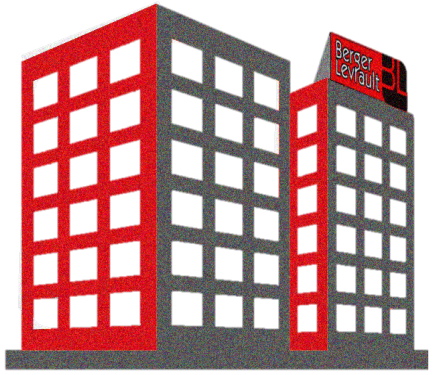
L’exploitation de traces logicielles à Berger-Levrault.

Travail réalisé par:
BOUKHAROUBA Ikram
Ingénieure de recherche et développement

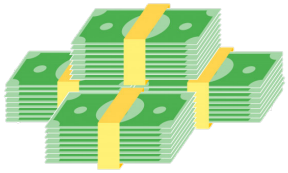
Faisons connaissance:

- Berger-Levrault est un éditeur de logiciels travaillant principalement pour le secteur public

2000 employés dans 5 continents



Berger-Levrault

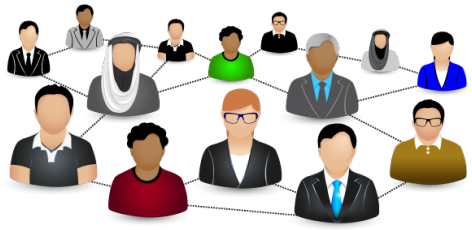


186 M$ de revenue

25% de revenue

1463
Maison d'édition

1981
Vers le digital



51.000
clients avec des millions d'utilisateurs

Faisons connaissance:

2014- 2017	Licence en Informatique de l'université de Guelma/Algérie
2017- 2019	Master en reseaux et cybersécurité de l'université d'Annaba/Algérie
2019- 2020	MSc Computer Science for Communication Networks (Télécom SudParis)
Mars 2020	Stage de fin d'étude à BL
2020- 2021	Ingénieure R&D à BL
Depuis septembre 2021	Ingénieure R&D et Doctorante en contrat CIFRE entre l'IRIT et BL

C’est quoi une trace logicielle ?

Notion d'historique (garder une trace)

Première référence aux traces :

Influence d'un événement sur son environnement

Notion d'évènement (empreinte)

De manière générale…

Séquence chronologique d'évènements

Eléments ordonnés les uns par rapport aux autres

Décrit quelque-chose de réalisé = un fait

Capture d'un élément du monde à un instant donné, et qui compose un historique

Et pour un logiciel ?

C’est quoi une trace d’activité utilisateur

- Ensemble de données structurées et horodatées, générées directement par un logiciel
- Captures les interactions d’un utilisateur avec l’interface d’une app/logiciel
  - *Coté frontend (GUI): capture des composants graphiques impliqués dans l'interaction de l'utilisateur (clic, survol, ...), capture de la navigation de l'utilisateur (chemin) dans le produit*
- Captures le comportement de l’app/logiciel suite à une activité IHM d’utilisateur
  - *Coté backend(Server): capture des services déclenchés, processus métier, accès aux données, erreurs métiers et/ou techniques*

Exemple d'une trace Sedit chez BL (frontend)

Sedit


```
{
   "agentName":"TraceAgent",
   "softwareName":"201028.1230",
   "softwareRelease":"N/A",
   "softwareVersion":"2020.6-SNAPSHOT",
   "userName":"ADMIN",
   "sessionId":"CC2C696F860592967E761F33998D0555",
   "remoteAdress":"10.32.100.86",
   "data":{
      "title":"Fermer",
      "isEnable":"true",
      "counter":"1135"
   },
   "traceType":"BUSINESS",
   "timeStamp":"2020-10-28 02:07:32.505 PM",
   "browserTabID":"9778fdc3-eb93-498f-94ad-d9e4ea33d34f",
   "event":"ON_CLICK",
   "action":"SELECT",
   "actionTarget":"BUTTON",
   "actionTargetClass":"fr.bl.client.core.refui.base.components.BLImageButton",
   "actionDetail":"Clic sur un bouton (image)"
}
```


Quand (horodatage)

Qui (utilisateur)

Quoi (granularité haute)

Pourquoi tracer l'activité des utilisateurs sur un logiciel ?

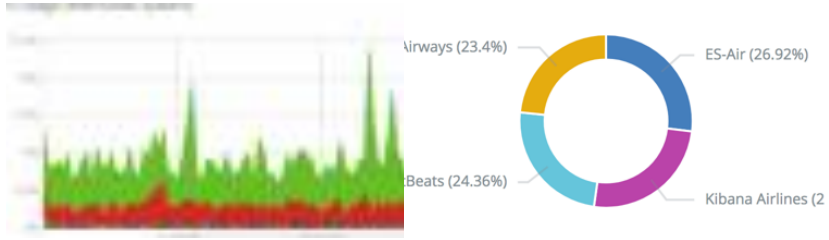
Les journaux sont des mines d'or !

Informations sur **l'usage** du logiciel

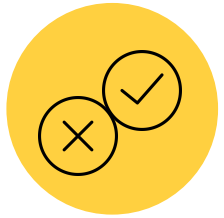
Découvertes d'usages détournés :

- *Positifs*
- *Négatifs*

Informations sur l'état/**fonctionnement** du logiciel

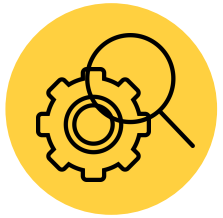


Analyse d'anomalies / Détection incidents



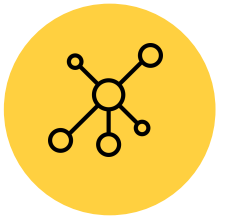
Amélioration des tests

Profiling / recommandations / prédictions



Système auto-observé

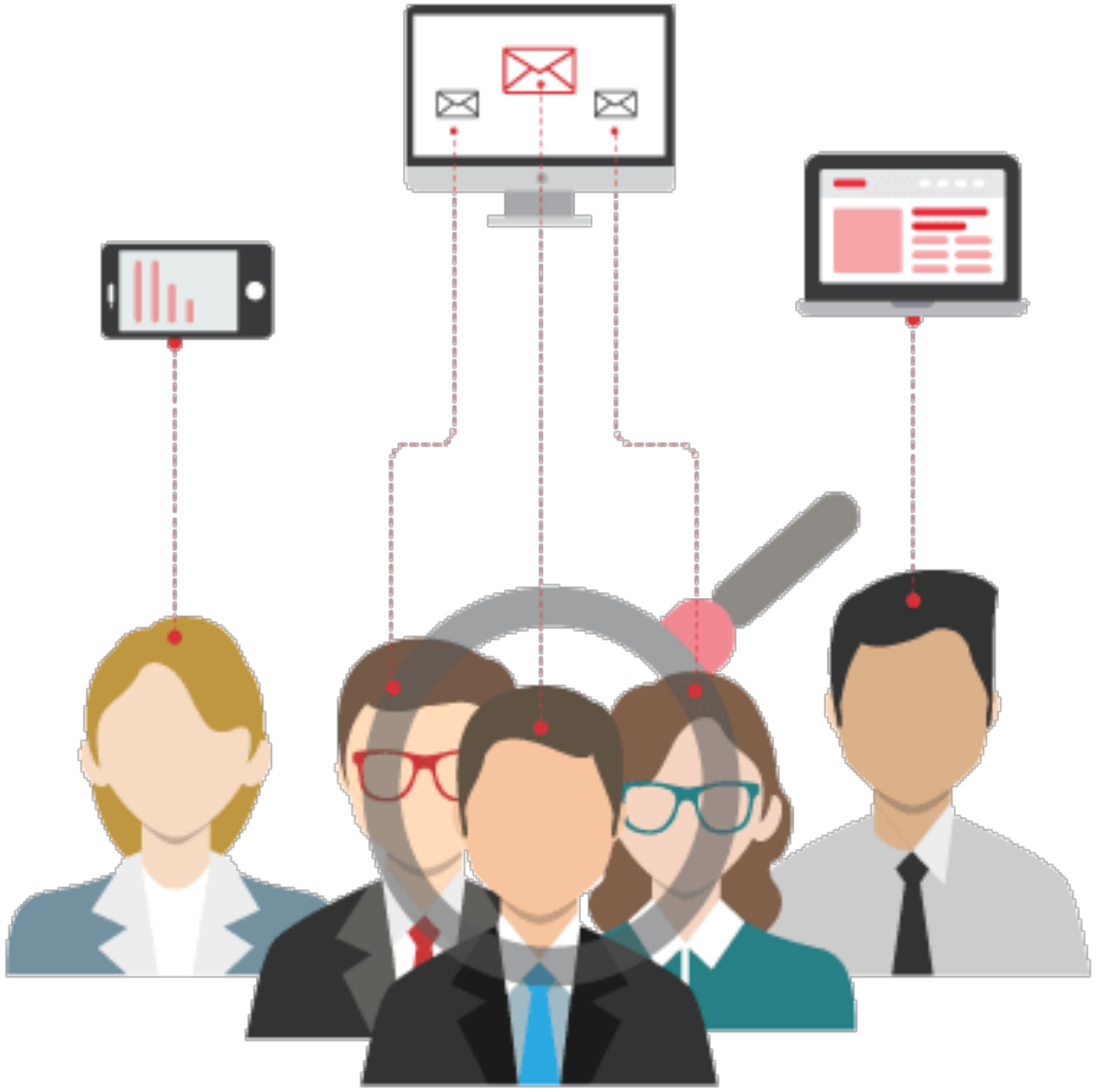
Système auto-testé



Système auto-réparable / auto-adaptable

Exemple d’exploitation de traces logicielles à BL!

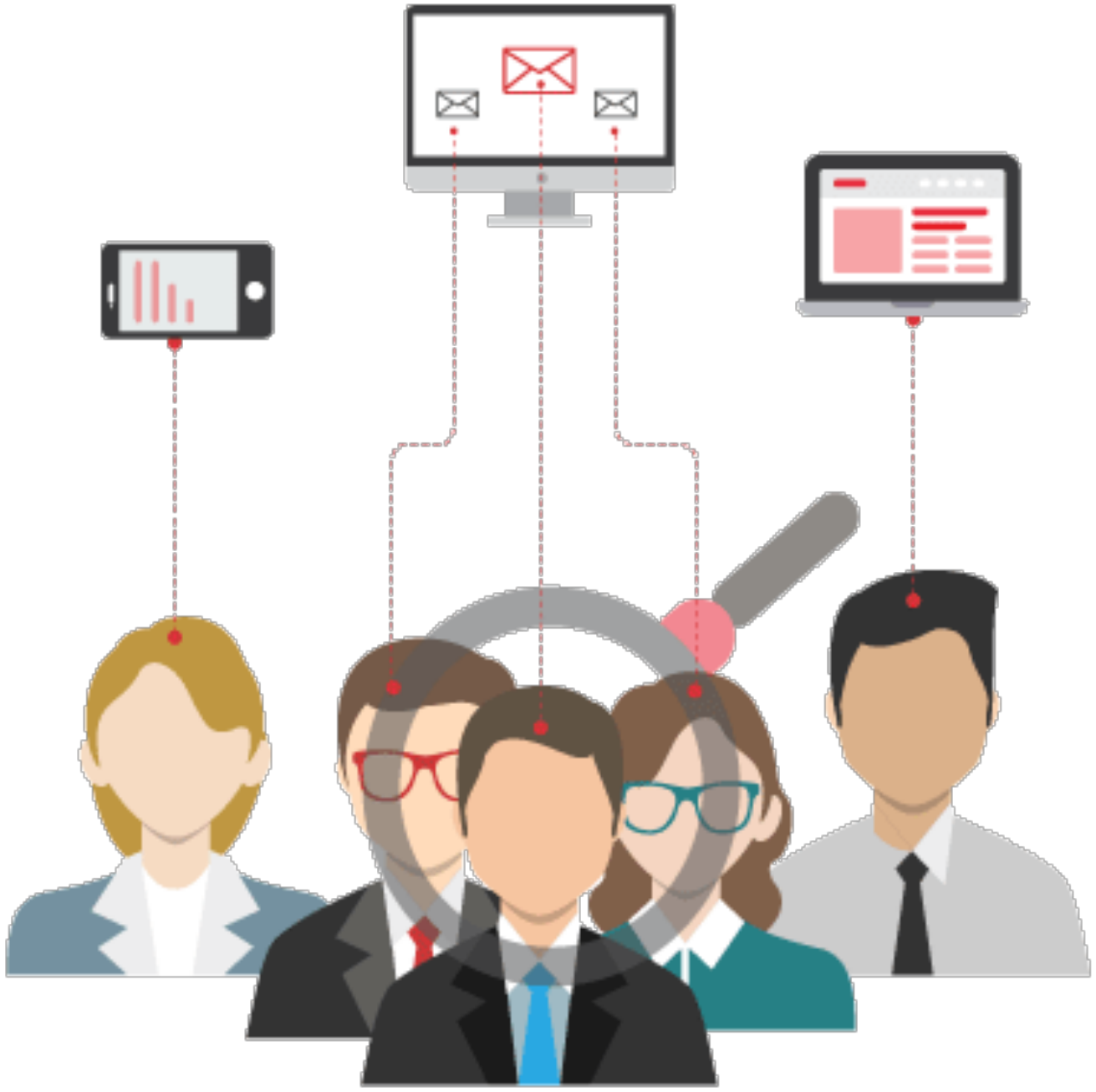
1- Tracer les accès et gérer les incidents



La traçabilité pour des raisons de sécurité:

1. Tracer les accès et prévoir des procédures pour gérer les incidents afin de pouvoir réagir en cas de violation de données
2. Identifier un accès frauduleux
3. Détecter une utilisation abusive de données personnelles
4. Déterminer l'origine d'un incident

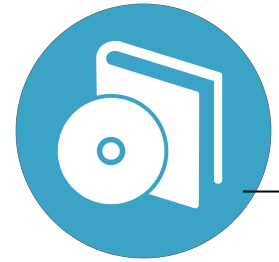
2- Analyse des traces logicielles pour la détection de la fraude



Présentation du scénario de fraude sur BLEnfance(1)

BLEnfance est un logiciel qui accompagne les mairies et associations dans la prise de décision et l'organisation d'activités liées aux enfants dans l'espace scolaire (cantine) ou périscolaire (crèche, centre de loisirs). Il fait la gestion des factures selon différents critères comme le quotient familial (Q-CAF), la période de facturation, etc.

Présentation du scénario de fraude sur BLEnfance(2)

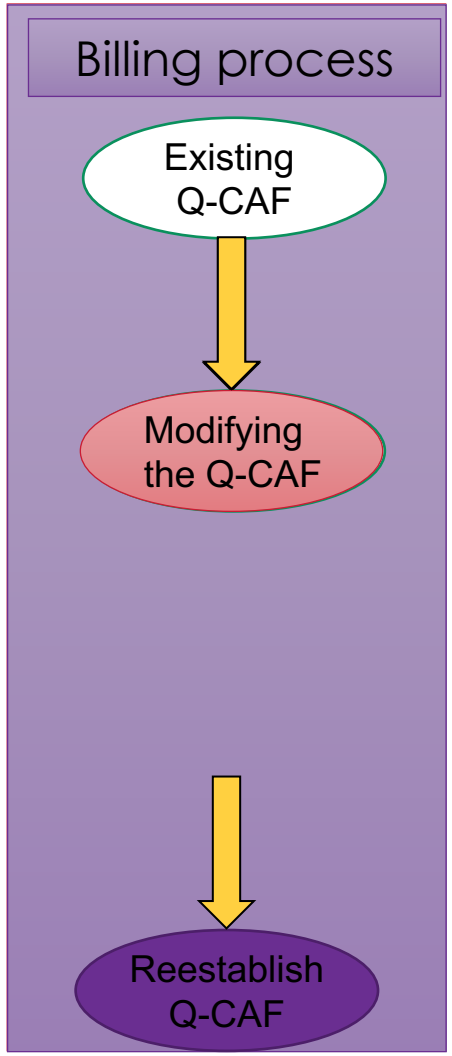


BLEnfance

Q-CAF
(family quotient)
Calculated based on the revenues of the family

Bills management

Agent



The fraud scenario is represented by the succession of this activities by the same or different agents for the same family on the same period.

High value of the **Q-CAF**
=
Low value of the **bill**

After identifying the possible fraud case over the BLEnfance software we will put in place the prototype that detect this kind of manipulations by exploiting BLEnfance's execution logs

Exemple de trace logicielle sur BLEnfance

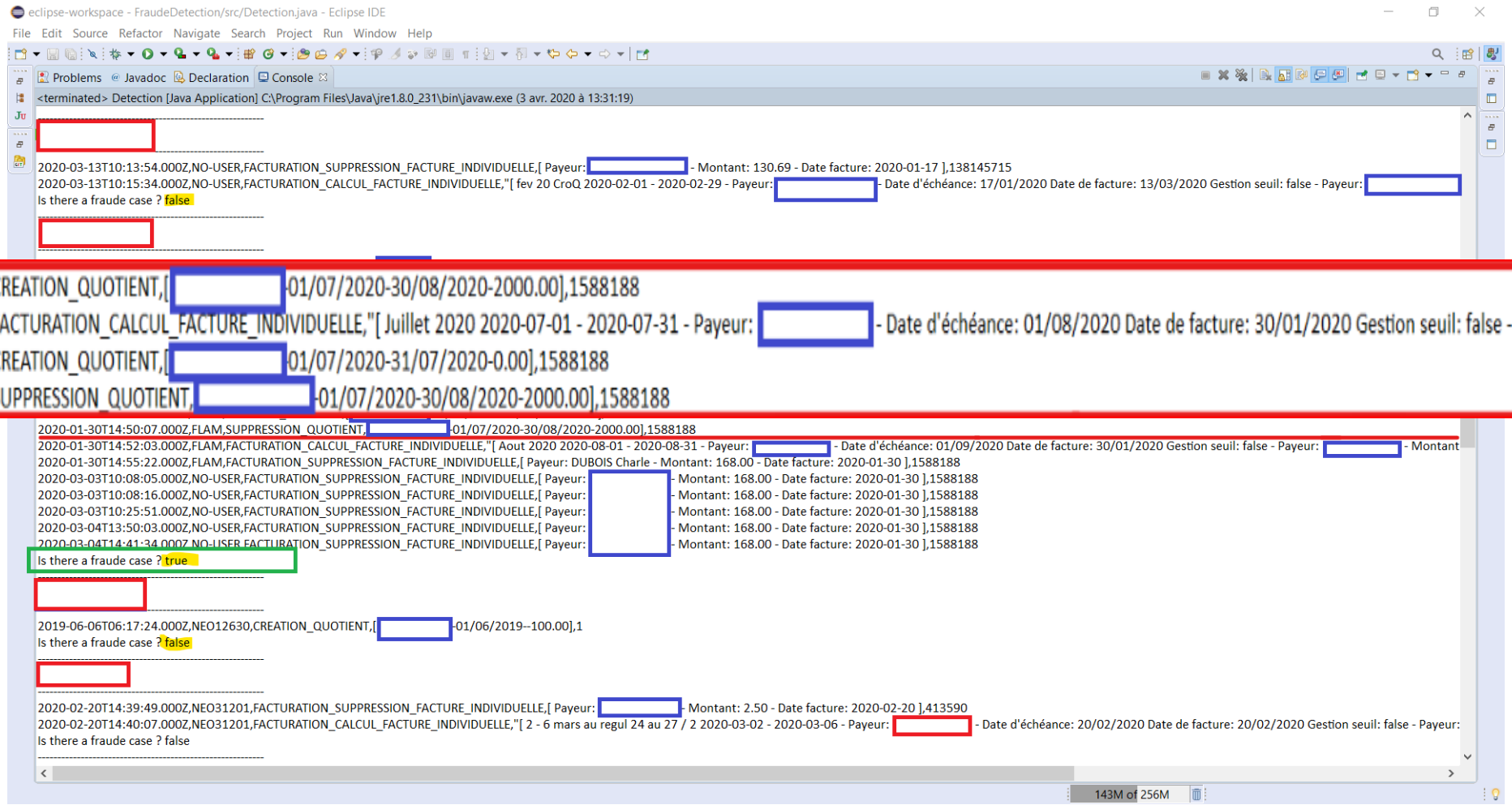
```
_id: ObjectId("593960d8c2dcec0f345eeb2b")
_class: "fr.bl.logmanager.logger.entity.LogDB"
dateHeure: 2017-06-08T14:36:08.000+00:00
login: "NEO31330"
ip: "10.31.101.95"
action: "MODIFICATION_ACTIVITE"
valeurPrecedente: "[ 30 Repas Enfants]"
valeurActuelle: "[30 Repas Enfants]"
idClient: 3633919
```

Présentation du scénario de fraude sur BLEnfance(3)

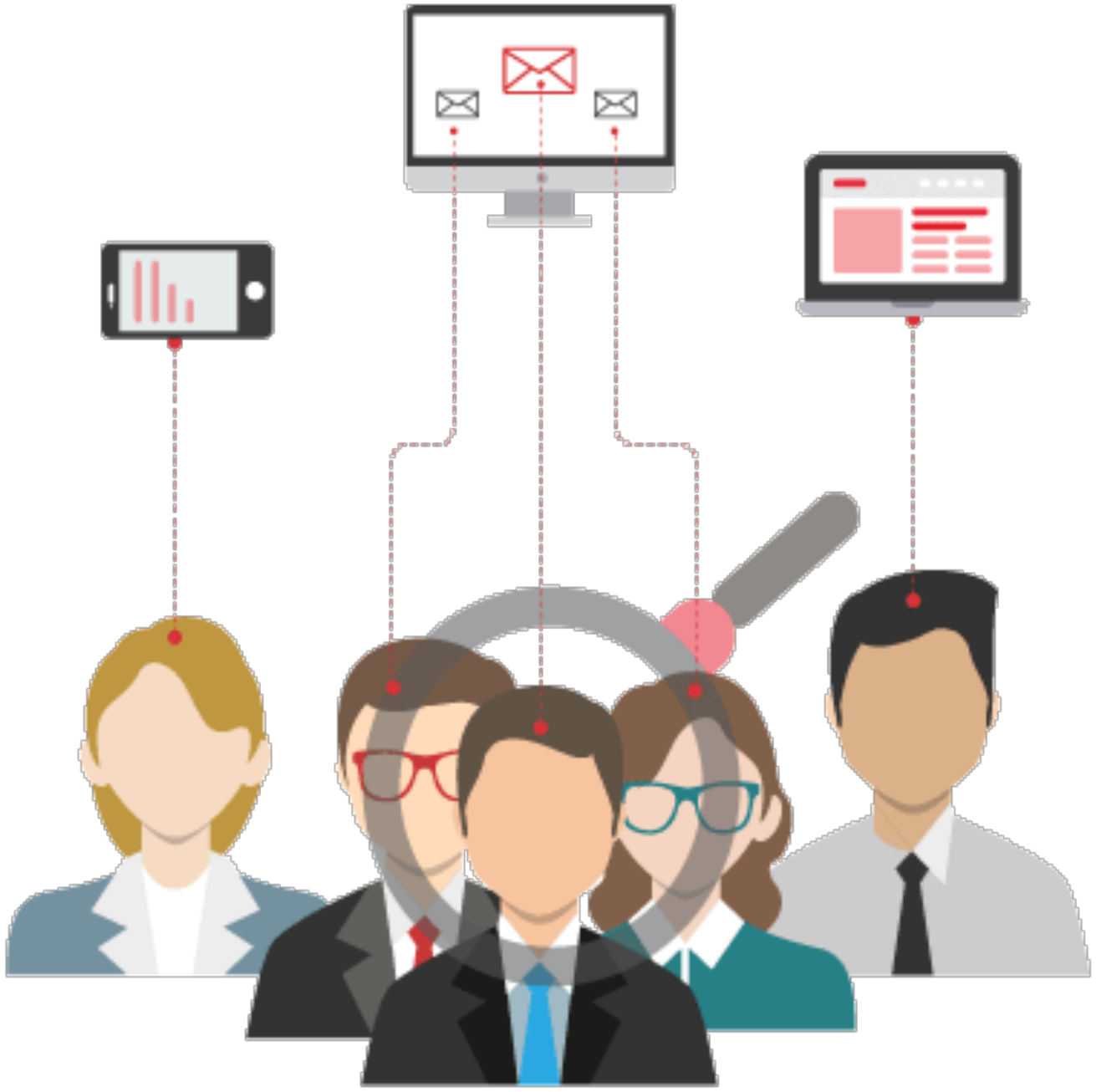
Ce cas de fraude concerne 3 cas d'utilisation (CU), qui sont indépendamment légitimes. **La succession de ces 3 CU** pour une même famille sur les mêmes périodes constitue certainement une fraude, sachant que **plus le Q-CAF est élevé, plus la facture est basse.**

- CU1 : "Modification de Q-CAF pour frauder la facturation"
- CU2 : "Facturation pour la période frauduleuse"
- CU3 : "Modification de Q-CAF pour masquer la fraude"

Résultats de détection du CU de fraude



3- Analyse des traces logicielles (Front) des utilisateurs pour l'analyse comportementale



Pourquoi faire?

Évolution de logicielle ➔ + complexité

+ temps d'adaptation

Maintenir un niveau de satisfaction utilisateur

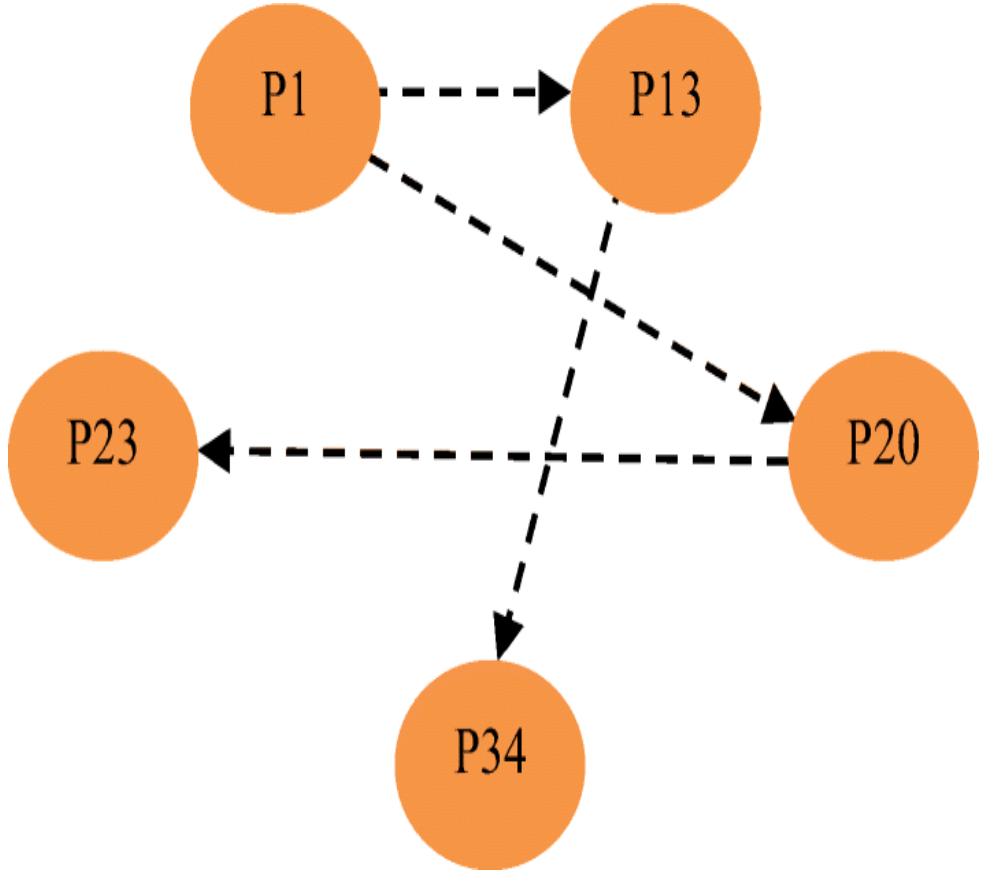
Savoir comment les utilisateurs interagissent avec le logiciel est essentiel pour comprendre l'expérience que vous leur offrez.

Nos objectifs à BL!

1. Informer les campagnes de test avec des scénarios réalistes
2. Minimiser les temps de prise en main des applications par les utilisateurs finaux
3. Optimiser l'assistance aux utilisateurs par les services support, détecter des variations d'usages
4. Prédire les comportements et tendre vers une assistance prédictive à l'usage des interfaces utilisateurs

Le graphe de navigation utilisateurs

- Un parcours utilisateur est une représentation visuelle de ce qui se passe lorsque les gens utilisent un site Web ou une application.
- C'est l'itinéraire qu'un utilisateur emprunte pour trouver les informations qu'il recherche ou réaliser une tache sur un site Web ou une application.
- Un nœud = la page visitée par l'utilisateur
- Un lien = le passage d'une page X à une Page Y.



Le graphe de navigation: Ça sert à quoi ?

1. Comment les utilisateurs naviguent sur notre site/logiciel?
2. Quels raccourcis prennent-ils ?
3. Où tombent-ils ?
4. Y a-t-il des pages non accessibles?

Étapes de construction de graphe de Navigation sur Sedit

1. Traitement de traces front, on ne garde que les traces d'action d'ouverture/fermeture des pages (autres: clic bouton, popups etc)
2. Construction de session utilisateurs
3. Agrégation des sessions de même utilisateur
4. Construction de graph de navigation utilisateur

Trace Visualization

localhost:3030

Berger Levrault

SEDIT events vizualisation :

Manipulate unique user interactions:

User: 129 ,actions = 51

Nodes style

Node size by time spent on

Edges style

Show concatenaed edges wi

Show edges in the order of navigation

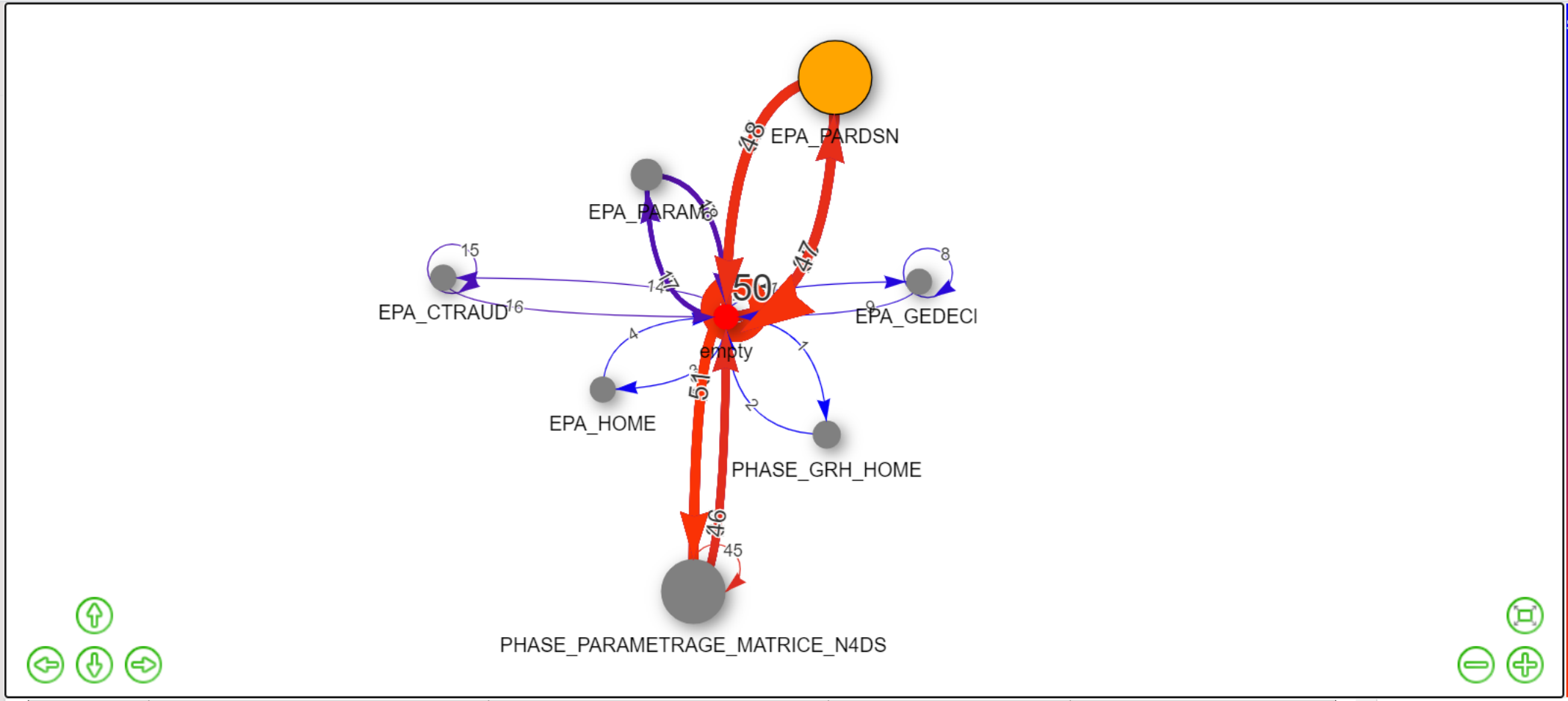
Animate edges!

Manipulate multiple users interactions:

Choose Multiple Users...

Filter nodes

- Visited phaseId
- All phaseId



NB EVENT	FROM PHASE_ID	TO PHASE_ID	DETAILS OF THE EVENT	DURATION OF THE EVENT
1	empty	PHASE_GRH_HOME	Selection de l onglet N°: 0	0.039 seconds
2	PHASE_GRH_HOME	empty	Selection de l onglet	9.957 seconds
3	empty	EPA_HOME	Selection de l onglet N°: 1	0.718 seconds

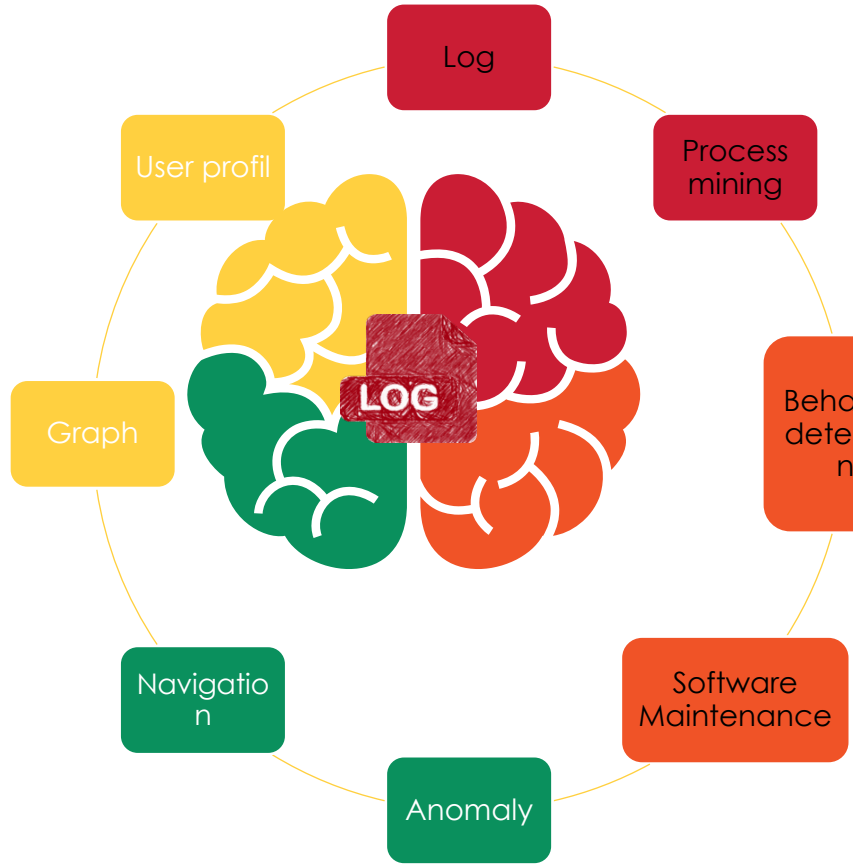
CodePhase's legend :

- Agent Module Codephases.
- Missing Codephases.
- Other Module CodePhases.

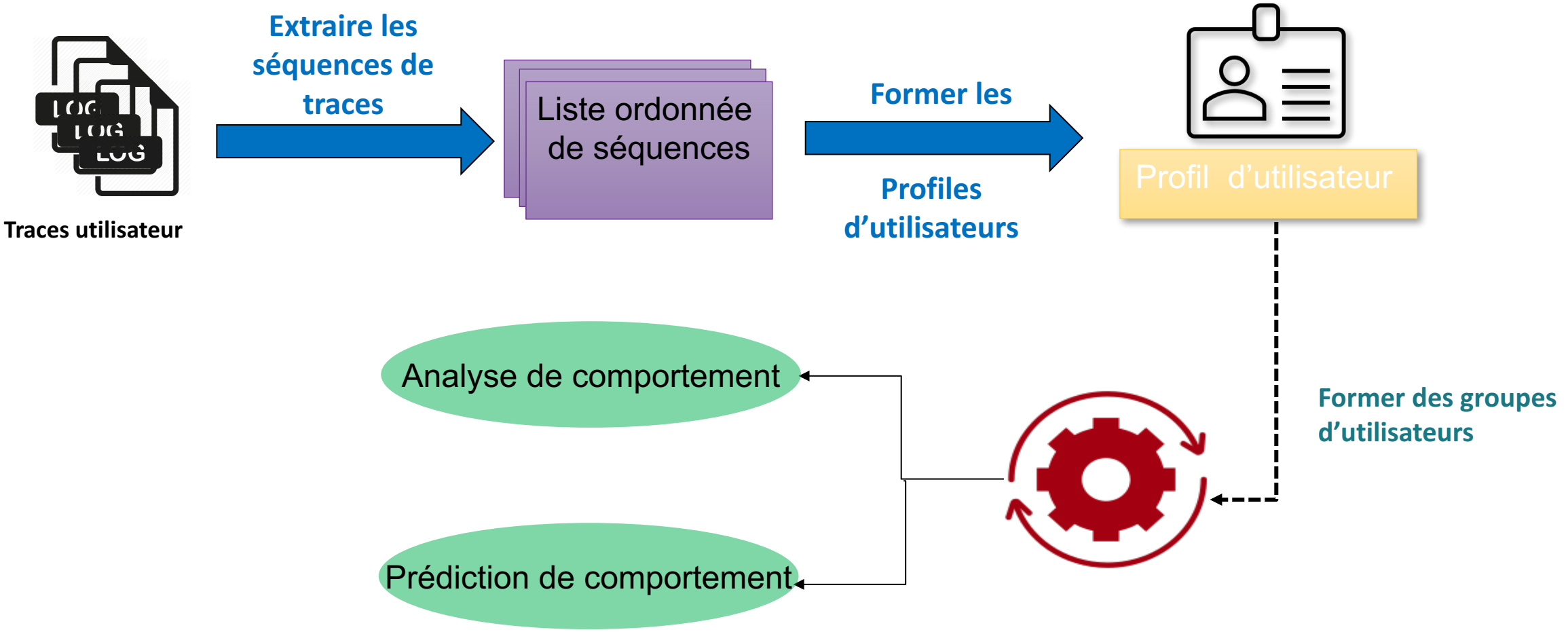
Et après?

Analyser les graphes de navigation utilisateur afin de:

1. Construire un profil utilisateurs!
2. Construire les groupes utilisateurs (les utilisateurs avec des graphes de navigation similaire = profil peut être similaire?)
3. Prédiction de prochaine activité utilisateurs selon les précédentes activités des utilisateurs de même groupe!



Pour récapituler:



Sachez bien que:

1. Une trace logicielle n'est pas toujours une trace de qualité!
2. Il peut y avoir des informations qui manque, des pages non tracée/ non testée donc pas de trace générée, .. etc
3. Un effort conséquent est nécessaire pour la phase de prétraitement!

Questions ?