Towards Self-Adaptable Languages

Gwendal Jouneaux [1] Olivier Barais [1]
Benoit Combemale [1] Gunter Mussbacher [2]

[1]Univ. Rennes, Inria, IRISA – Rennes, France

[2]McGill University – Montreal, Canada

ALE Seminar — May 24, 2022

Context

Software ...

Context

Software ...

- Evolve in complex/changing environment (e.g, Cloud, embedded systems)

Context

Software ...

- Evolve in complex/changing environment (e.g, Cloud, embedded systems)
- Need dynamic adaptation to best deliver the service (e.g., Waymo[1], Netflix[1])

[1] Cf. `https://waymo.com`, `https://www.netflix.com`

Context

Software ...

- Evolve in complex/changing environment (e.g, Cloud, embedded systems)
- Need dynamic adaptation to best deliver the service (e.g., Waymo[1], Netflix[1])

Software languages ...

[1] Cf. `https://waymo.com`, `https://www.netflix.com`

Context

Software ...

- Evolve in complex/changing environment (e.g, Cloud, embedded systems)
- Need dynamic adaptation to best deliver the service (e.g., Waymo[1], Netflix[1])

Software languages ...

- Can abstract concerns into high level constructs (e.g., memory management)

[1] Cf. `https://waymo.com`, `https://www.netflix.com`

Context

Software ...

- Evolve in complex/changing environment (e.g, Cloud, embedded systems)
- Need dynamic adaptation to best deliver the service (e.g., Waymo[1], Netflix[1])

Software languages ...

- Can abstract concerns into high level constructs (e.g., memory management)

Vision : abstract self-adaption into high level language constructs

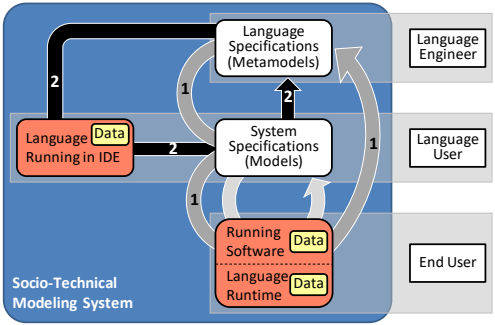
[1] Cf. `https://waymo.com`, `https://www.netflix.com`

What is a Self-Adaptable Language ?

" *A software language that abstracts the design and execution of feedback loops in the design-time environment and the run-time environment* "

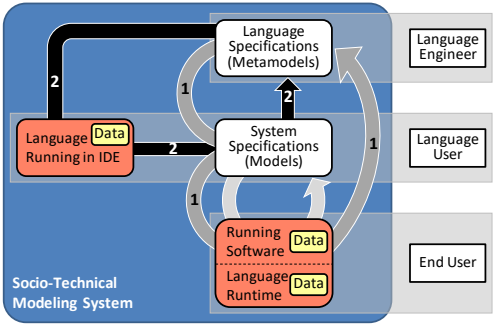
1. Free the language user from the implementation of :
   - The feedback loop
   - The trade-off analysis
2. Allow continuous and automatic evolution of itself

L-MODA | Languages, Models, and Data



L-MODA Conceptual Framework for Self-Adaptable Languages

L-MODA | Languages, Models, and Data

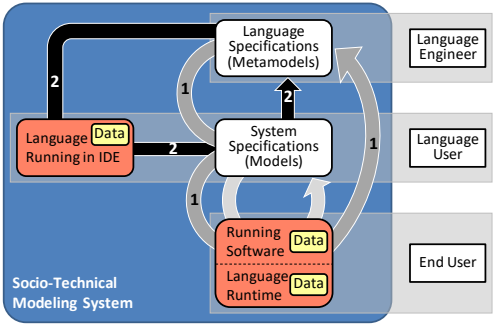


L-MODA Conceptual Framework for Self-Adaptable Languages

1) Runtime Feedback Loop

Use run-time data, model & metamodel
→ adaptation of language semantics

L-MODA | Languages, Models, and Data



L-MODA Conceptual Framework for Self-Adaptable Languages

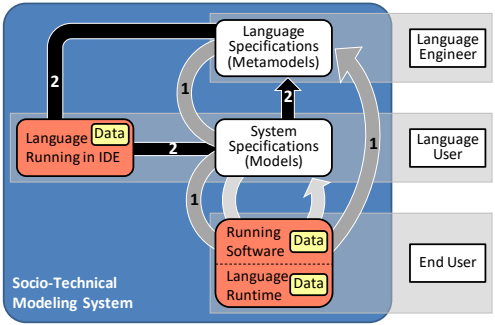
1) Runtime Feedback Loop

Use run-time data, model & metamodel
→ adaptation of language semantics

2) Design Feedback Loop

Use design-time data, models & metamodel
→ adaptation of syntax, pragmatics & semantics

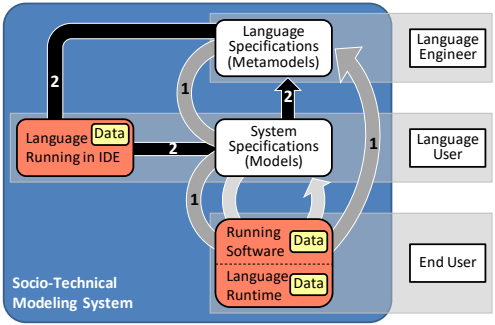
L-MODA | Stakeholders



L-MODA Conceptual Framework for Self-Adaptable Languages

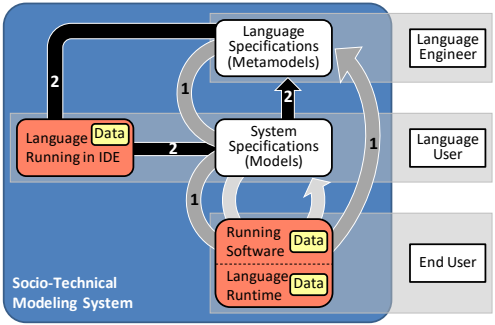
L-MODA | Stakeholders

Various uses of the feedback loops ...



L-MODA Conceptual Framework for Self-Adaptable Languages

L-MODA | Stakeholders



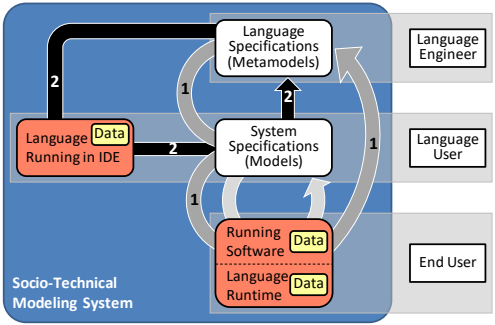
L-MODA Conceptual Framework for Self-Adaptable Languages

Various uses of the feedback loops ...

Examples for the Runtime Feedback Loop :

Delegation of responsabilities

L-MODA | Stakeholders



L-MODA Conceptual Framework for Self-Adaptable Languages

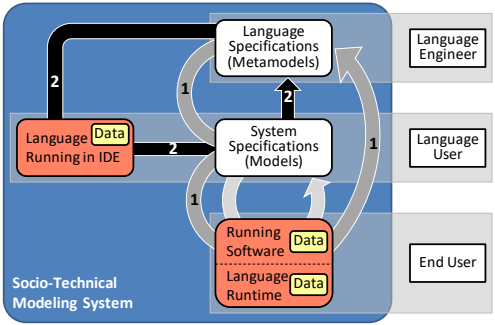
Various uses of the feedback loops ...

Examples for the Runtime Feedback Loop :

> Language engineer in complete control
> Tailor the language to a particular trade-off

Delegation of responsabilities

L-MODA | Stakeholders



L-MODA Conceptual Framework for Self-Adaptable Languages

Various uses of the feedback loops ...

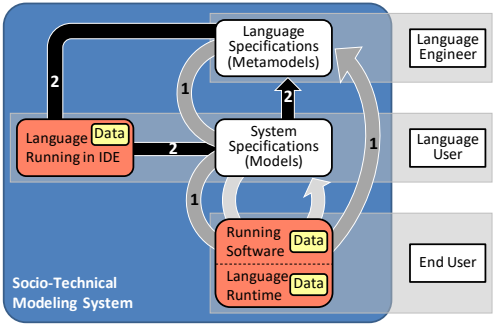
Examples for the Runtime Feedback Loop :

Language engineer in complete control
Tailor the language to a particular trade-off

Language user custom adaptations
Configure the adaptations for a system

Delegation of responsabilities

L-MODA | Stakeholders



L-MODA Conceptual Framework for Self-Adaptable Languages

Various uses of the feedback loops ...

Examples for the Runtime Feedback Loop :

Language engineer in complete control
Tailor the language to a particular trade-off

Language user custom adaptations
Configure the adaptations for a system

End-user preferences
Indicate preference for trade-offs

Delegation of responsabilities

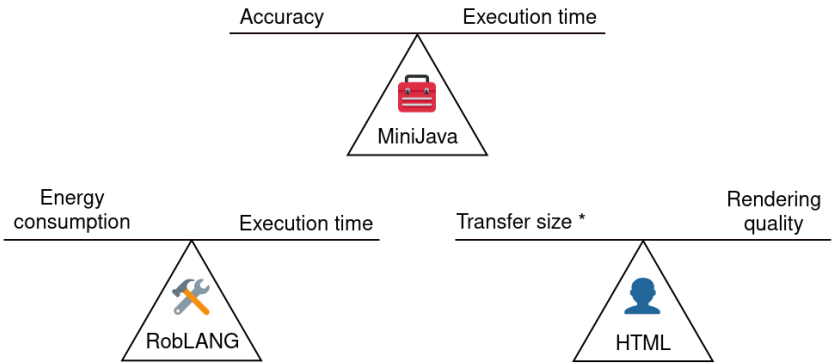
Experimentation

The case of Self-Adaptable Virtual Machines

What are Self-Adaptable Virtual Machines

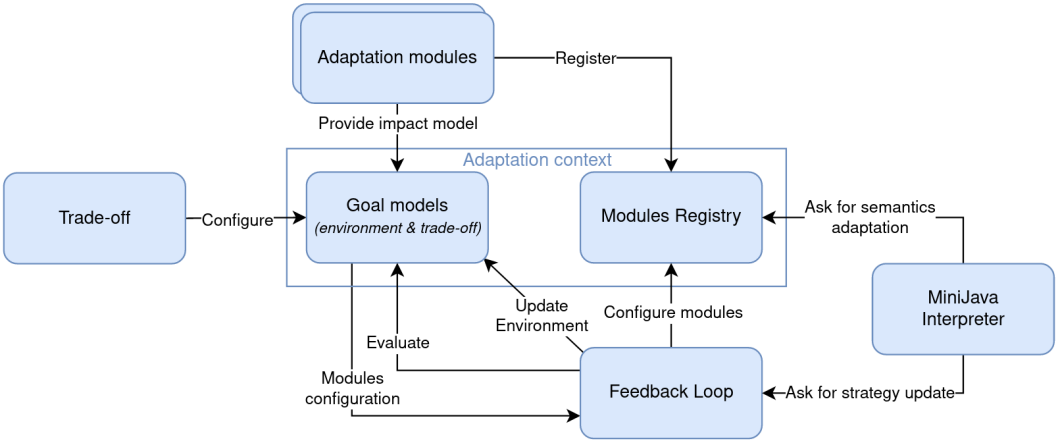
- A specific case of Self-Adaptable Languages
- Runtime Feedback loop in language operational semantics
- *In our experiment :* Pluggable architecture with delegation of responsibilities

Motivating Examples

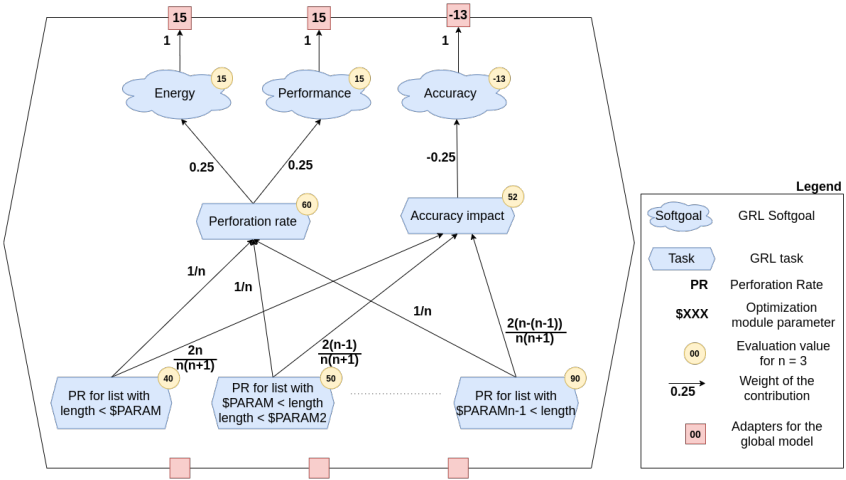


* Transfer size is proportional to energy consumption (Cf. `https://www.websitecarbon.com/`)

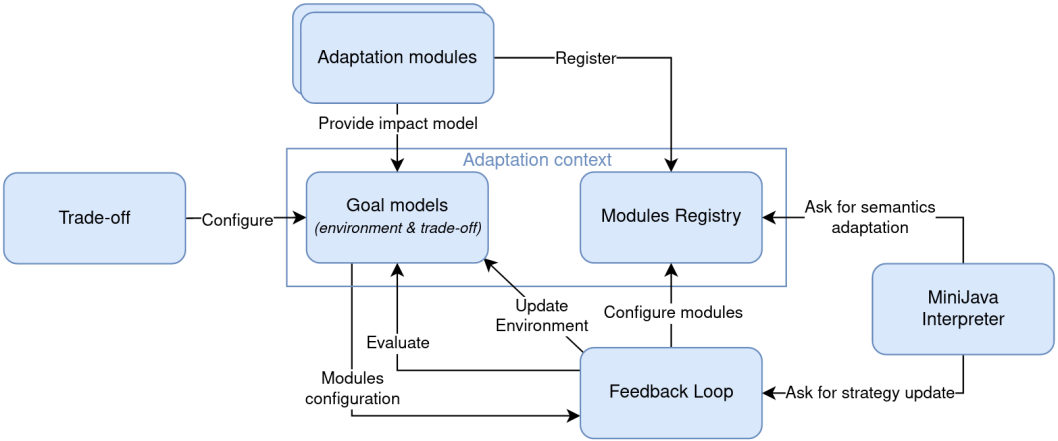
Architecture of Self-Adaptable Virtual Machines (MiniJava)



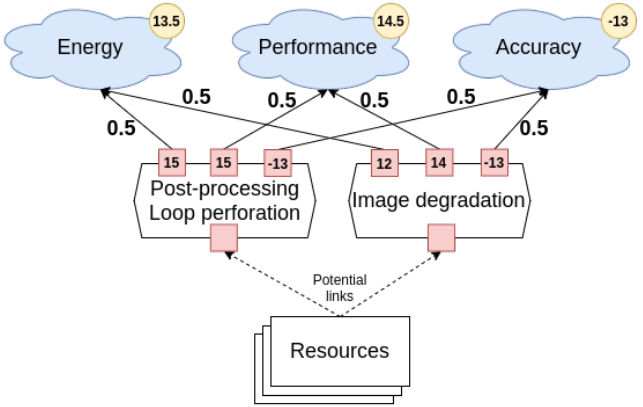
Impact Models

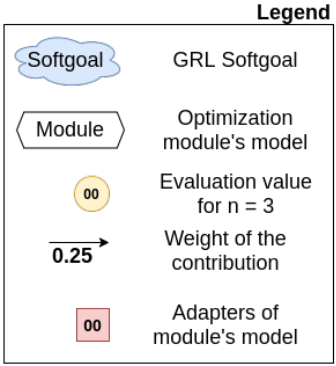


Architecture of Self-Adaptable Virtual Machines (MiniJava)

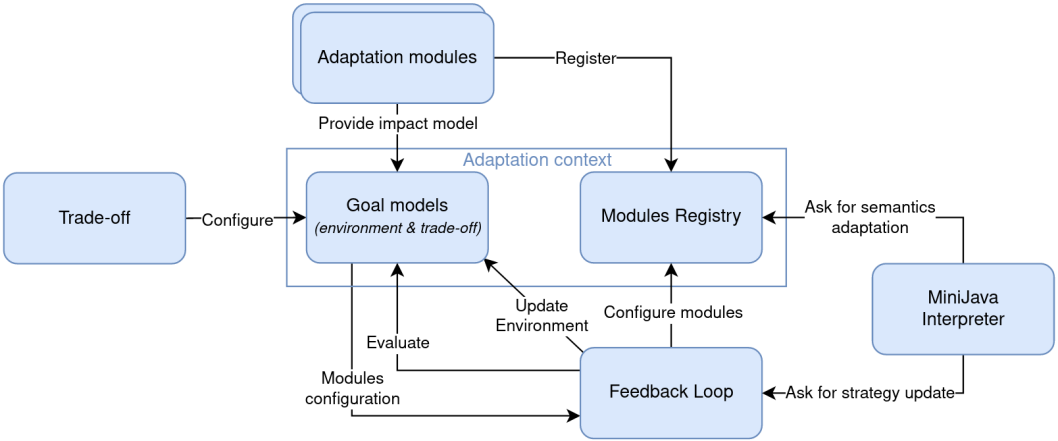


Global Goal Model





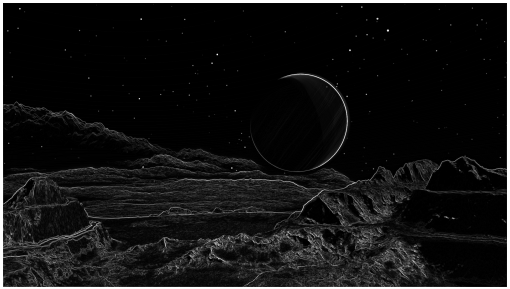
Architecture of Self-Adaptable Virtual Machines (MiniJava)



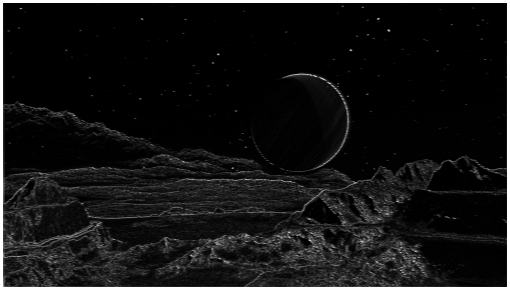
Adaptations

Adaptations (MiniJava)

Applied Approximate Loop Unrolling [1] on image processing algorithm (Sobel)



Standard output



Approximated output

[1] M. Rodriguez-Cancio, B. Combemale, and B. Baudry, "Approximate loop unrolling," in *Proceedings of the 16th ACM International Conference on Computing Frontiers*, ACM, 2019

Adaptations (RobLANG)

Applied a motor speed reduction on basic actions

Rational : $P_i = P_{max}(\frac{Speed_i}{Speed_{max}})^3$ [2]

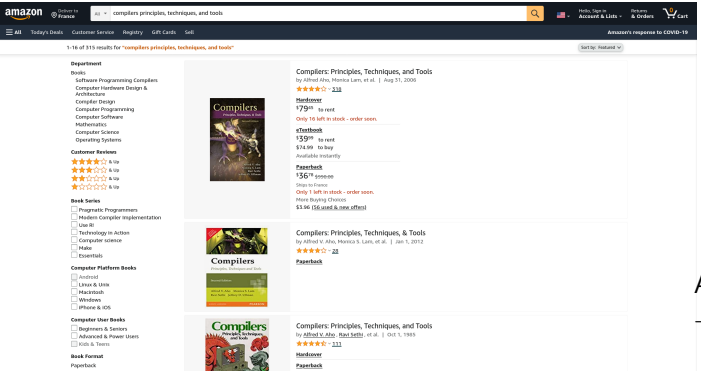
Three programs studied :

- Move forward/backward
- Turn left/right
- Combination of moves and turns (square patterns)

---

[2] A. Al-Mofleh, S. Taib, W. Salah, *et al.*, "Importance of energy efficiency: From the perspective of electrical equipments," in *Proceedings of the 2nd International Conference on Science and Technology (ICSTIE)*, 2008

Adaptations (HTML)

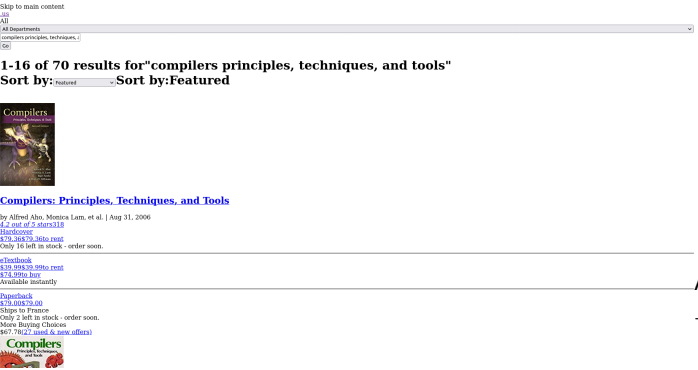


Standard website

- Conditional loading of resources
- Perforation of HTML lists
- Image degradation

Applied on the top 100 websites
$\rightarrow$ 45 still deliver the content

Adaptations (HTML)



Adapted website

- Conditional loading of resources
- Perforation of HTML lists
- Image degradation

Applied on the top 100 websites
$\rightarrow$ 45 still deliver the content

Evaluate the relevance of proposed adaptation

TL;DR : Good results but ...

- Correct adaptations of MiniJava
- Up to 10x more actions on RobLANG
- Energy reduction from -8.7% to 97.2% with a mean of 63.8% [54.2%, 73.4%]
- Performance overhead
- Lack of control on the adaptations
- Deal with the diversity of programs oblivious of the adaptations performed

Compare Language-level vs System-level abstractions

Attempt to implement as library of the language

VMs	Feedback loop	Trade-off reasoning	Feedback loop calls	Interaction with the domain
MiniJava	=	=	+	+
RobLang	=	+ +	+	-
HTML (JS)	=	=	=	-

Comparison of implementation simplicity (+ in favor of language-level)

Conclusion

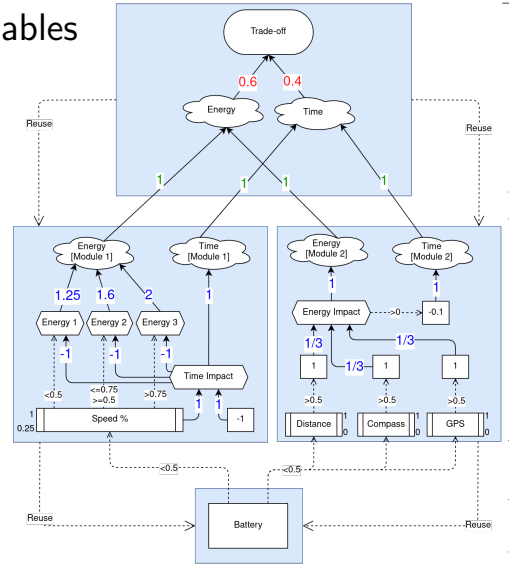
- The concept of Self-Adaptable Language and its conceptual framework
- A framework to implement Self-Adaptable Virtual Machines
- Ongoing work on specification of adaptive semantics

Perspectives

- DSLs to specify adaptations and impacts
- Formalisation of adaptive operational semantics
- Testing and debugging programs defined using SALs

Thanks for your attention !

Goal Model with Variables



References


M. Rodriguez-Cancio, B. Combemale, and B. Baudry, "Approximate loop unrolling," in *Proceedings of the 16th ACM International Conference on Computing Frontiers*, ACM, 2019.


A. Al-Mofleh, S. Taib, W. Salah, and M. Azizan, "Importance of energy efficiency: From the perspective of electrical equipments," in *Proceedings of the 2nd International Conference on Science and Technology (ICSTIE)*, 2008.