Fine-grained Language Composition

Carl Friedrich Bolz

Edd Barrett

Lukas Diekmann

Laurence Tratt

KING'S College LONDON

Software Development Team

March 15, 2016

Our problem

Languages get **better** but also **bigger**.

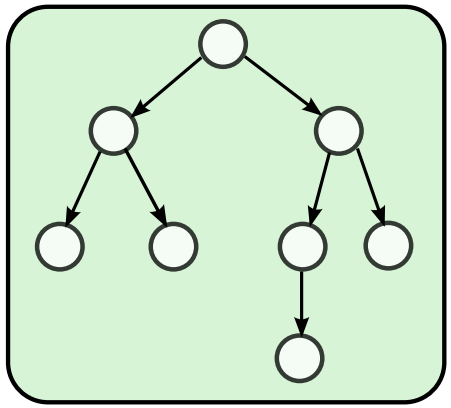
Language composition

Foreign Function Interface

Too **coarse** and typically high-level to low-level.

Language Composition Challenges

Parsing



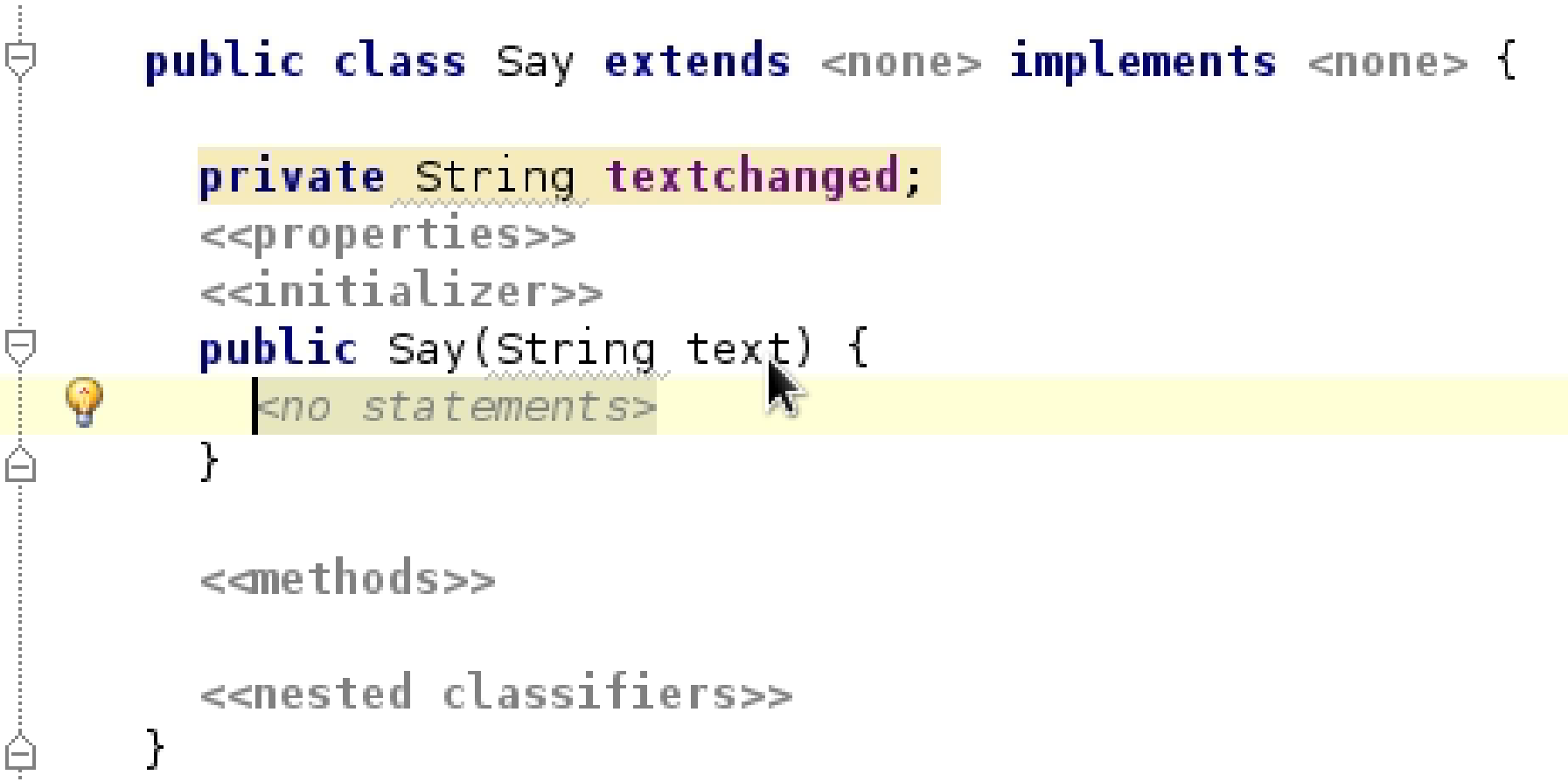
Running

```
SUB     AX,AX
MOV     ES,AX
SUB     BH,BH
MOV     BL,INT_NUMBER
SHL     BX,1
SHL     BX,1
MOV     DI,ES:[BX]
MOV     ES,ES:[BX+2]
ADD     DI,4
LEA     SI,TAG
MOV     CX,TAG_LEN
```

Parsing compositions

- **LR** undefined
- **Generalised** ambiguous
- **PEG** shadowing

Syntax Directed Editing



The challenge

Challenge:
SDE's power +
a text editor feel?

Solution: Language Box Editor

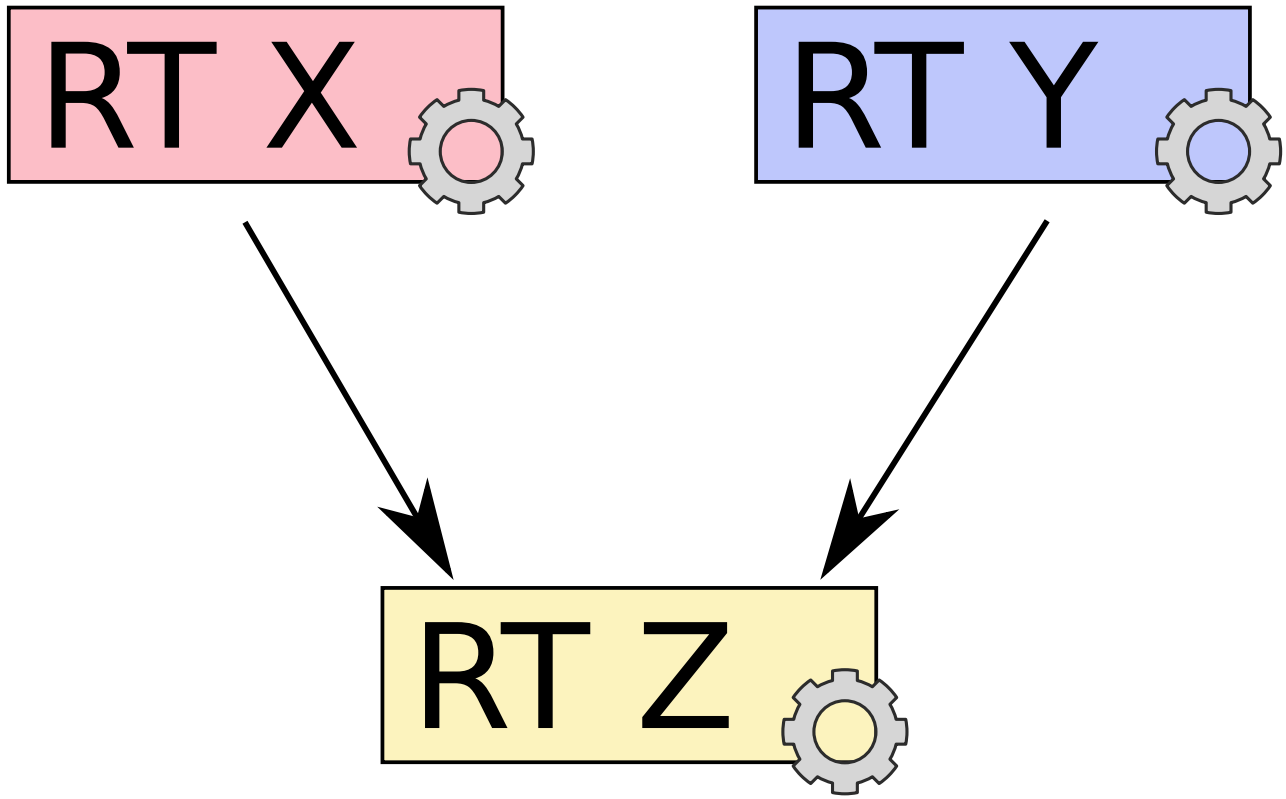
Underlying language composition challenges

Parsing

Running

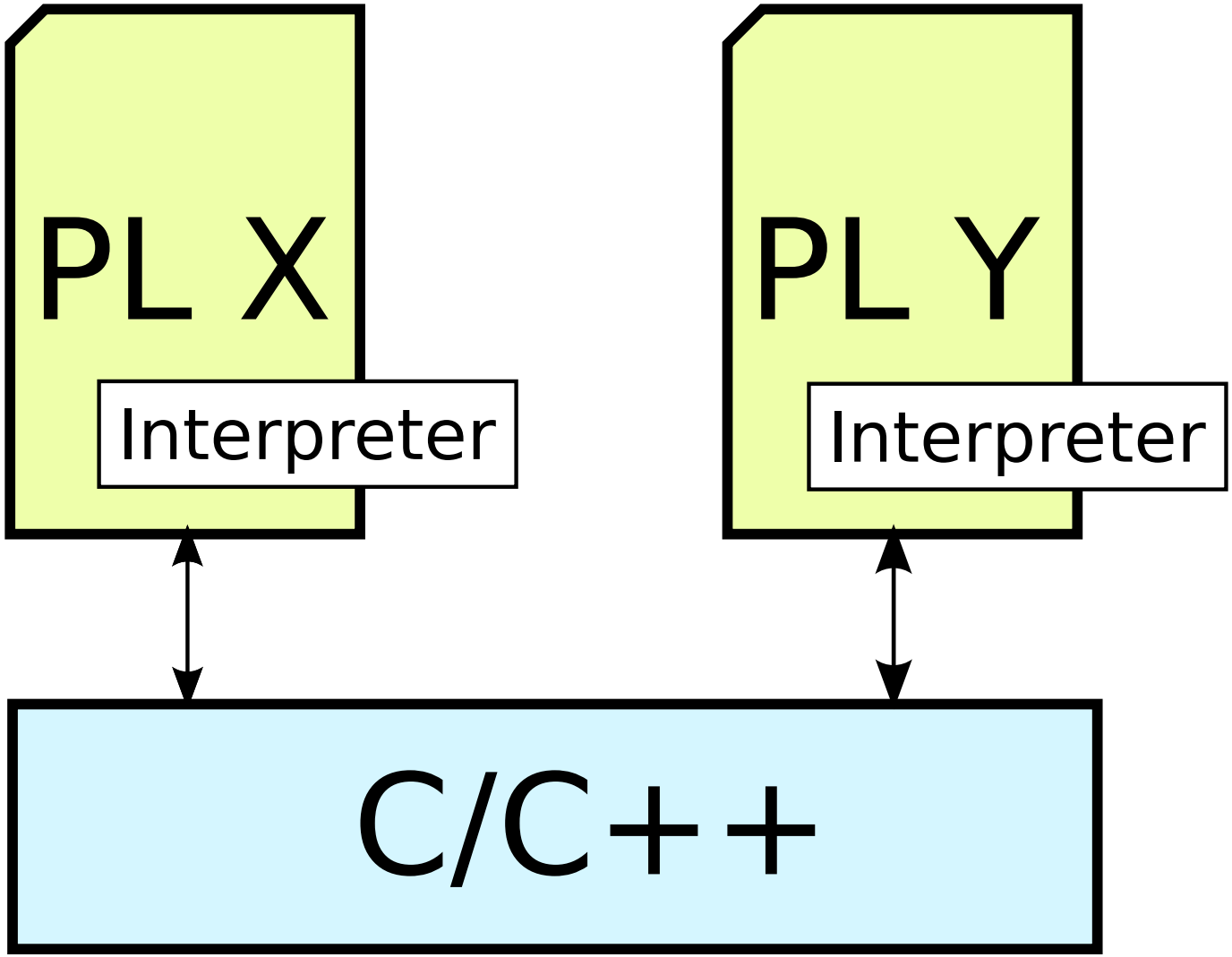
```
SUB     AX,AX
MOV     ES,AX
SUB     BH,BH
MOV     BL,INT_NUMBER
SHL     BX,1
SHL     BX,1
MOV     DI,ES:[BX]
MOV     ES,ES:[BX+2]
ADD     DI,4
LEA     SI,TAG
MOV     CX,TAG_LEN
```

Composing Runtimes

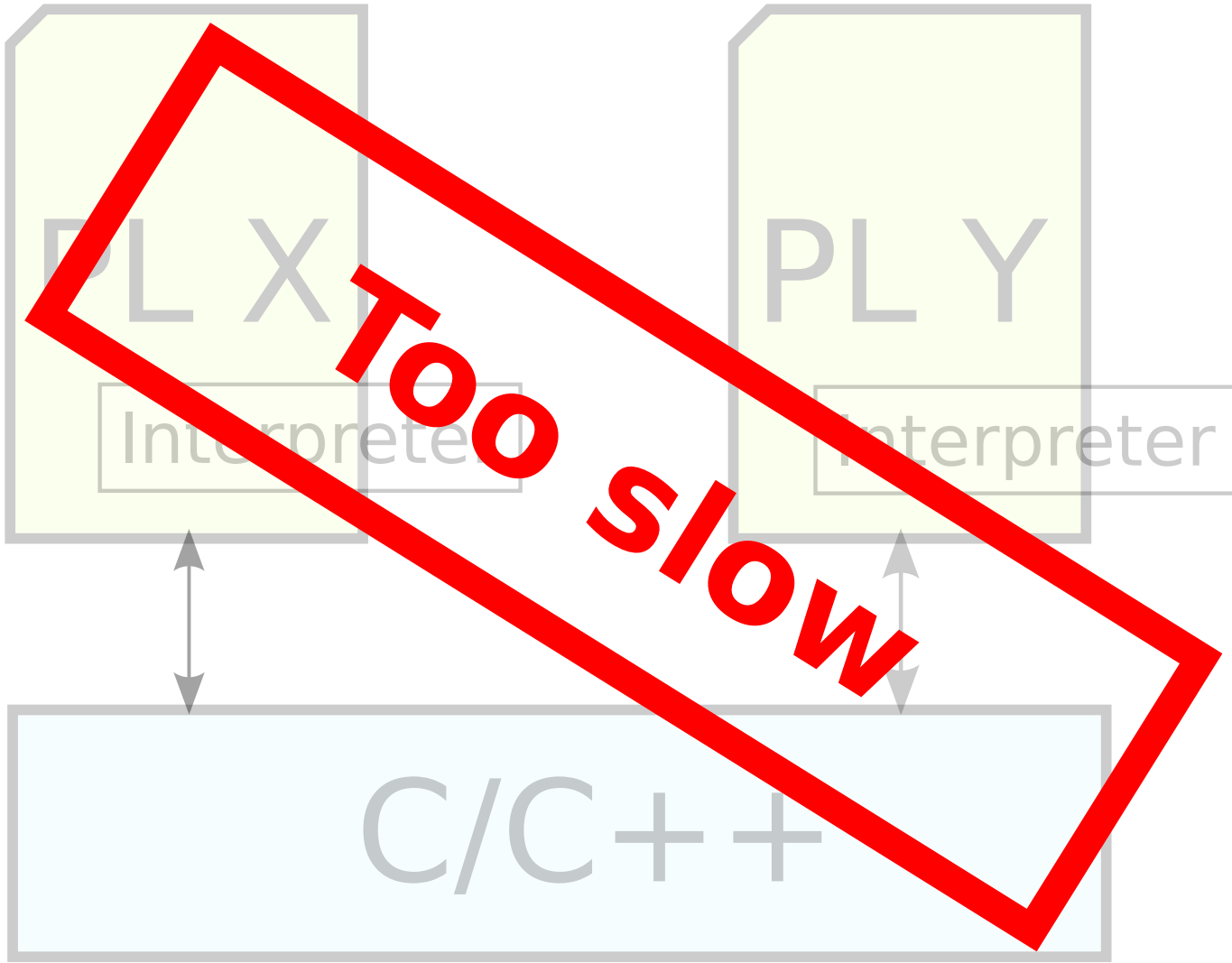


Easy?

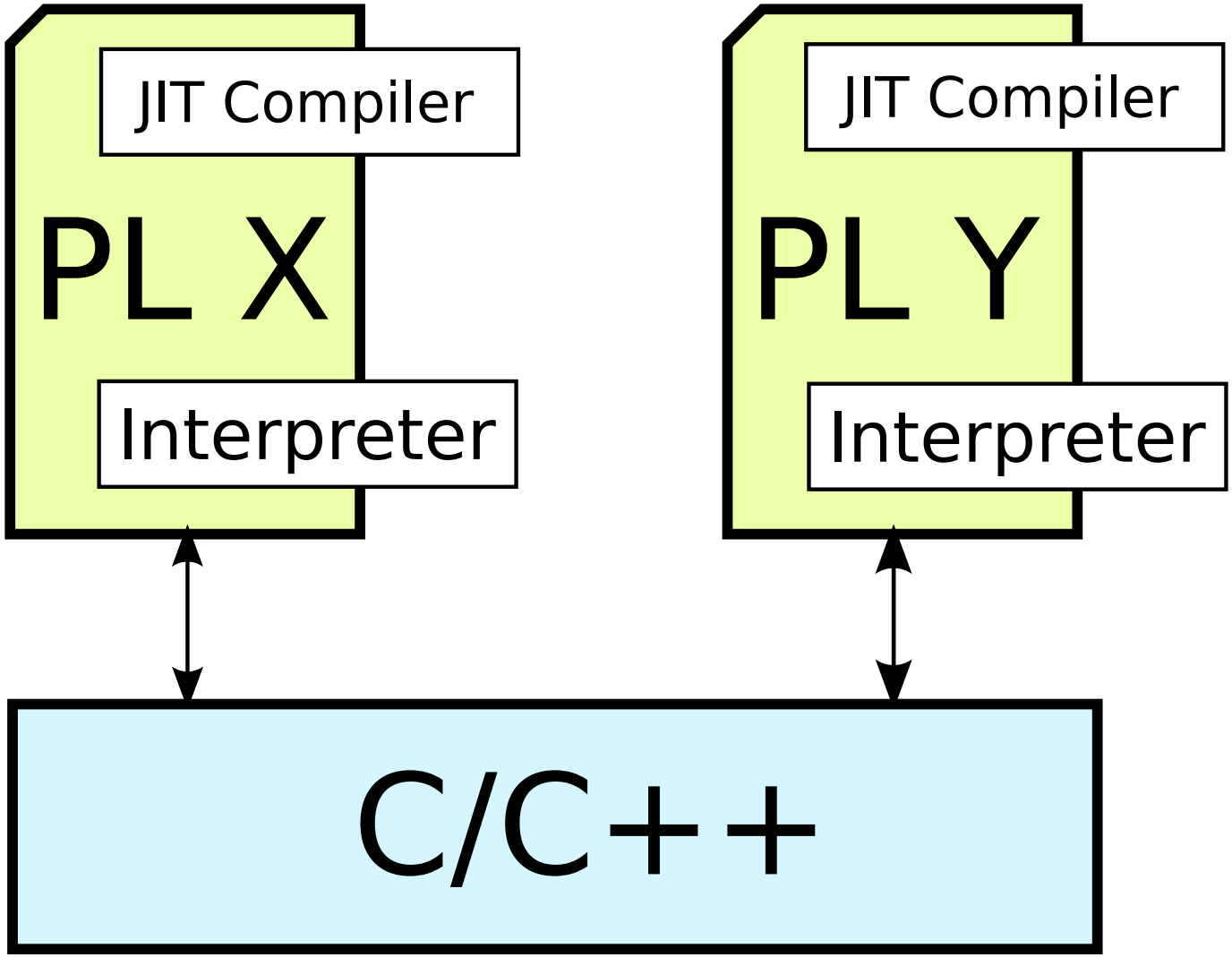
Runtime Composition



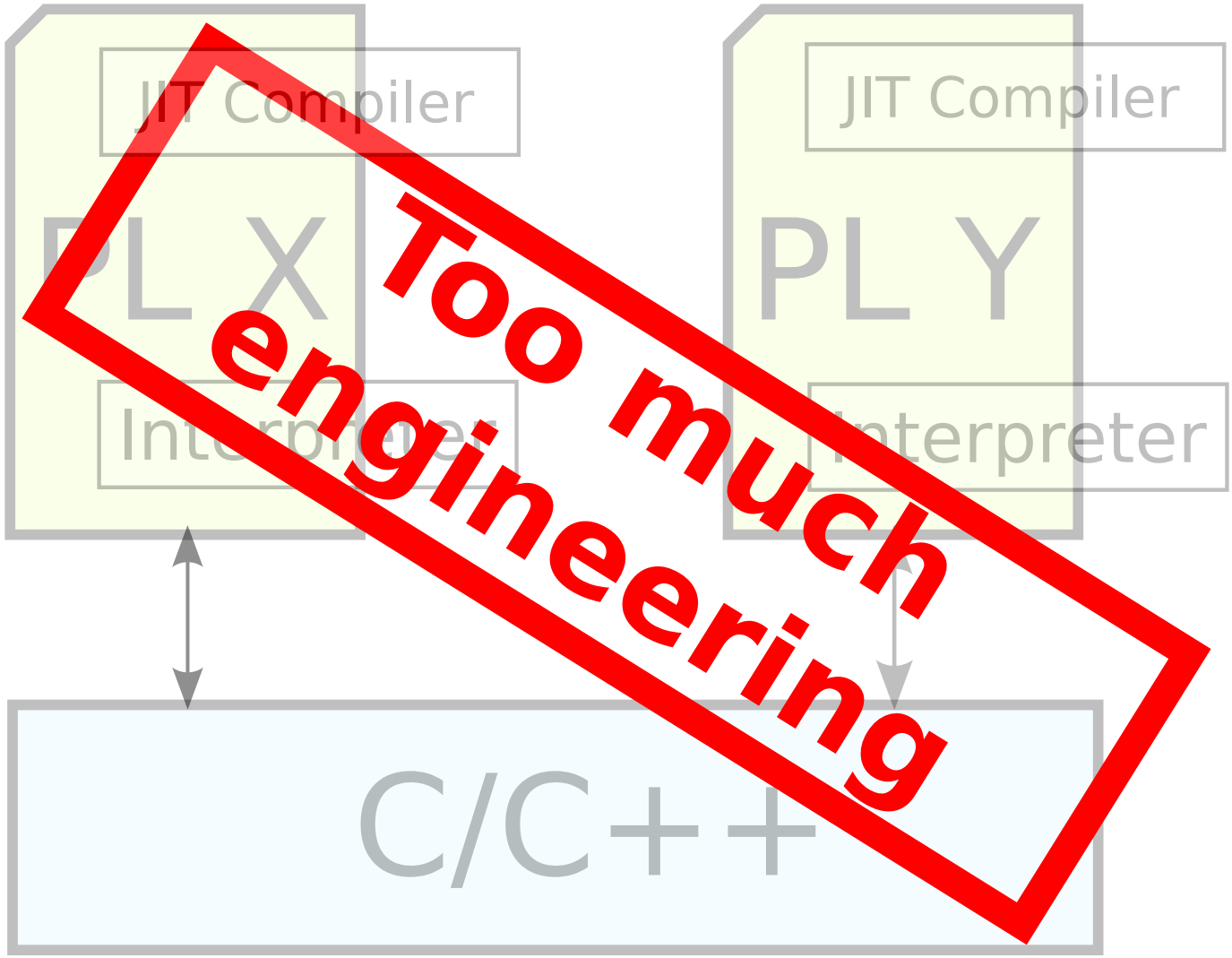
Runtime Composition



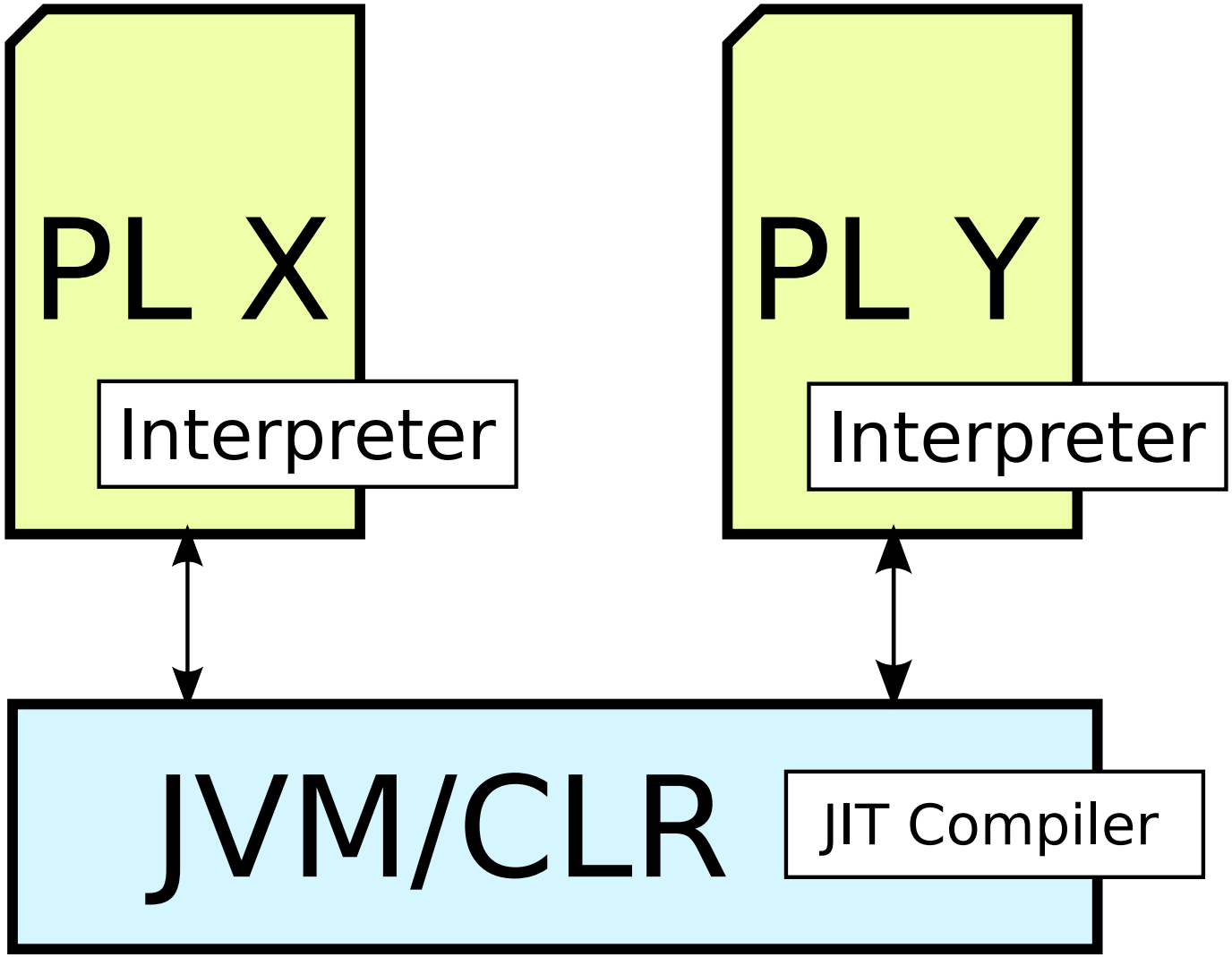
Runtime Composition



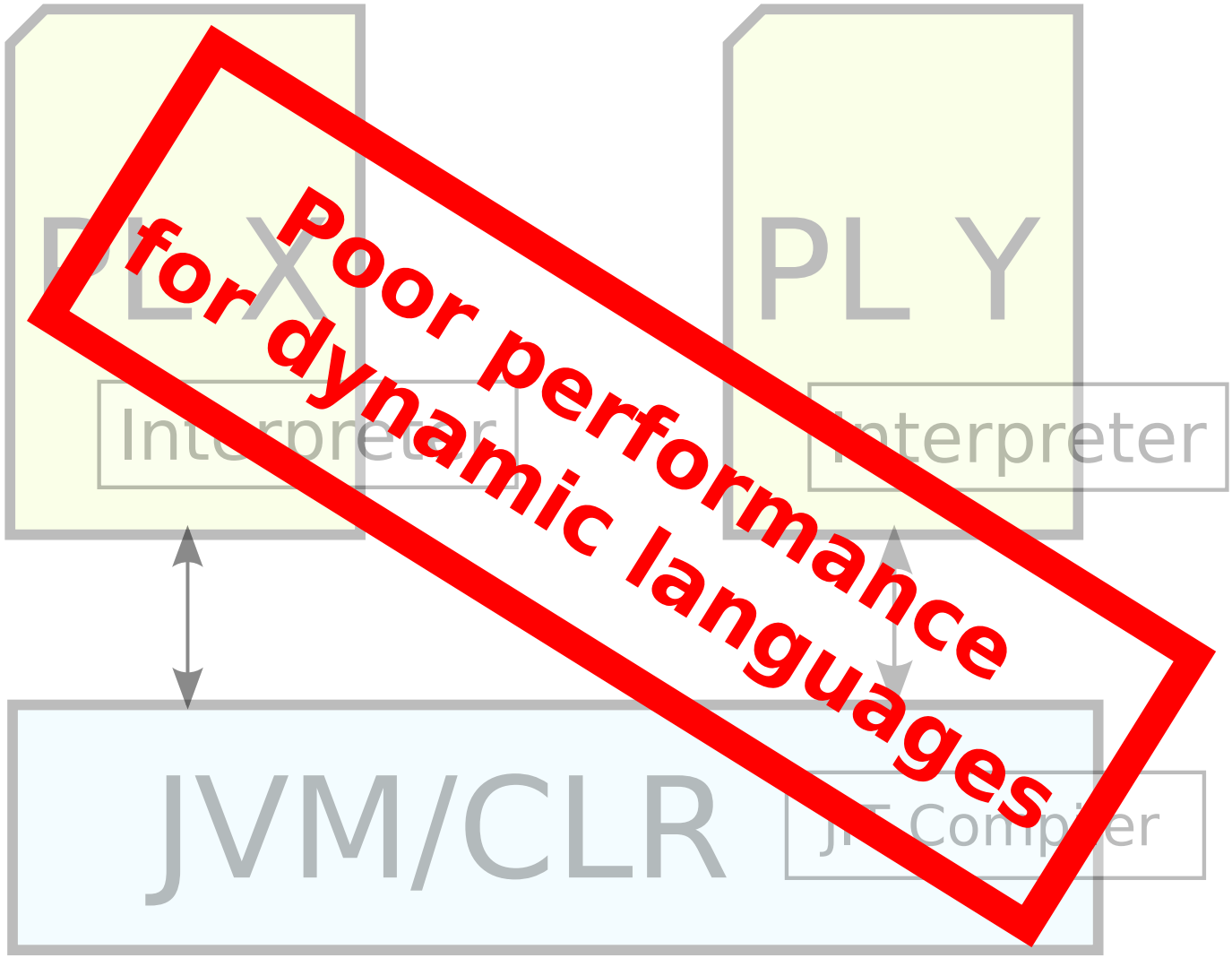
Runtime Composition



Runtime Composition

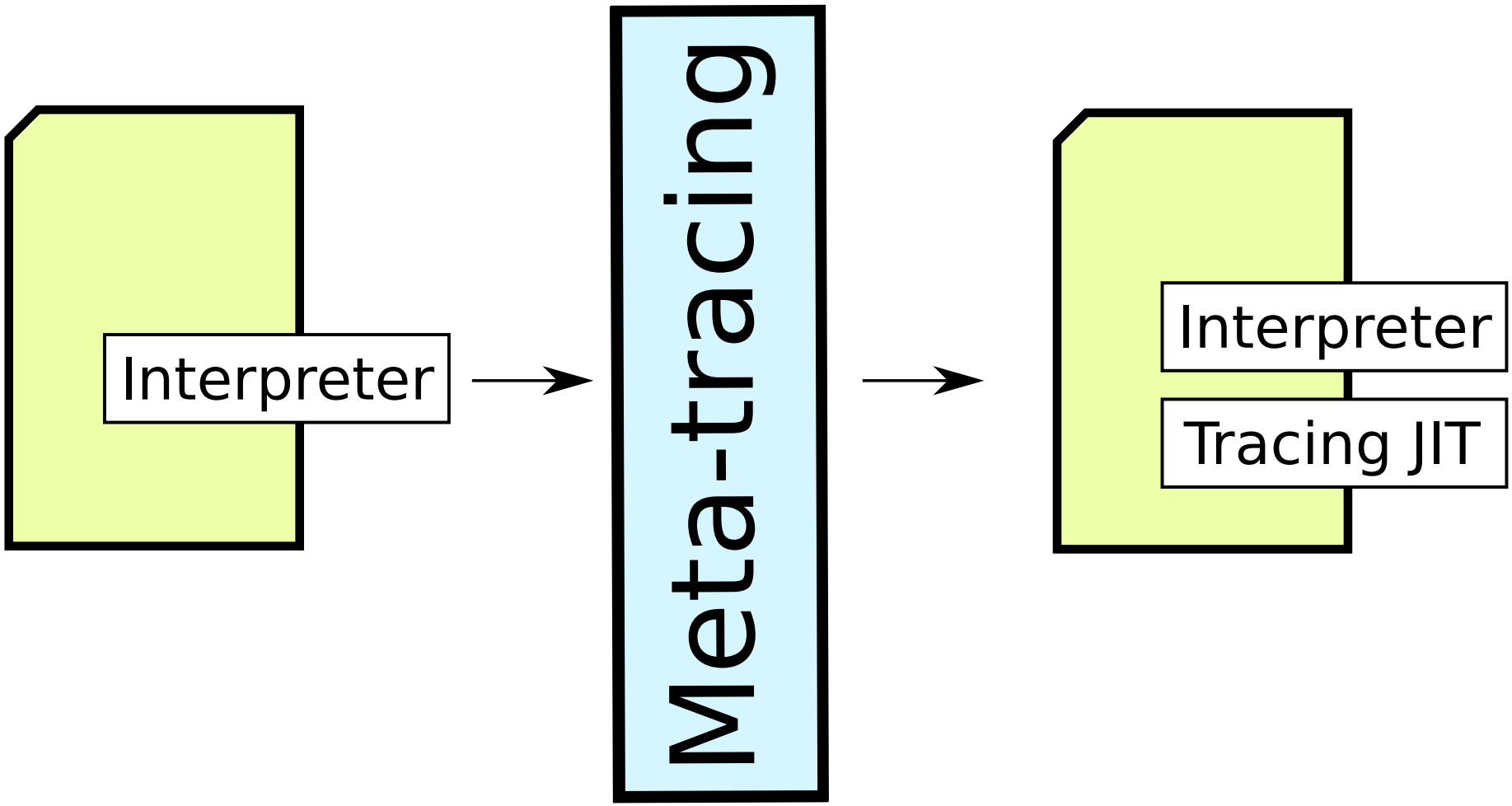


Runtime Composition

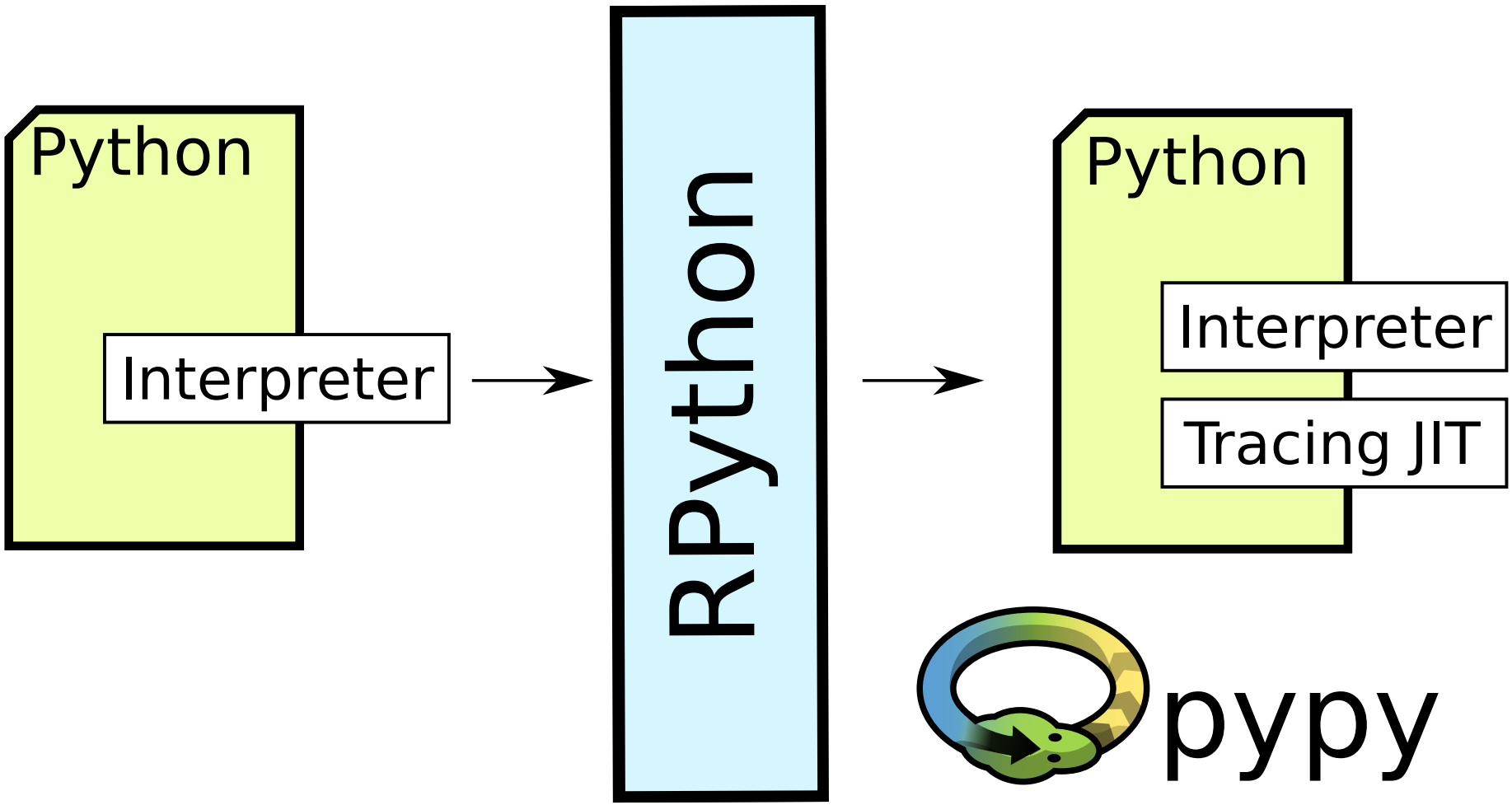


Solution: Meta-tracing

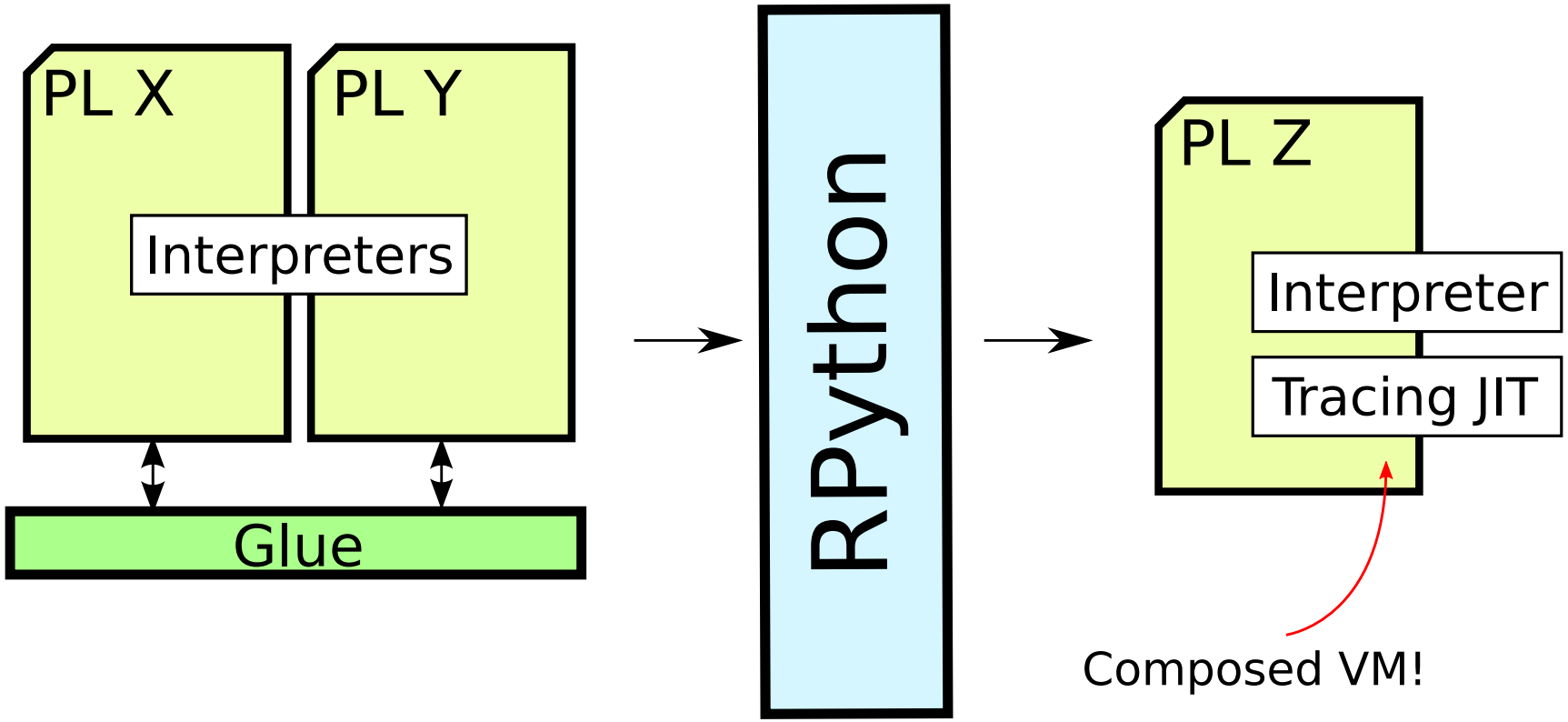
Meta-tracing



Meta-tracing



How Does this Apply to VM Composition?



Little Engineering + High Performance

Composed VMs to Date

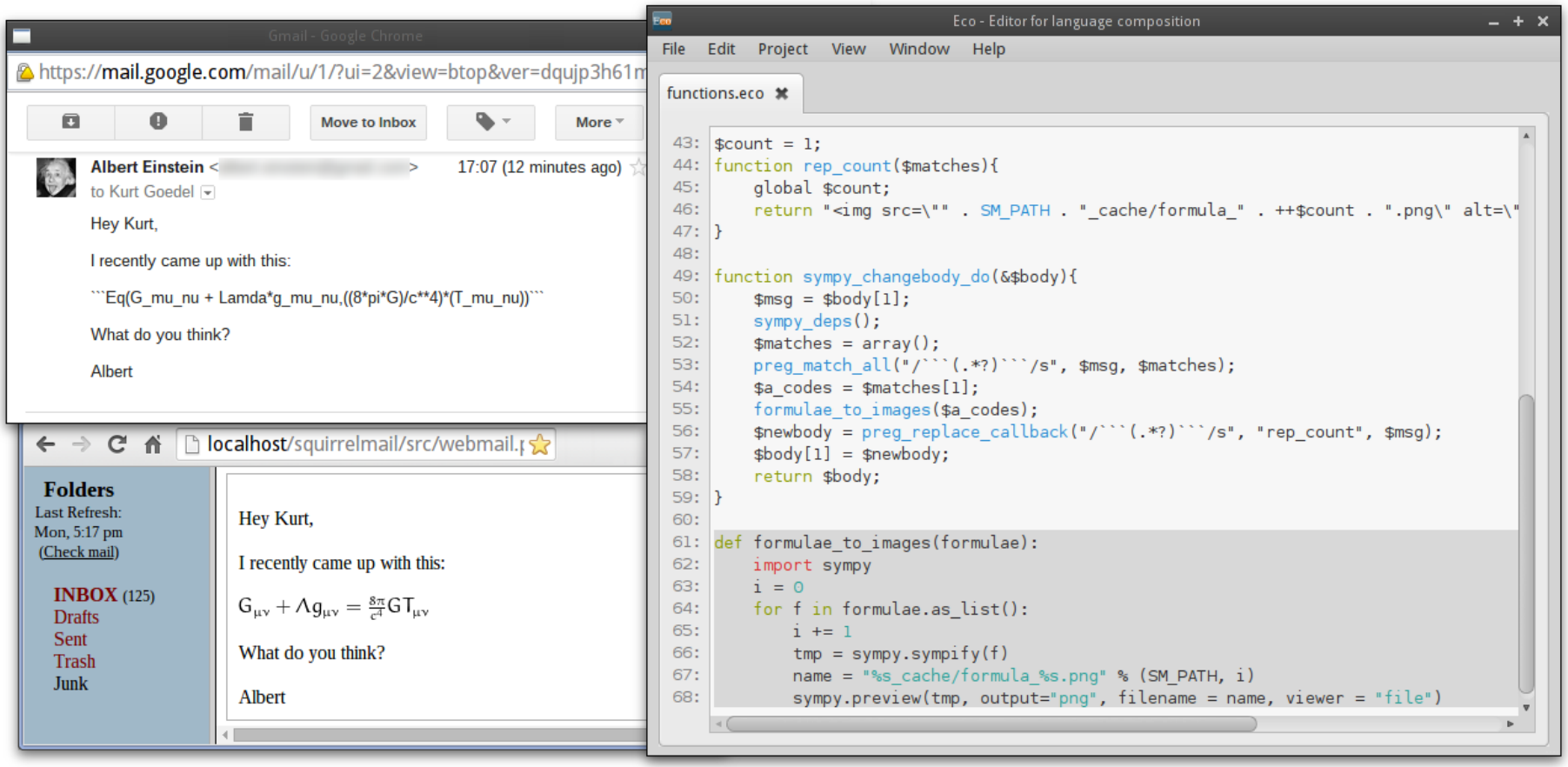
So far we have:

- Unipycation: Python + Prolog
- PyHyp: Python + PHP
- SQPyte: Python + SQLite

Demo

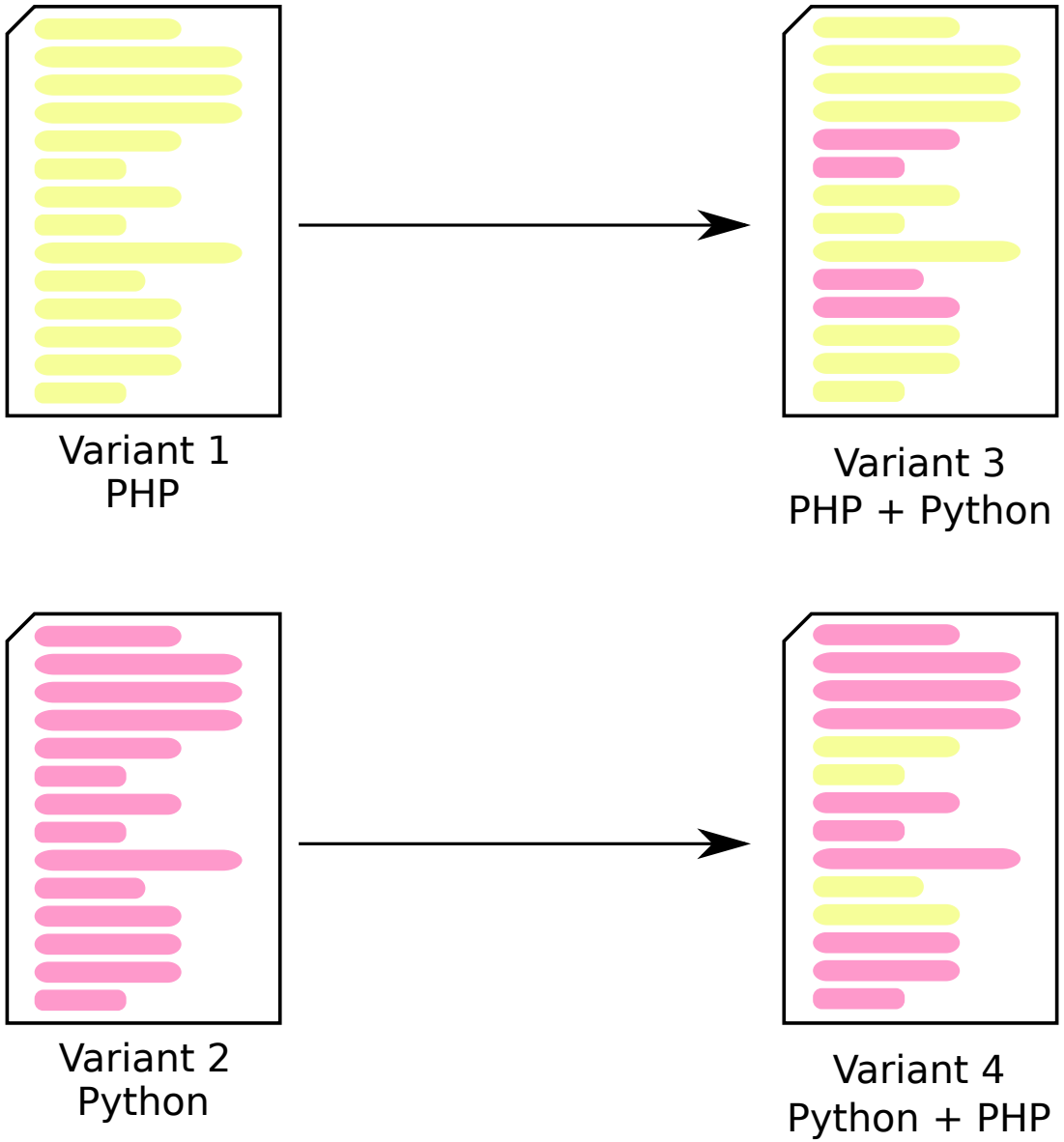
Let’s see one of these VMs in action!

PyHyp = PyPy + HippyVM


Gmail - Google Chrome
https://mail.google.com/mail/u/1/?ui=2&view=btop&ver=dqujp3h61m
Move to Inbox
More
Albert Einstein <
17:07 (12 minutes ago)
to Kurt Goedel
Hey Kurt,
I recently came up with this:
```Eq(G_mu_nu + Lamda*g_mu_nu,((8*pi*G)/c**4)*(T_mu_nu))```
What do you think?
Albert
localhost/squirrelmail/src/webmail.
Folders
Last Refresh:
Mon, 5:17 pm
(Check mail)
INBOX (125)
Drafts
Sent
Trash
Junk
Hey Kurt,
I recently came up with this:
$G_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi}{c^4} G T_{\mu\nu}$
What do you think?
Albert
Eco - Editor for language composition
File Edit Project View Window Help
functions.eco
43: $count = 1;
44: function rep_count($matches){
45: global $count;
46: return "<img src=\"" . SM_PATH . "_cache/formula_" . ++$count . ".png\" alt=\"
47: }
48:
49: function sympy_changebody_do(&$body){
50: $msg = $body[1];
51: sympy_deps();
52: $matches = array();
53: preg_match_all("/```(.*?)```/s", $msg, $matches);
54: $a_codes = $matches[1];
55: formulae_to_images($a_codes);
56: $newbody = preg_replace_callback("/```(.*?)```/s", "rep_count", $msg);
57: $body[1] = $newbody;
58: return $body;
59: }
60:
61: def formulae_to_images(formulae):
62: import sympy
63: i = 0
64: for f in formulae.as_list():
65: i += 1
66: tmp = sympy.sympify(f)
67: name = "%s_cache/formula_%s.png" % (SM_PATH, i)
68: sympy.preview(tmp, output="png", filename = name, viewer = "file")
```

How should we measure performance?

Performance



Performance

- E.g. PyHyp composed benchmarks:
  - Usually 1-2x slower than mono-language versions.
  - In some cases composed benchmarks are faster.

(See our papers for detailed performance analysis)

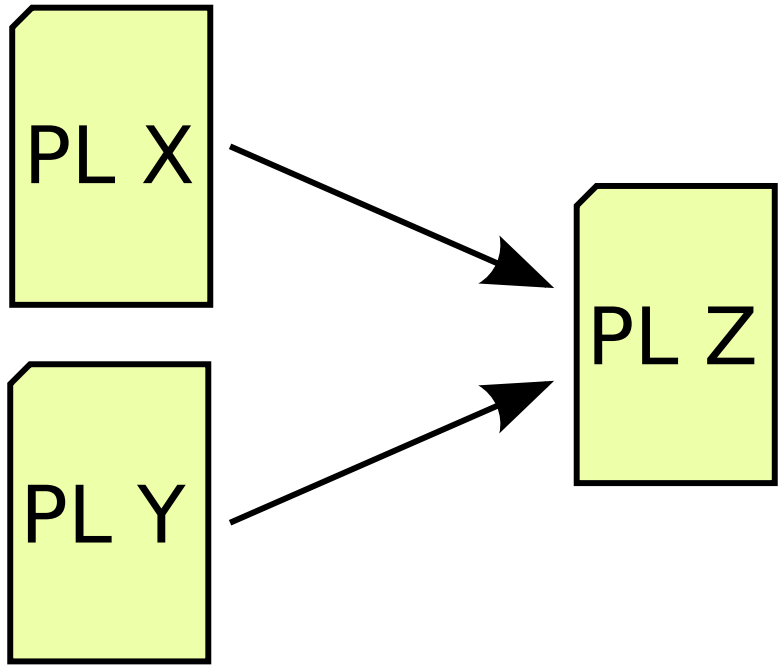
Qualitative Comments and Conclusion

- Editing composed programs with language boxes.
  - Practical way to parse composed programs.
  - Palatable user experience.
- Implementing composed VMs with meta-tracing.
  - Little engineering effort.
  - Good performance.
- Designing x-language interfaces: Hard!
  - Mapping data structures between languages.
  - Mutability differences.
  - Scoping differences.
  - ...

Further Reading

- “Unipycation: A Case Study in Cross-language”, Tracing Edd Barrett, Carl Friedrich Bolz, Laurence Tratt.
- “Approaches to Interpreter Composition”, Edd Barrett, Carl Friedrich Bolz, Laurence Tratt.
- “Fine-grained Language Composition: A Case Study”, Edd Barrett, Carl Friedrich Bolz, Lukas Diekmann, Laurence Tratt.
- “Making an Embedded DBMS JIT-friendly”, Carl Friedrich Bolz, Darya Kurilova, Laurence Tratt.
- “Eco: A Language Composition Editor”, Lukas Diekmann, Laurence Tratt.

Thanks for Listening



Discussion / Questions