

> EMULSION

**Vers des modèles épidémiologiques
transparents, flexibles et multi-échelles**

**10 novembre 2020 / BIOEPAR, INRAE /
Sébastien Picault**

INRAE

Oniris
ÉCOLE NATIONALE
VÉTÉINAIRE AGROALIMENTAIRE ET DE LA VÉTÉRINAIRE

**Région
PAYS DE LA LOIRE**

Modélisation multi-échelles de la propagation des pathogènes

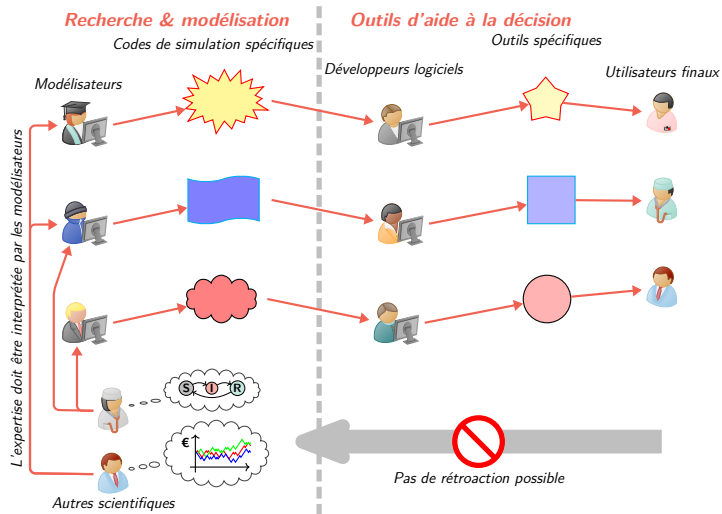
- ▶ intégrer des expertises et échelles multiples
- ▶ comparer des hypothèses par la modélisation
- ▶ proposer / évaluer des mesures de maîtrise
- ▶ innover : outils d'aide à la décision

Points de progression

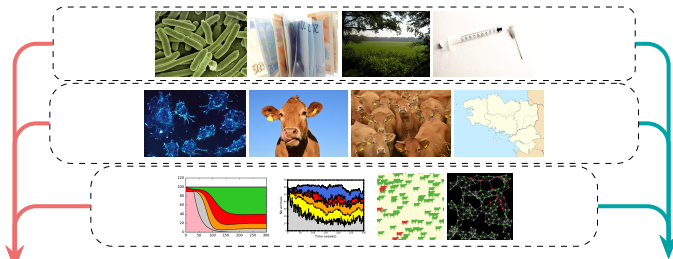
- ▶ simplifier et accélérer l'intégration logicielle
- ▶ maintenabilité, performance et évolutivité
- ▶ co-construction des modèles avec d'autres scientifiques



➤ Situation classique et limitations



➤ Développer des méthodes d'Intelligence Artificielle



connaissances
disciplinaires

échelles

paradigmes

Code ad hoc



Modélisateurs

Modèle explicite,
intelligible, révisable



représentation des
connaissances

Moteur générique
de simulation



système
multi-agents
multi-niveaux

Informaticiens

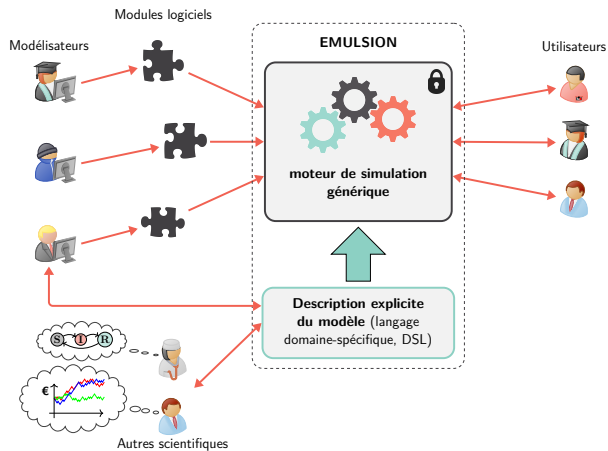
INRAE

EMULSION

10 novembre 2020, CATI IMOTEP/ BIOEPAR, DYNAMO/ Sébastien Picault



➤ L'approche EMULSION



Cadre générique pour la modélisation épidémiologique

- ▶ **Transparence**
hypothèses, structure du modèle, paramètres explicites
- ▶ **Lisibilité**
texte structuré → diagrammes
- ▶ **Révisabilité**
peu (pas) de code à écrire

→ développement accéléré et fiabilisé

[Picault & al. 2017a, 2019b]

Diffusion open source : <https://sourcesup.renater.fr/www/emulsion-public>

INRAE

EMULSION

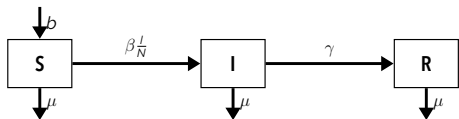
10 novembre 2020, CATI IMOTEP/ BIOEPAR, DYNAMO/ Sébastien Picault



➤ Processus : diagrammes de flux → machines à états

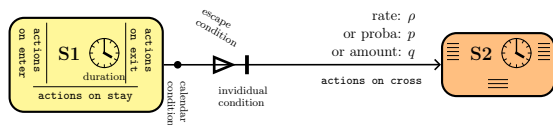
Transformation dans la représentation des connaissances

Diagrammes de flux



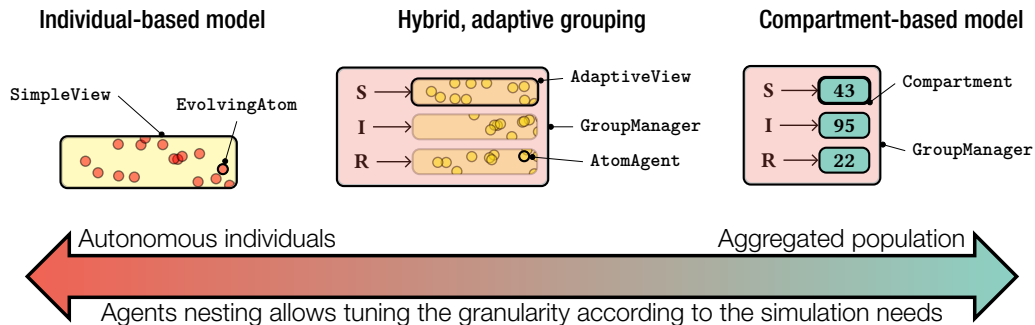
- 😊 représentation générique
- 😊 indépendant du paradigme (compartiments / IBM...)
- 😞 connaissances implicites
- 😞 mélange de domaines
- 😞 ajouts à l'implémentation

Machines à états



- 😊 une machine ↔ un processus
- 😊 flux entrants et sortants explicites
- 😊 durées, conditions, actions **individuelles** explicites
- 😊 description ↦ implémentation

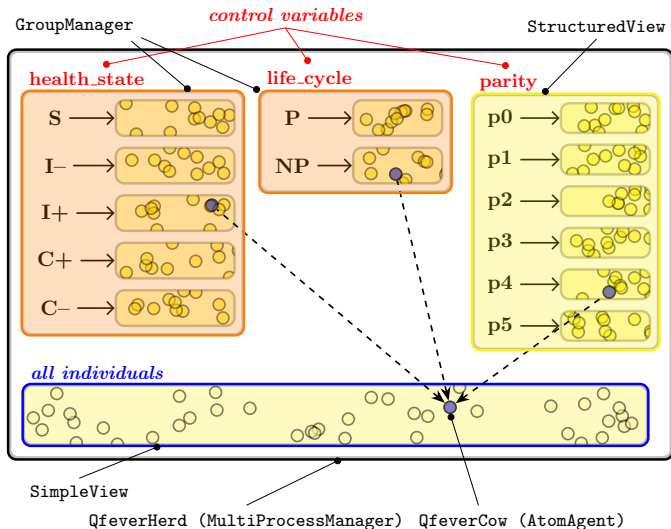
► Adapter la granularité des modèles aux besoins



- architecture multi-agents multi-niveaux
- organisation par **design patterns**

[Mathieu, Morvan, Picault 2018]

➤ Architecture modulaire, processus multiples



➤ Le framework EMULSION

- ▶ écrit en Python (4 000 lignes de code), diffusion open source (1.1):
<https://sourcesup.renater.fr/www/emulsion-public>
- ▶ documentation : tutoriels, exemples de modèles, API, rapport technique
- ▶ software paper, *PLOS Computational Biology* (2019)
- ▶ *work in progress* : ajout régulier de modules

Fonctionnalités (commande `emulsion`)

- ▶ exécution des simulations (`run`) avec graphiques simples (`plot`)
- ▶ représentations des modèles (`diagram`, `show`, `describe`)
- ▶ vérification syntaxique des modèles (`check`)
- ▶ génération de squelette de code spécifique (`generate`)



INRAE

EMULSION

10 novembre 2020, CATI IMOTEP/ BIOEPAR, DYNAMO/ Sébastien Picault



➤ Exemple : maladies respiratoires des jeunes bovins

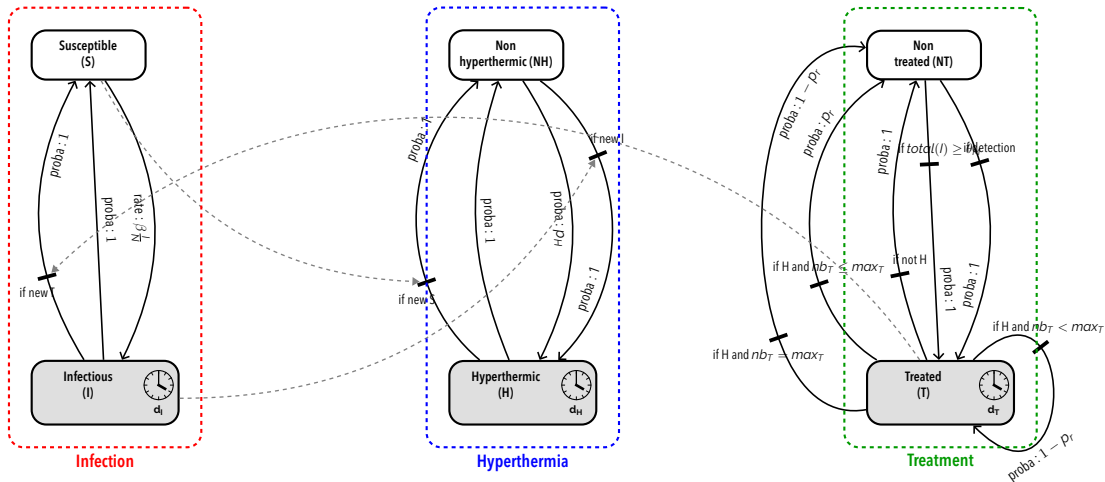
Bovine Respiratory Disease (BRD)

- ▶ multi-pathogènes (*M. haemolytica*, *M. bovis*...)
- ▶ lots de petite taille (10) → forte stochasticité
- ▶ retard à l'engraissement (→ pertes économiques)
- ▶ deux objectifs conjoints :
 - ▶ réduire la durée d'infection à l'échelle du lot
 - ▶ raisonner l'usage des antibiotiques
- ▶ évaluer et comparer des mesures de maîtrise
 - ▶ métaphylaxie
 - ▶ détection précoce (thermobolus)



[Picault, Ezanno, Assié 2019]

➤ Trois processus couplés



➤ Modèle EMULSION : extraits choisis

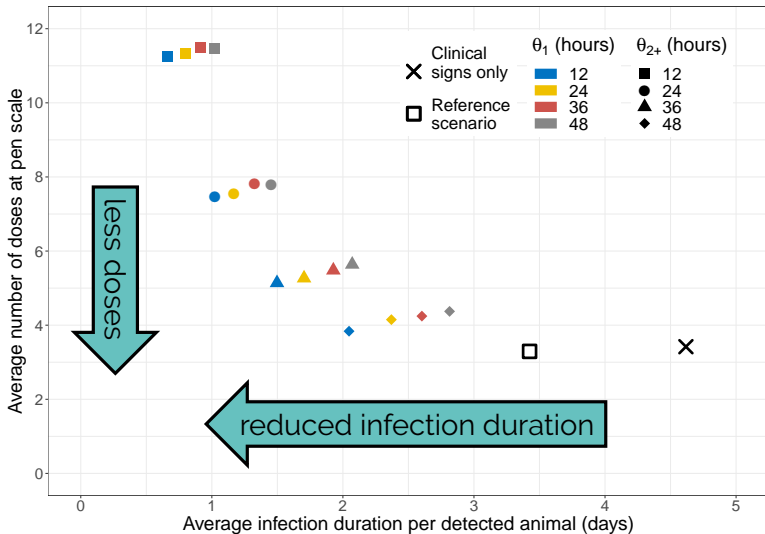
Processus (machines à états)

```
state_machines:
  health_state:
    desc: 'Description of the infectious process'
    states:
      - S:
          name: Susceptible
          desc: 'Suceptible of becoming infected'
      - I:
          name: Infectious
          desc: 'Infected and able to transmit
          pathogens. When becoming I, the animal
          also becomes hyperthermic (H), both
          states having the same duration. I
          animals have different shedding levels,
          depending on the pathogen. When becoming
          S again, the animal also returns to
          normal temperature'
          duration: dur_I
          on_enter:
            - become: hyperthermic
    transitions:
      - {from: S, to: I, rate:
          total_force_of_infection}
      - {from: I, to: S, proba: 1}
```

Paramètres et expressions

```
parameters:
  init_pen_size:
    desc: 'Initial size of the pen'
    value: 10
    source: 'consistent for a given pen type'
  total_force_of_infection:
    desc: 'The force of infection depends on the
    sum of individual infectiousness and exposure
    to general microbism of the herd'
    value: 'external_risk +
    (pen_force_of_infection / total_pen)'
  detection:
    desc: 'Condition for detecting sick
    animals. If clinical signs are visible, they
    are always taken into account. Otherwise,
    detection may rely either on the usage of an
    intrarumen bolus, or on the detection of
    rectal hyperthermia by farmer.'
    value: 'OR(has_clinical_signs,
            IfThenElse(has_bolus,
                        bolus_detection,
                        farmer_detection))'
```

➤ Durée d'infection vs. doses d'antibiotiques



➤ Autres applications : pathosystèmes variés

- ▶ Q fever, échelle département
- ▶ campylobactérioses
- ▶ **modèle de réintroduction de la brucellose bovine** (thèse Anses S. Mlala, dir. V. Hénaux)
très grand réalisme (conduite, tests, mouvements commerciaux)
- ▶ **challenge de modélisation PPA**
modèle détaillé avec contacts faune domestique/faune sauvage, spatialisé à large échelle
(1/4 France, 500 000 sangliers, 4700 élevages élevages) + diverses mesures de maîtrise
→ données épidémiologiques simulées dont les "joueurs" doivent prédire l'évolution et la maîtrise
- ▶ modèle BVD (laitier + allaitant + mixte) à l'échelle territoriale
- ▶ modèle RVF (appui thèse H. Cecilia)
- ▶ modèle SDRP (thèse V. Sicard)

→ **enrichissement DSL + moteur**



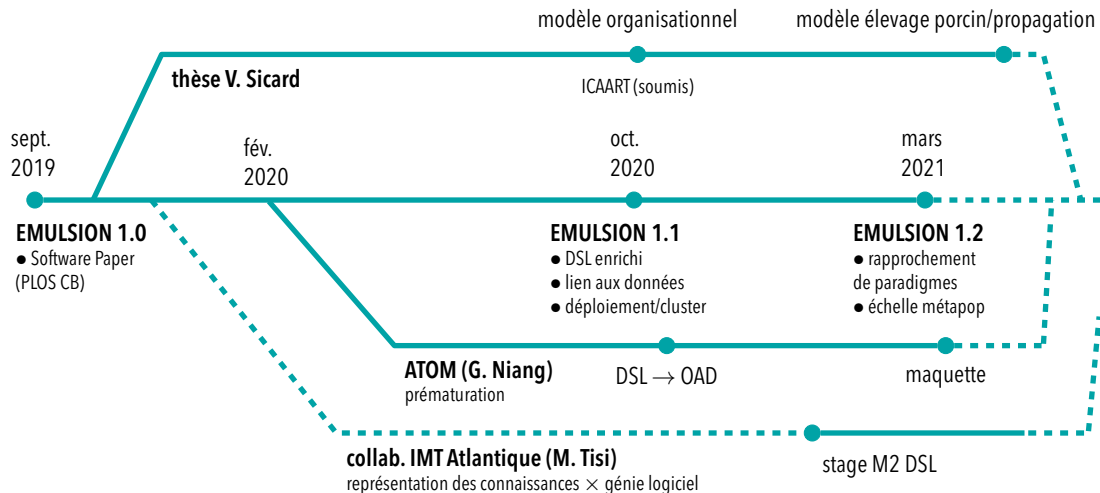
INRAE

EMULSION

10 novembre 2020, CATI IMOTEP/ BIOEPAR, DYNAMO/ Sébastien Picault

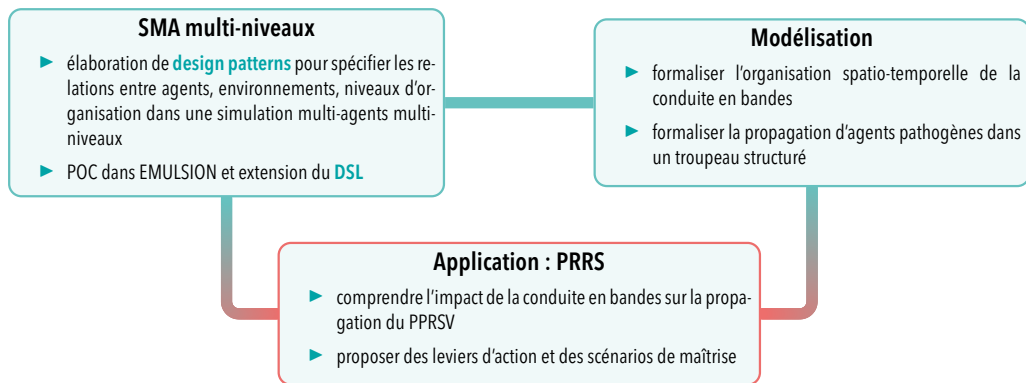


➤ Travail sur les méthodes informatiques



➤ Recherches SMA × modélisation, et applications

“Couplages entre agents, environnements et niveaux d’organisation dans les modèles mécanistes en épidémiologie. Application à la modélisation du syndrome dysgénésique et respiratoire porcin”



[Sicard, Andraud, Picault 2020 subm.]



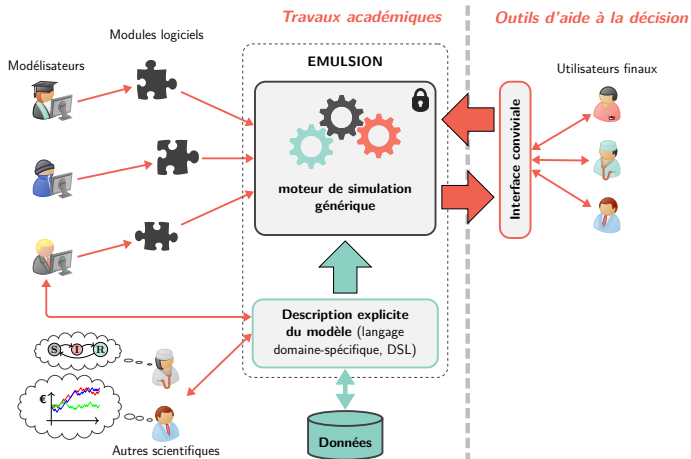
INRAE

EMULSION

10 novembre 2020, CATI IMOTEP/ BIOEPAR, DYNAMO/ Sébastien Picault



➤ Vers la production d'OAD



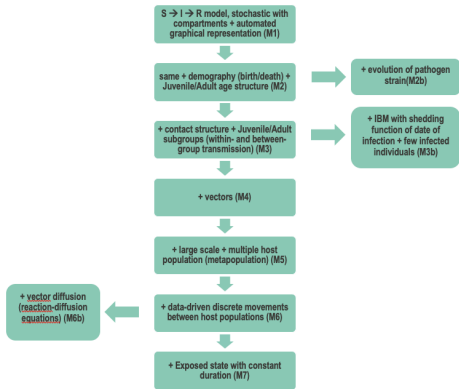
- ▶ homogénéité logicielle
- ▶ pilotage par le DSL

→ projet ATOM
(prématuration INRA)

- ▶ enrichissement du DSL
- ▶ génération d'un outil web
- ▶ méthodes pour la génération de code

➤ Comparer les méthodes et plateformes : benchmark EU

Modèles de complexité graduée



Trois autres plateformes

1. Broadwick (Roslin Institute) – Java
2. Kendrick (IRD, France/Vietnam) – Pharo
3. SimInf (SVA, Suède) – R

- ▶ lisibilité des modèles?
- ▶ "effort" de développement?
- ▶ convergences / verrous?
- ▶ performances?


Appel à participation !

➤ **Merci pour votre attention !**

site web d'EMULSION :

<https://sourcesup.renater.fr/www/emulsion-public>

suivez nos travaux !

 bioepar_dynamo



INRAE

EMULSION

10 novembre 2020, CATI IMOTEP/ BIOEPAR, DYNAMO/ Sébastien Picault

