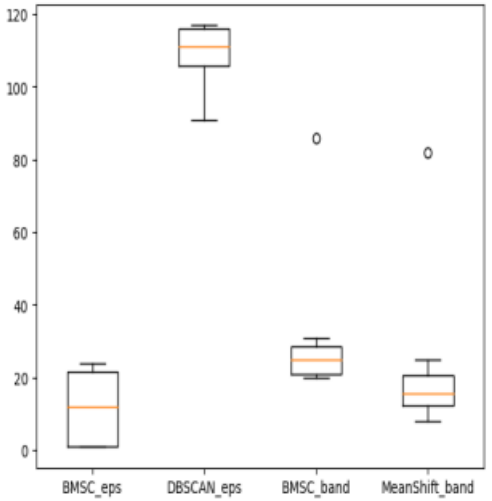
UNIVERSITÉ LAVAL

How does the stability and robustness of BMSC algorithms compare to that of Mean Shift and DBSCAN?: Results

Non Extreme Distribution(NED)



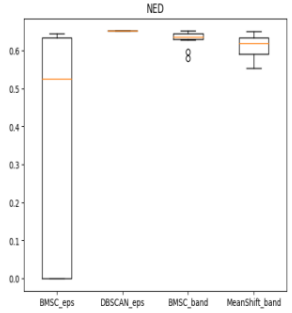
microservices

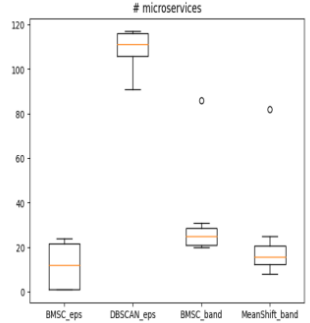


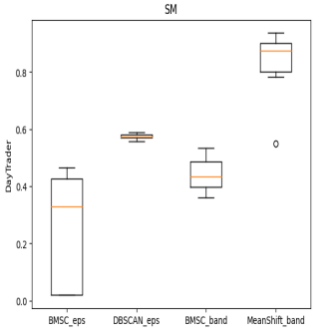
UNIVERSITÉ LAVAL

How does the stability and robustness of BMSC algorithms compare to that of Mean Shift and DBSCAN?: Results

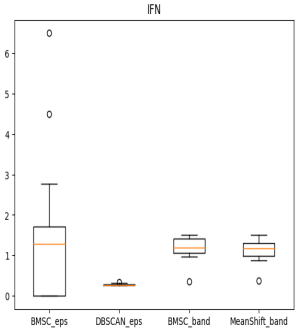
Non Extreme Distribution(NED)



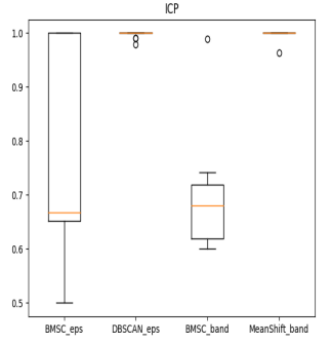




Structural Modularity (SM)



InterFace Number (IFN)



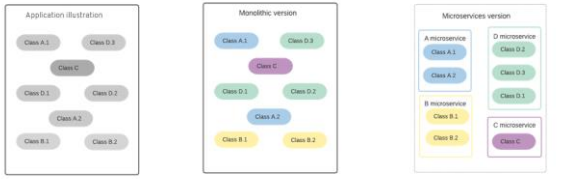
Inter Call Percentage (ICP)

In contrast to the literature, our analysis suggests that for our case, BMSC is more susceptible to the selection of its hyperparameters, specifically the epsilon parameter, compared to DBSCAN and Mean Shift when used independently. However, BMSC demonstrates greater consistency in the resulting decompositions across hyperparameter variations.

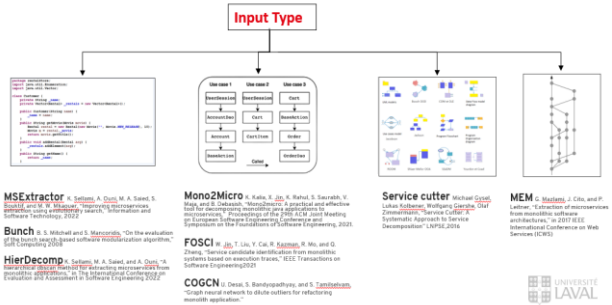
UNIVERSITÉ LAVAL

Conclusion

Context 2/2

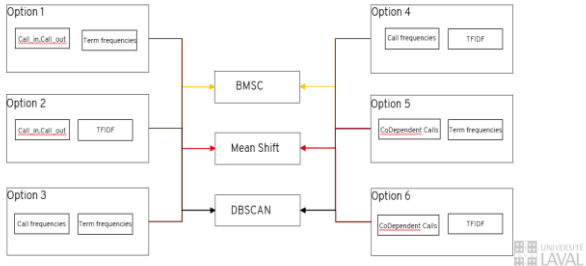


Related Work 1/2

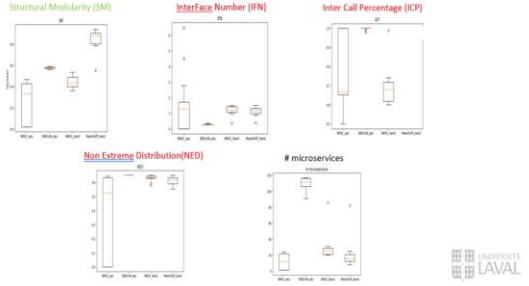


Proposed Approach 1/2

What is the most effective and promising option among the choices of our approach?: Protocol



What is the most effective and promising option among the choices of our approach?: Results



Questions ?