

Tests à données aléatoires par mutations pour les systèmes de programmation par contraintes

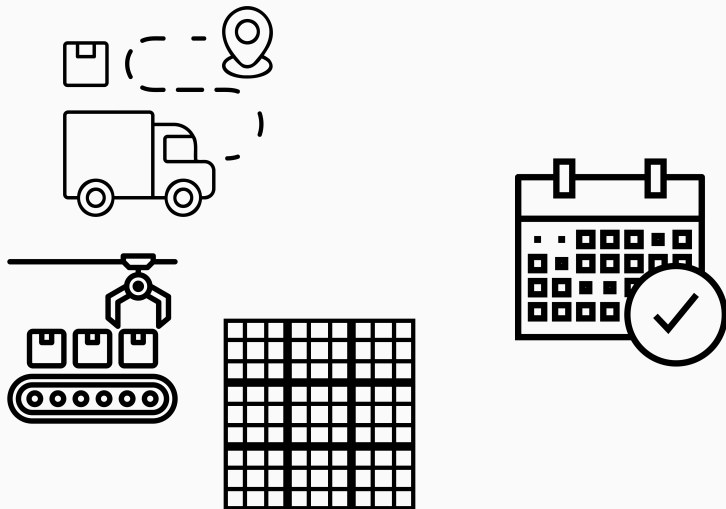
Wout Vanroose¹, Ignace Bleukx¹, Jo Devriendt¹, Dimos Tsouros¹, **Hélène Verhaeghe**^{1,2}, Tias Guns¹

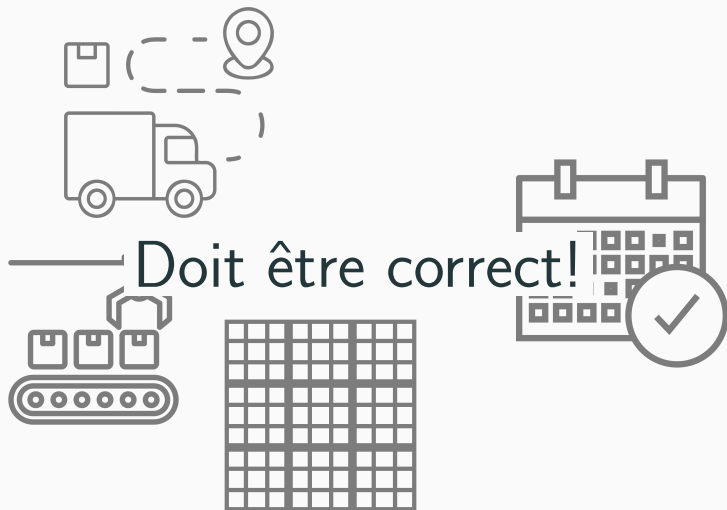
2 Juillet 2025

¹ KULeuven, Leuven, Belgique

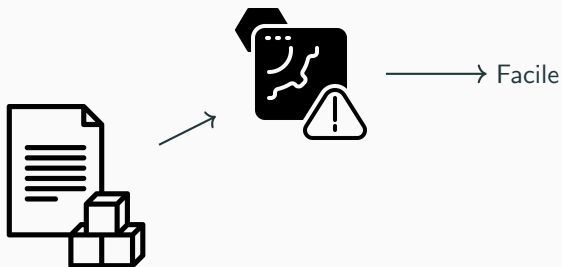
² UCLouvain, Louvain-la-Neuve, Belgique, helene.verhaeghe@uclouvain.be

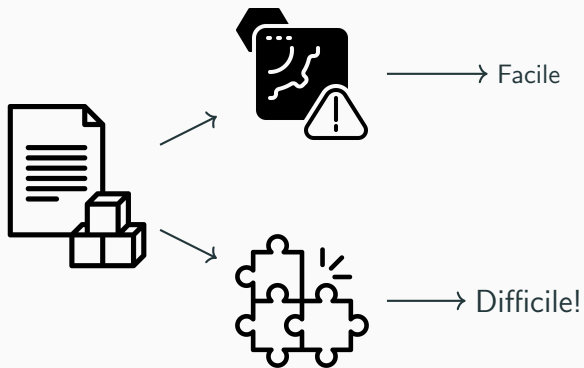






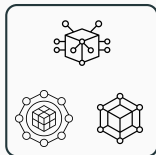




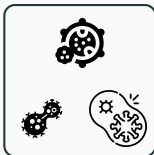


Hurricane

Modèles de base



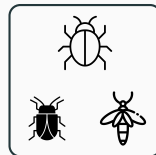
Mutations



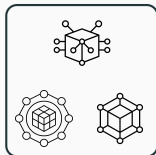
Vérifications



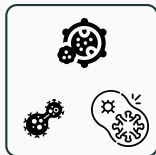
A déboguer



Modèles de base



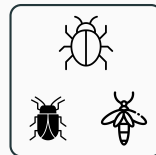
Mutations



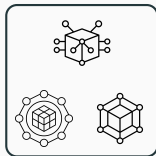
Vérifications



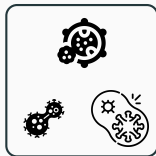
A déboguer



Modèles de base



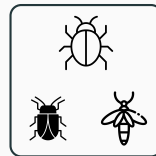
Mutations



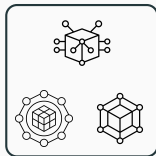
Vérifications



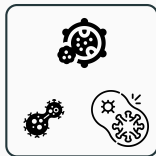
A déboguer



Modèles de base



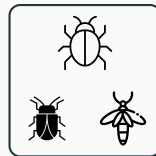
Mutations



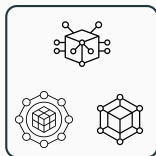
Vérifications



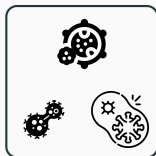
A déboguer



Modèles de base



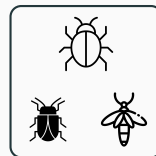
Mutations



Vérifications



A déboguer

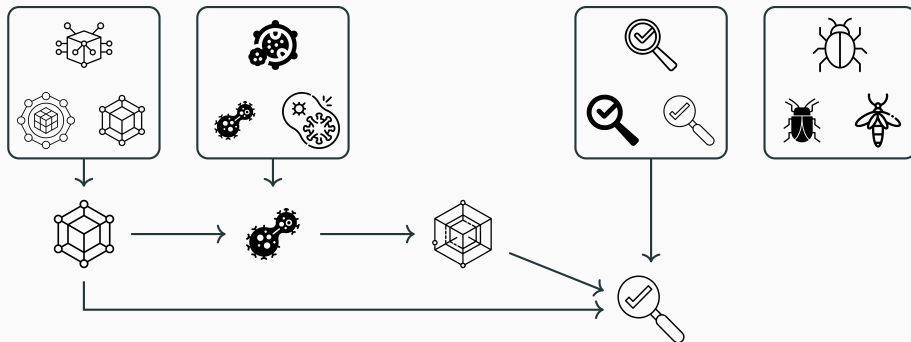


Modèles de base

Mutations

Vérifications

A déboguer

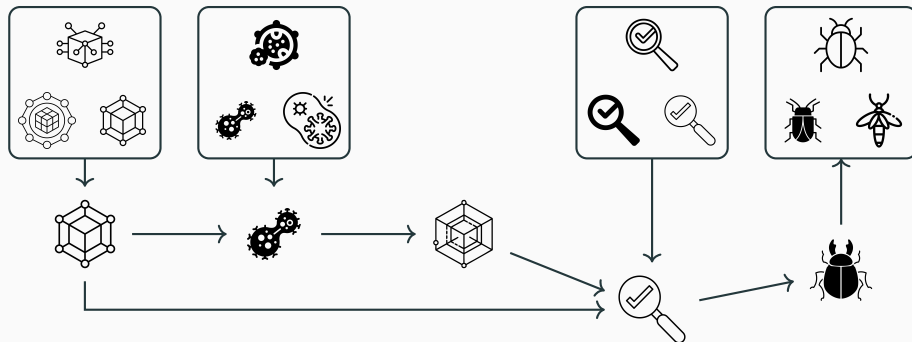


Modèles de base

Mutations

Vérifications

A déboguer

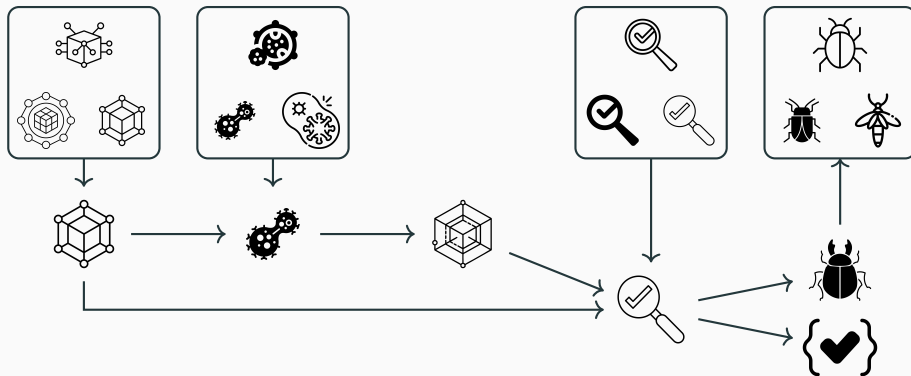


Modèles de base

Mutations

Vérifications

A déboguer

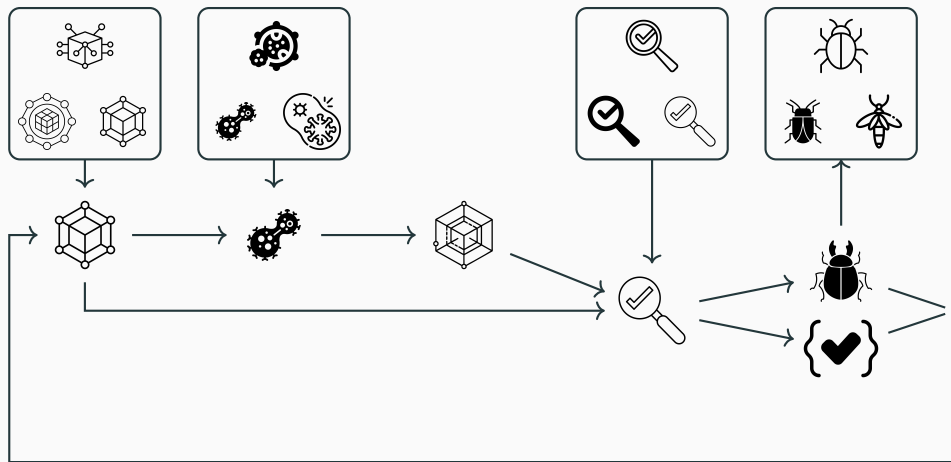


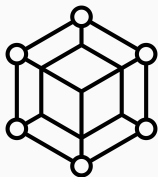
Modèles de base

Mutations

Vérifications

A déboguer

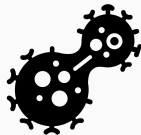




Modèles des tests unitaires (avec min 1 contrainte et 1 solution) :

- Couvrent (normalement) toutes les modélisations possibles
- Petits modèles
 - Facile à comprendre
 - Facile à résoudre
 - Facile à trouver la source du bogue
 - Facile à répliquer

But : Préserver certaines propriétés vérifiables (i.e., satisfiabilité du problème)



Différentes mutations :

- mutations haut-niveau
- mutations de sous-expressions
- mutations de reformulation

contraintes	nom	mutation
a	NEG	$a \wedge \neg(\neg a)$
a, b	AND	$(a \wedge b) \wedge \neg(a \wedge \neg b) \wedge$ $\neg(\neg a \wedge b) \wedge \neg(\neg a \wedge \neg b)$
a, b	OR	$(a \vee b) \wedge (a \vee \neg b) \wedge$ $(\neg a \vee b) \wedge \neg(\neg a \vee \neg b)$
a, b	IMPL	$(a \Rightarrow b) \wedge (\neg a \Rightarrow b) \wedge (b \Rightarrow a) \wedge$ $(\neg b \Rightarrow a) \wedge \neg(a \Rightarrow \neg b) \wedge (\neg a \Rightarrow \neg b) \wedge$ $\neg(b \Rightarrow \neg a) \wedge (\neg b \Rightarrow \neg a)$
a, b	XOR	$(a \oplus \neg b) \wedge (\neg a \oplus b) \wedge$ $\neg(a \oplus b) \wedge \neg(\neg a \oplus \neg b)$



Mutation par constantes

$$x \geq y$$

$$x = y$$

Mutation par constantes

$$5 \times x \geq y \times 5$$

$$7 + x = y + 7$$

Mutation par constantes

$$5 \times x \geq y \times 5$$

$$7 + x = y + 7$$

Fusion sémantique

$$7 \times (x + y) = z \quad \min(p, q) \geq 3$$

Mutation par constantes

$$5 \times x \geq y \times 5$$

$$7 + x = y + 7$$

Fusion sémantique

$$7 \times \underline{(x + y)} = z \quad \underline{\min(p, q)} \geq 3$$

Mutation par constantes

$$5 \times x \geq y \times 5$$

$$7 + x = y + 7$$

Fusion sémantique

$$\begin{array}{ccc} 7 \times \underline{(x + y)} = z & & \underline{\min(p, q)} \geq 3 \\ \searrow & & \downarrow \\ v = \underline{(x + y)} + \underline{\min(p, q)} \end{array}$$

Mutation par constantes

$$5 \times x \geq y \times 5$$

$$7 + x = y + 7$$

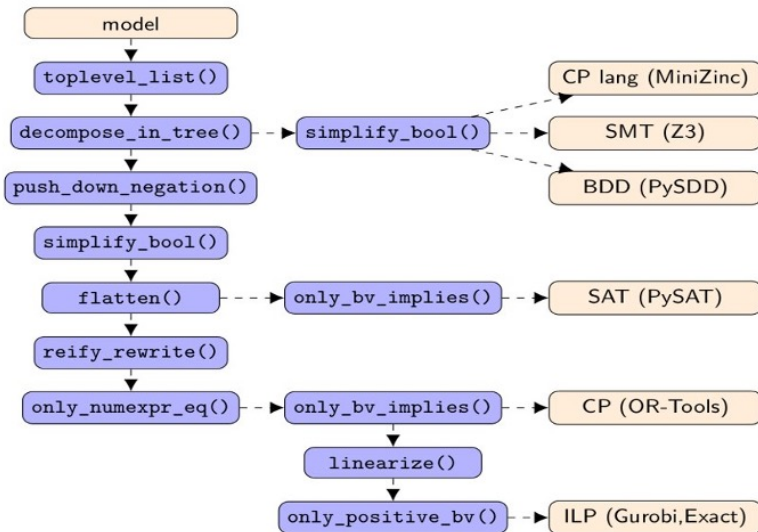
Fusion sémantique

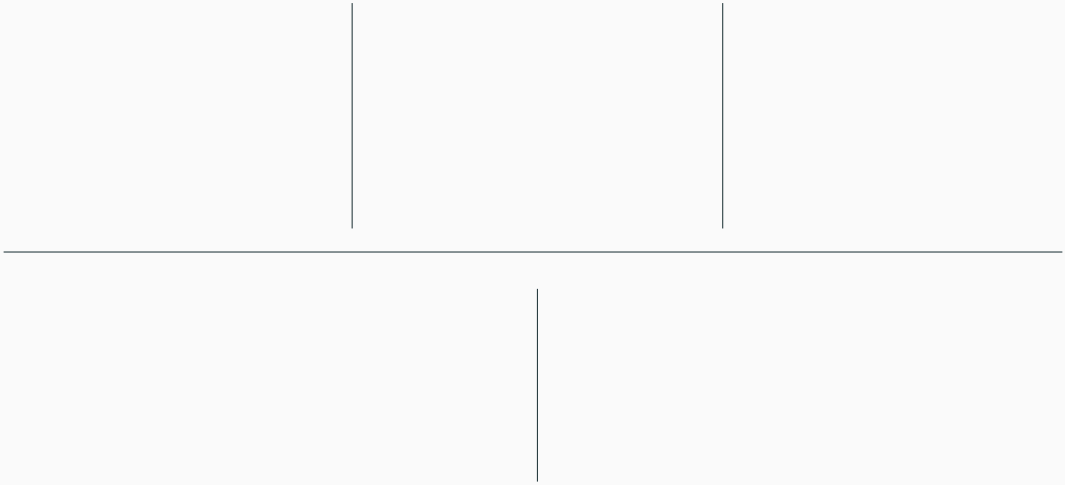
$$7 \times \underline{(x + y)} = z \qquad \underline{\min(p, q)} \geq 3$$


$$v = \underline{(x + y)} + \underline{\min(p, q)}$$

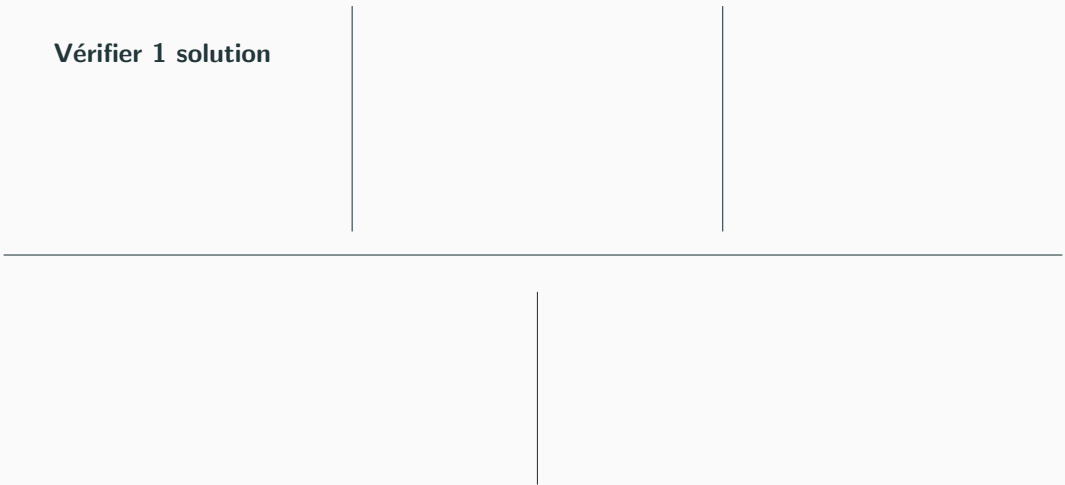
$$7 \times \underline{(v - \min(p, q))} = z$$

$$\underline{v - (x + y)} \geq 3$$

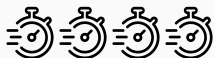




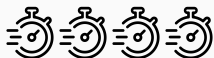
Vérifier 1 solution



Vérifier 1 solution

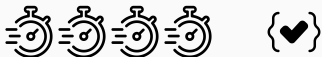


Vérifier 1 solution

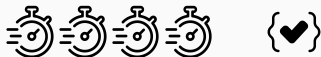


Vérifier la satisfiabilité

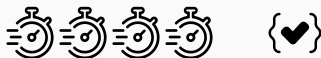
Vérifier 1 solution



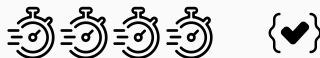
Vérifier la satisfiabilité



Vérifier 1 solution

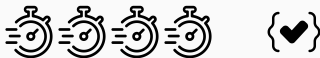


Vérifier la satisfiabilité

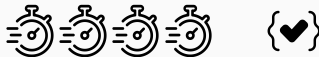


Vérifier l'optimal

Vérifier 1 solution



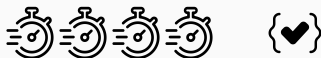
Vérifier la satisfiabilité



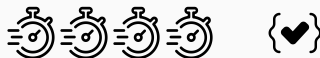
Vérifier l'optimal



Vérifier 1 solution



Vérifier la satisfiabilité

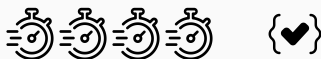


Vérifier l'optimal

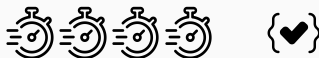


Vérifier le nombre de solutions

Vérifier 1 solution



Vérifier la satisfiabilité



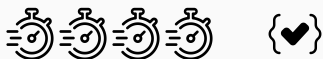
Vérifier l'optimal



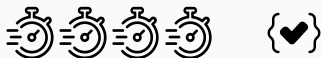
Vérifier le nombre de solutions



Vérifier 1 solution



Vérifier la satisfiabilité



Vérifier l'optimal

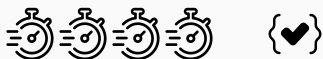


Vérifier le nombre de solutions

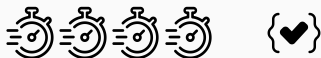


Vérifier l'équivalence

Vérifier 1 solution



Vérifier la satisfiabilité



Vérifier l'optimal



Vérifier le nombre de solutions



Vérifier l'équivalence



Résultats

10 Open ✓ 29 Closed		Author	Label	Projects	Milestones	Assignee	Sort
☐	☒	linearize alters number of solutions	bug	fast-test			1
#460 opened 4 days ago by Wouter							
☐	☒	flattening very slow for specific models	fast-test				
#589 opened last week by Wouter							
☐	☒	gurobi does not support floor division	bug	fast-test			
#598 opened 2 weeks ago by Wouter							
☐	☒	Xor decomposition is very inefficient	fast-test				
#585 opened on Jul 20 by Wouter							
☐	☒	relief element constraints do not get handled correctly	bug	fast-test			1
#582 opened on Jul 11 by Wouter							
☐	☒	Minizinc solve call crashes JSON-decoder on specific output	fast-test	upstream bug			3
#586 by Wouter was closed on Jul 13							
☐	☒	Negating Boolval() changes the original object	bug	fast-test			1
#571 by Wouter was closed on Jul 3							
☐	☒	Count only works in a comparison for minizinc	bug	fast-test			1
#570 by Wouter was closed on Jul 4							
☐	☒	possibility for infinite loop in only_bv_implies	fast-test				1
#569 opened on Jun 30 by Wouter							
☐	☒	boolval implies leads to error	bug	fast-test			3
#568 by Wouter was closed on Jul 7							
☐	☒	Reify_rewrite documentation missing precondition	fast-test				
#566 by Wouter was closed on Apr 26							
☐	☒	precondition of linearize_constraint not clear	fast-test				3
#567 by Wouter was closed on Apr 26							
☐	☒	Xor of Circuit global constraints crashes oracles	fast-test				2
#561 by Wouter was closed on Jun 19							
☐	☒	calling _intVarimpl makes model unsat	fast-test				4
#562 opened on Mar 27 by Wouter							
☐	☒	negated_normal crashes when input contains a boolexpr compared with an numsExpr	fast-test				3
#561 by Wouter was closed on Apr 21							
☐	☒	Table constraint on right hand side of implication crashes linearize_constraint	fast-test				2
#560 by Wouter was closed on May 16							
☐	☒	Element global constraint accepting lists leads to error in linearize_constraint	fast-test				1
#559 by Wouter was closed on May 16							
☐	☒	Comparison expression has no get_bounds	fast-test				1
#555 by Wouter was closed on Mar 27							
☐	☒	Wrongly unsatisfiable problem for oracles and gurobi	fast-test				2
#570 by rubenknip was closed on Nov 24, 2022							
☐	☒	wrongly unsat result on gurobi	fast-test				2
#560 by rubenknip was closed on Nov 28, 2022							
☐	☒	MiniZinc error: Memory violation detected (segmentation fault)	bug	fast-test			1
#563 by rubenknip was closed on Mar 15							
☐	☒	malloc(): Not enough space in some MiniZinc solvers	bug	fast-test			6
#564 by rubenknip was closed on Mar 15							
☐	☒	naming a variable starting with '-' crashes all MiniZinc's solvers	bug	fast-test			3
#562 by rubenknip was closed on Jan 28							
☐	☒	from_file seems to create two separate references for the same variable	fast-test				2
#561 opened on Nov 18, 2022 by rubenknip							

- Système testé : CPMpy
- Nombre de bogues trouvés : 52

#mutations	OR-tools			MiniZinc/Gecode			Total
	#models	#errors	#unique	#models	#errors	#unique	#unique
1	9166418	5747	1	218377	289	3	3
2	6672588	11002	3	216527	723	6	6
5	2270441	8975	5	128884	1495	8	11
10	344710	2783	7	57191	423	9	13

Verification	OR-tools			MiniZinc/Gecode			Total
	#models	#errors	#unique	#models	#errors	#unique	#unique
Toutes solutions	13441	460	4	11167	312	7	8
Nb de solution	14551	539	5	11623	325	6	8
Une solution	4095185	25695	5	194495	1983	8	10
Satisfaction	3679400	180	4	186119	116	5	8
Optimal	10651580	1633	2	217575	194	3	4

Conclusion

- Système de vérification basé sur un vote entre plusieurs solveur
 - détection de comportement inconsistant (i.e., gestion de la division)
- Mutations qui introduisent des contraintes globales
- Ajout d'autres mutations



- Nouvel outil Hurricane
- Conçu pour fonctionner en continu
- Modulaire

Merci pour votre attention!

Des questions?

<https://hverhaeghe.bitbucket.io/>