

How can parsers be used in industrial projects?

Federico Tomassetti, Language Architect at Strumenta

Hi, I am Federico!

I co-founded Strumenta, a Consulting Studio specialized in Language Engineering.

We design new languages and tooling for existing languages and parsers are present in many of the solutions we build.

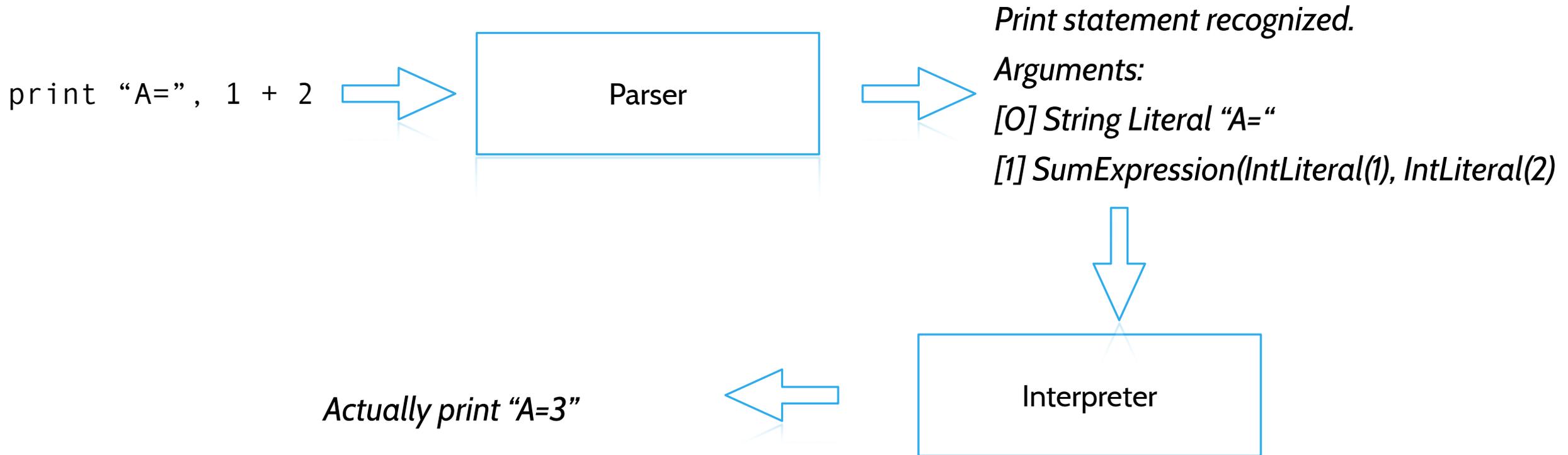
What a parser does?

A parser “understands” code and transform it into data, so that you can use it to do something smart

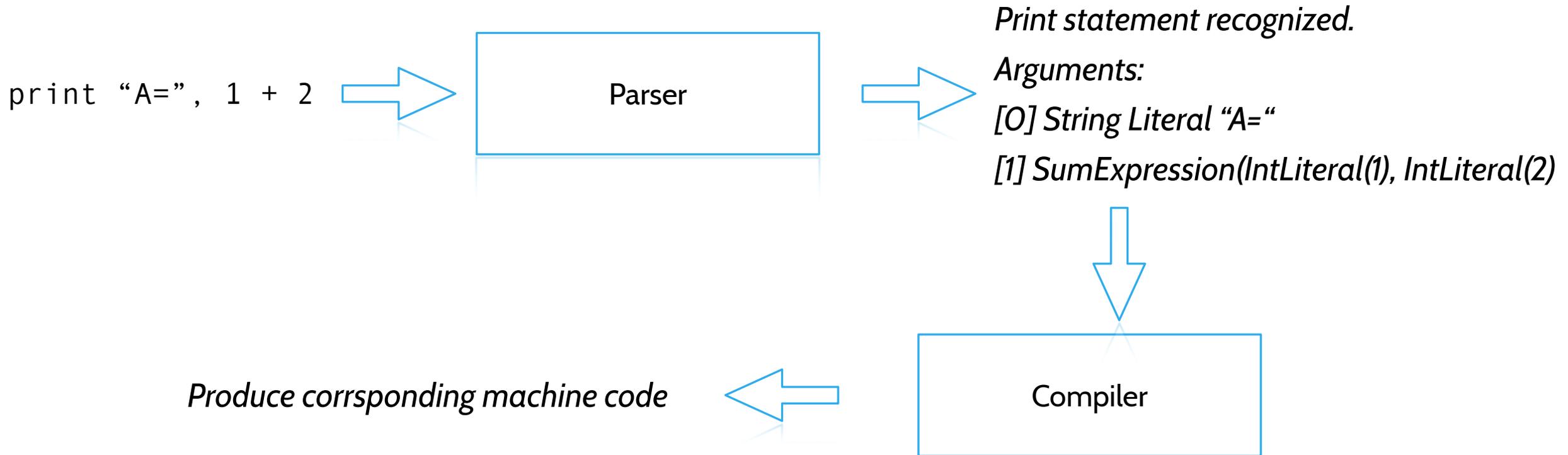
Which applications can use a parser?

- Interpreters
- Compilers
- Transpilers
- Editors
- Code Analysis
- Code Refactoring

Interpreters

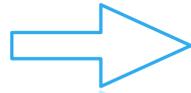


Compilers

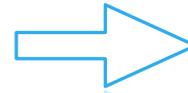


Transpiler

```
print "A=", 1 + 2
```



Parser



Print statement recognized.

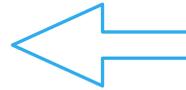
Arguments:

[0] String Literal "A="

[1] SumExpression(IntLiteral(1), IntLiteral(2))



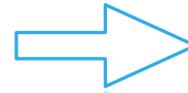
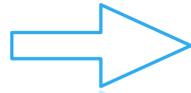
Transpiler



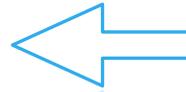
```
System.out.println("A=", 1 + 2);
```

Editor

```
var v = 3  
print "A=", 1 +
```



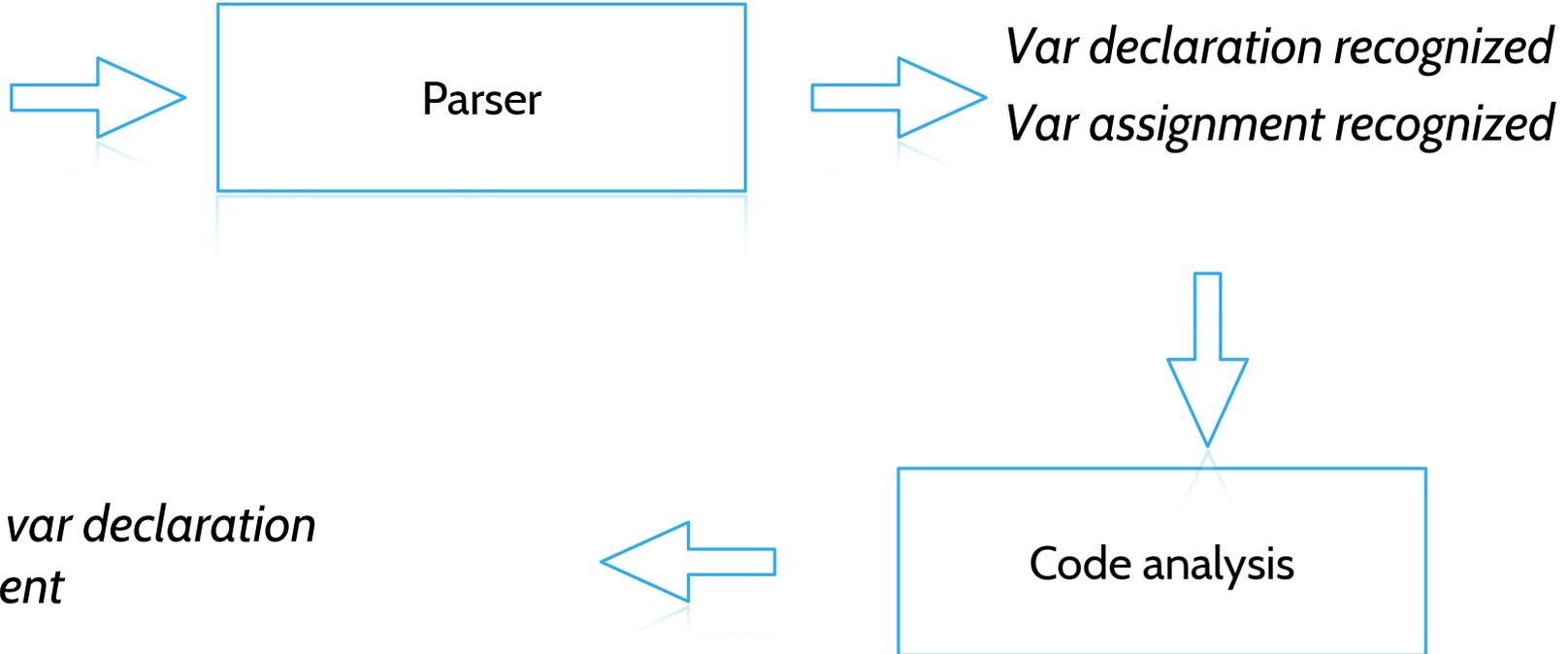
Var declaration recognized
Incomplete print statement recognized



- *Show syntactic error*
- *When autocompletion triggered suggest "v"*

Code analysis

```
var v  
v = 3
```



*- Suggest merging var declaration
and var assignment*

Code refactoring

```
var v  
v = 3
```



Var declaration recognized
Var assignment recognized



- Merge var declaration and var assignment

Do You Need a Parser to Parse?

Technology	Language to Parse	Expertise Needed
Regular expressions	Simple	Common programming
Parser combinators	Simple to Moderate	Little knowledge of parsing
Parser generator tools	Complex	Parsing experts
Custom parser	Complex	Professional parsing developers

Do You Need a Parser to Parse?

Technology	Language to Parse	Expertise Needed
Regular expressions	Simple	Common programming
Parser generator tools	Complex	Parsing experts
Custom parser	Complex	Professional parsing developers

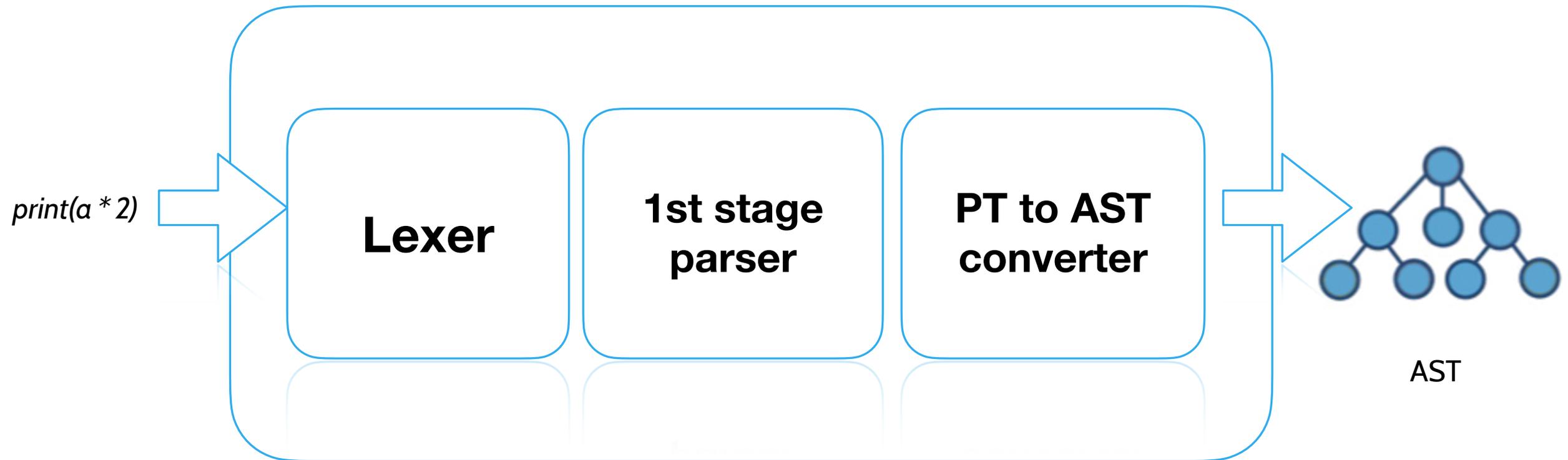
*This is the sweet-spot for most industrial applications:
one gets almost the performance of a custom made parser, but with a fraction of the cost*

The parser generator we use and recommend is ANTLR

How does a parser built with ANTLR look like?



How does a parser built with ANTLR look like?



How does a parser built with ANTLR look like?



How does a parser built with ANTLR look like?

```
OPEN_PAREN : '(';
CLOSE_PAREN : ')';
fragment DECIMAL_SEPARATOR : [.,];
NUMBER : ([0-9]+(DECIMAL_SEPARATOR[0-9]*)?) | DECIMAL_SEPARATOR[0-9]+ ;
SEMI : ';';
COLON : ':';
ID : ('*' {getCharPositionInLine()}>7}? '*'? [a-zA-Z])?
    [$£#@%$a-zA-Z]{getCharPositionInLine()}>7}? [$£#@%$a-zA-Z0-9_]* ;
NEWLINE : (('r'? '\n')|'\r') -> skip;
WS : [ \t] {getCharPositionInLine()}>6}? [ \t]* -> skip ; // skip spaces, tabs
```

A lexer produces a list of tokens.

Each token is composed by:

- a type (e.g., IDENTIFIER or NUMBER)
- its position in the source
- the text

How does a parser built with ANTLR look like?



How does a parser built with ANTLR look like?

```
ifstatement:  
    (beginif  
    thenBody+=statement*  
    elseIfClause*  
    elseClause?  
    endif)  
;  
elseIfClause:  
    (elseifstmt statement*)  
;  
elseClause:  
    (elsestmt statement*)  
;
```

The 1st stage parser organizes the tokens into a tree: the parse tree.

For example it could organizes a list of tokens into the parse tree for an if-stmt, recognizing the condition, the then clause, and the else clause

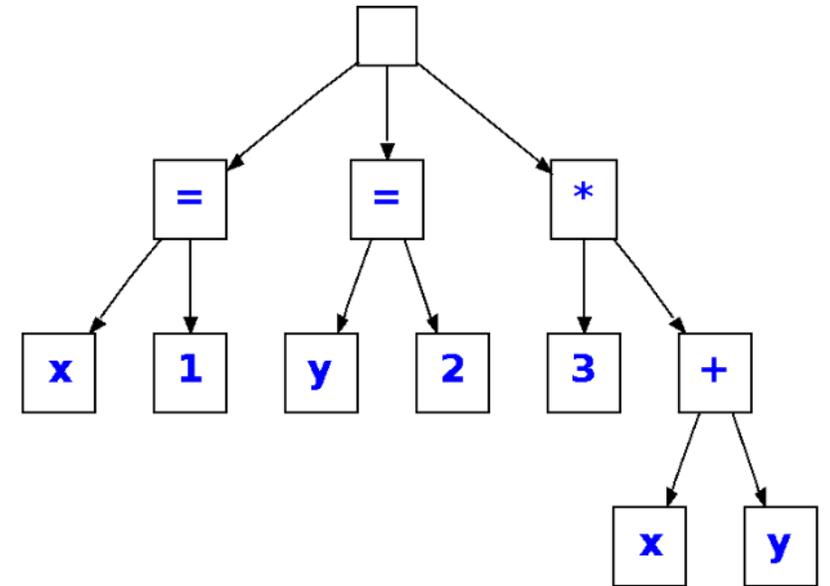
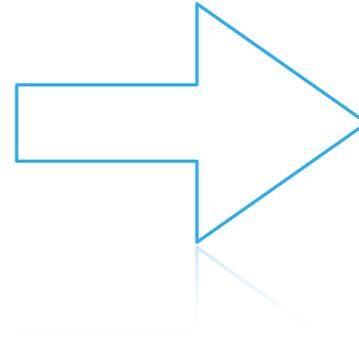
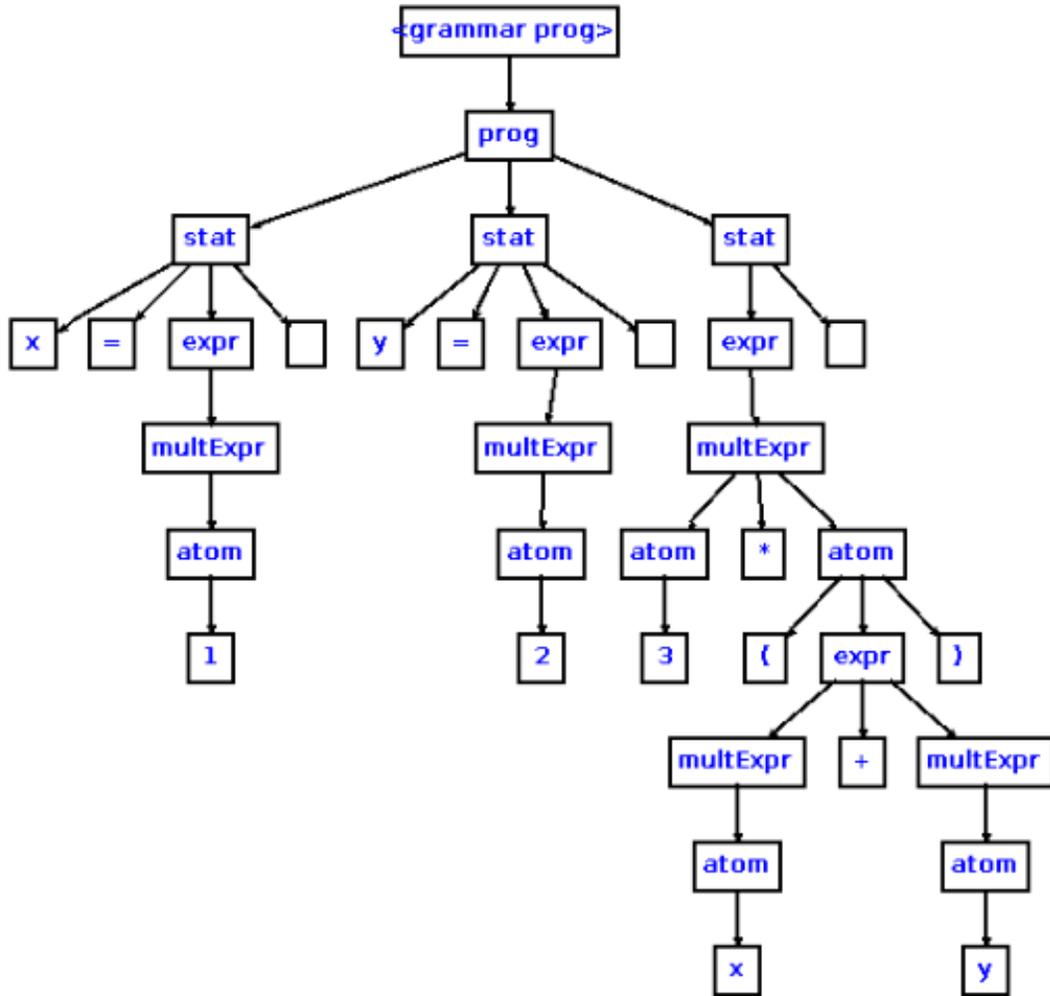
How does a parser built with ANTLR look like?



How does a parser built with ANTLR look like?

```
fun RpgParser.ExpressionContext.toAst(conf: ToAstConfiguration = ToAstConfiguration()): Expression {
    return when {
        this.number() != null -> this.number()!!.toAst(conf)
        this.identifier() != null -> this.identifier().toAst(conf)
        this.comparisonOperator() != null -> when {
            this.comparisonOperator().GT() != null ->
                GreaterThanExpr(this.expression(i: 0).toAst(conf), this.expression(i: 1).toAst(conf))
            this.comparisonOperator().GE() != null ->
                GreaterEqualThanExpr(this.expression(i: 0).toAst(conf), this.expression(i: 1).toAst(conf))
            this.comparisonOperator().LT() != null ->
                LessThanExpr(this.expression(i: 0).toAst(conf), this.expression(i: 1).toAst(conf))
            this.comparisonOperator().LE() != null ->
                LessEqualThanExpr(this.expression(i: 0).toAst(conf), this.expression(i: 1).toAst(conf))
            this.comparisonOperator().NE() != null ->
                DifferentThanExpr(this.expression(i: 0).toAst(conf), this.expression(i: 1).toAst(conf))
            else -> todo(conf = conf)
        }
    }
}
```

Parse tree vs AST



Parse tree vs AST

Parse Tree

The parse tree is a concrete representation of the input. The parse tree retains all of the information of the input.

This is how a parser would think about the code

AST

The AST is an abstract representation of the input. For example, parenthesis are thrown away, as they were useful to determine precedence

This is how a user would think about the code

How do we interface with a parser?

You want to process the AST.

You typically want operations to traverse the tree:

- finding ancestor of a given type (e.g., the method containing a certain expression)
- find descendants of a given type (e.g., all variable references)
- add links (for symbol resolution)

We have created our own Open-Source library to express ASTs, called Kolasu

What we cover in the rest of the presentation

1. Focus on one problem: legacy modernization
2. Focus on two applications solving this problem and based on parsers: interpreters and transpilers

Legacy Modernization



Is the platform going to be available in the future?

Has the platform good enough performance?

Can you find developers who know that languages?

Are more modern language much more productive?

Legacy Modernization



Can't I just rewrite the system?

They did it by making the single worst strategic mistake that any software company can make:

They decided to rewrite the code from scratch.

<https://www.joelonsoftware.com/2000/04/06/things-you-should-never-do-part-i/>

Legacy Modernization



Scale of the problem

Reuters calculates that there's **still** 220 billion lines of **COBOL code** currently being used in production today, and that every day, **COBOL** systems handle \$3 trillion in commerce.

One may also ask, is Cobol still used in 2019? **COBOL still** runs the world in traditional banking, lots of large scale government systems, insurance and health care

<https://askinglot.com/how-many-lines-of-cobol-code-are-there>

Legacy Modernization: alternatives



1. Alternative one: I translate the code from the old language (e.g., COBOL) to the new language (e.g., Java)
2. Alternative two: I make an interpreter which can read the old code but execute in a new platform

Alternative one is good if from now on you want to code in the new language and on a new platform

Alternative two is good if you want to keep using the old language but you need to run it on a new platform

Transpilers: why we need them

1. Legacy Modernization: my code is expressed in a language which is not well suited for today's needs
2. Compatibility: for example, browsers can run only Javascript
3. I want to write in a simple language, but I need a fast language for execution

Transpilers: why we need them

Browsers support one language nowadays: Javascript

So, unless you give them Javascript they cannot handle it.

What if you would be more productive in a statically typed language?

You can use TypeScript

Or you can use ClojureScript, or Elm, or PureScript, or CoffeeScript

Transpilers: why we need them

```
22
23 main =
24   Browser.element
25     { init = init
26       , view = view
27       , update = update
28       , subscriptions = subscriptions
29     }
30
31
32
33 -- MODEL
34
35
36 type alias Model =
37   { zone : Time.Zone
38     , time : Time.Posix
39   }
40
41
```

Transpiler example: VBA to C++ Transpiler

We build a VBA to C++ transpiler for an actuarial and consulting company. A couple of options were evaluated: a VBA to VB.NET transpiler or a VBA to C++ transpiler.

The first option would have made possible to integrate with other .NET languages, the second one would have made easier to get a good level of performance. The company opted for guaranteed performance.

Transpiler example: VBA to C++ Transpiler

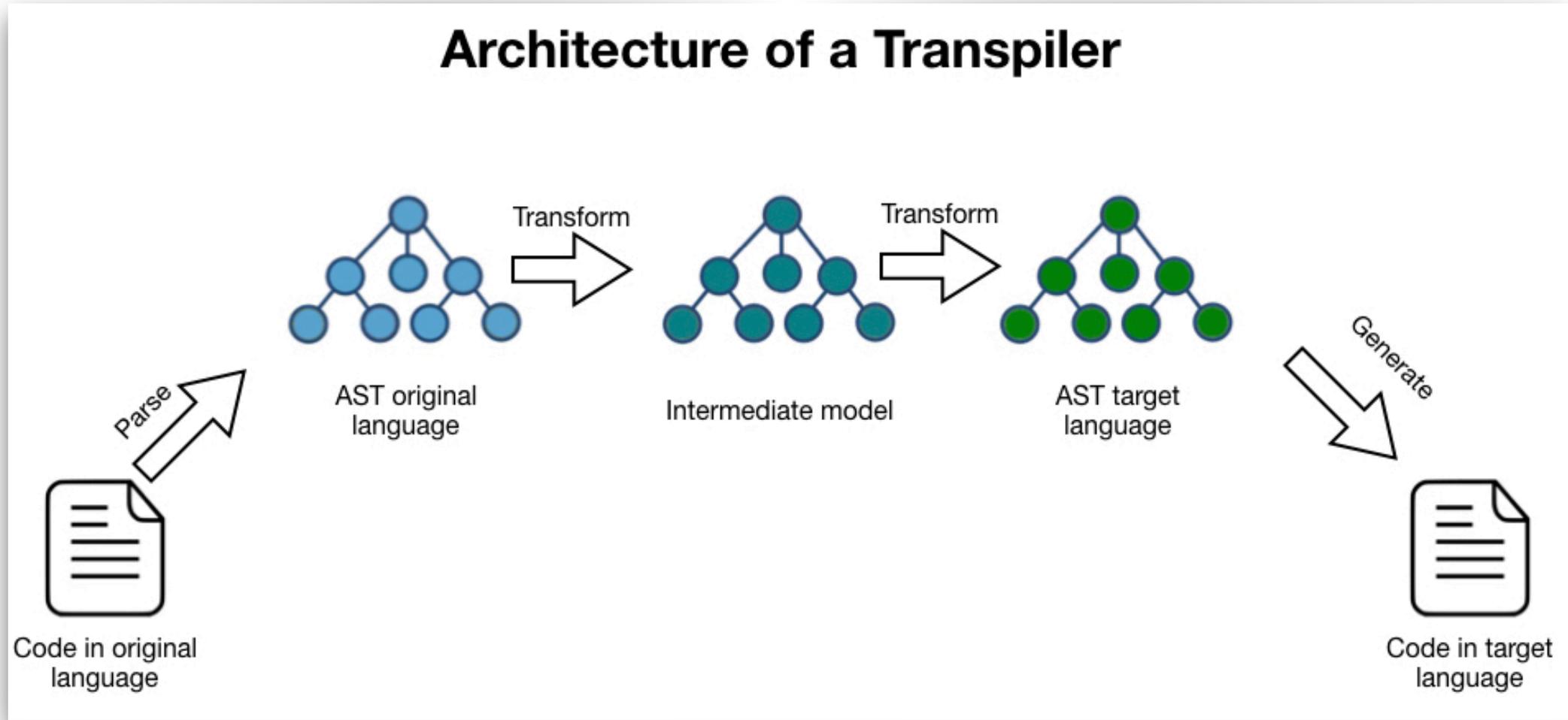
Why building something that transpile VBA?

- VBA is a simple language that can be used by actuaries. They start using it on Excel. The client used a custom platform that supports VBA to make easier for actuaries to transition to their platform. Level of development skills varies: some are good, some are not
- VBA can run on Excel (interpreted) and on the custom platform of the client (compiled/transpiled)

Transpiler example: VBA to C++ Transpiler

- VBA is no more developed. Existing compilers are closed source and with bad performance
- Company needs to perform simple calculations, but on giant databases (millions of data points) each day
- The transpiler can optimize code written by user (e.g. memoization, parallelization) and then transpile it in C++

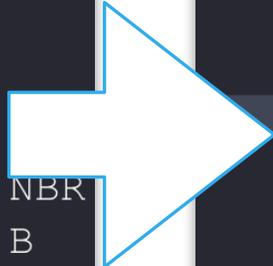
How does a transpiler work?



Transpiler example: RPG to Java

```
FIB
  BEGSR
  SELECT
  WHEN      NBR = 0
  EVAL     RESULT = 0
  WHEN      NBR = 1
  EVAL     RESULT = 1
  OTHER
  FOR      COUNT = 2 TO NBR
  EVAL     RESULT = A + B
  EVAL     A = B
  EVAL     B = RESULT
  ENDFOR
  ENDSL
  ENDSR
```

```
void FIB() {
    if (this.NBR == 1) {
        this.RESULT = 1;
    } else {
        for (this.COUNT = 2; this.COUNT <= this.NBR; this.COUNT++) {
            this.RESULT = this.A + this.B;
            this.A = this.B;
            this.B = this.RESULT;
        }
    }
}
```



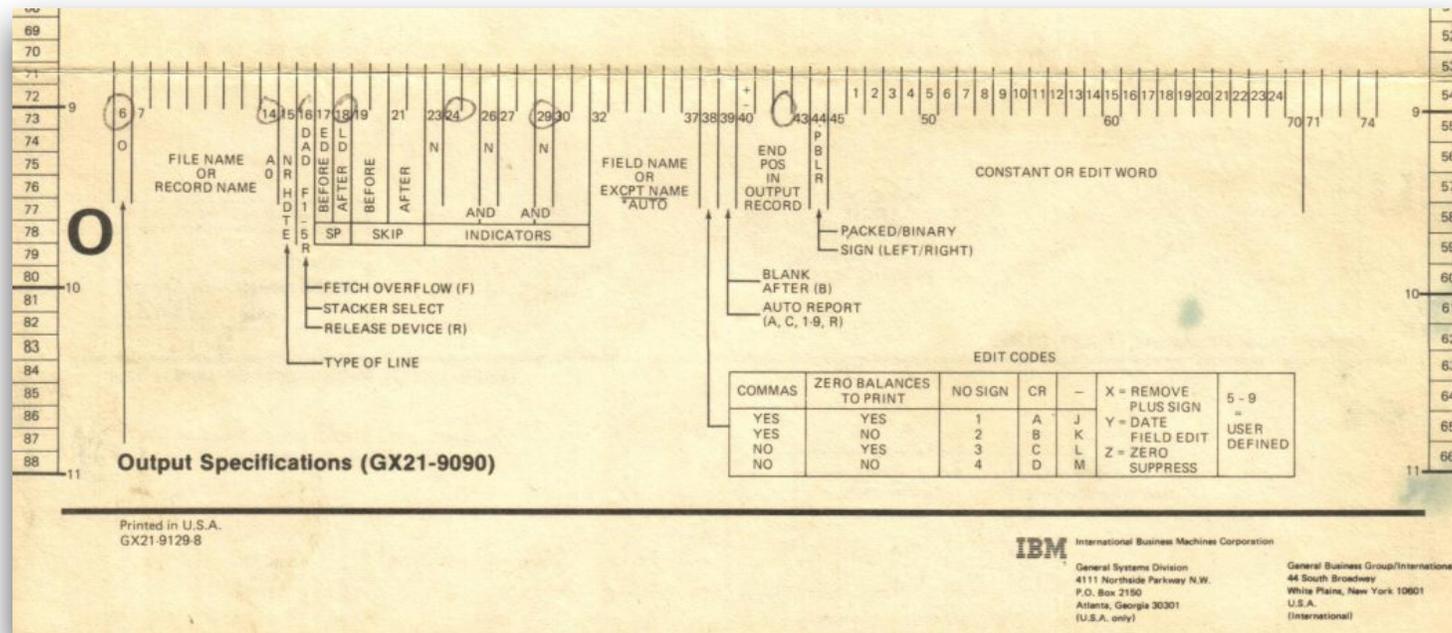
RPG?

RPG is a high level programming language for business applications.

It has been created by IBM and it is available only the IBM i systems (AS400).

It was created in 1959 and it is still used.

Until version IV, it was a fixed-format language.



RPG?

```
D CustomerInfo      DS          QUALIFIED BASED(@)
D   Name           20A
D   Address        50A

D ProductInfo      DS          QUALIFIED BASED(@)
D   Number         5A
D   Description    20A
D   Cost           9P 2

D SalesTransaction...
D                   DS          QUALIFIED
D   Buyer          LIKEDS(CustomerInfo)
D   Seller         LIKEDS(CustomerInfo)
D   NumProducts   10I 0
D   Products      LIKEDS(ProductInfo)
D                   DIM(10)
```

RPG?

```
C      FEXE      BEGSR
C              EVAL      U$IN35=*BLANKS
C      * Search element with Url
C              EVAL      $X=%LOOKUP('Url':$$SVARCD)
C      * Not found is an Error
5      C              IF      $X>0
C              EVAL      U$IN35='1'
5      C              ELSE
C              EVAL      $$URL=$$SVARVA($R)
C      * Replace all variables of execution in url
C              EXSR      REPVAR
C      * Replace all variables of initialisation in url
C              EVAL      $$SVAR=U$SVARSK_INI
C              EXSR      REPVAR
C      * Invoke url
C              CLEAR      $$SVAR
C              EVAL      $$SVARCD(01)='Url'
C              EVAL      $$SVARVA(01)=$$URL
C              CALL      'JD_URL'
C              PARM      $$FUNZ
C              PARM      $$METO
C              PARM      $$SVAR
C              ENDIF
C      *
C              ENDSR
```

Transpiler example: RPG to Java

```
* Calculates number of Fibonacci in an iterative way
D ppdat      S           8
D NBR        S           8  0
D RESULT     S           8  0  INZ (0)
D COUNT      S           8  0
D A          S           8  0  INZ (0)
D B          S           8  0  INZ (1)
```

Transpiler example: RPG to Java

```
public class CALCFIB {  
  
    private java.lang.String ppdat;  
  
    private long NBR;  
  
    private long RESULT = 0;  
  
    private long COUNT;  
  
    private long A = 0;  
  
    private long B = 1;  
  
    private java.lang.String dsp;
```

Transpiler example: RPG to Java

```
FIB          BEGSR
            SELECT
            WHEN          NBR = 0
            EVAL          RESULT = 0
            WHEN          NBR = 1
            EVAL          RESULT = 1
            OTHER
            FOR           COUNT = 2 TO NBR
            EVAL          RESULT = A + B
            EVAL          A = B
            EVAL          B = RESULT
            ENDFOR
            ENDSL
            ENDSR
```

Transpiler example: RPG to Java

```
void FIB() {  
    if (this.NBR == 1) {  
        this.RESULT = 1;  
    } else {  
        for (this.COUNT = 2; this.COUNT <= this.NBR; this.COUNT++) {  
            this.RESULT = this.A + this.B;  
            this.A = this.B;  
            this.B = this.RESULT;  
        }  
    }  
}
```

Interpreters: why we need them

1. They are typically the easiest way to execute code: if performance are not critical they are great
2. They give you more flexibility: for example *eval* in JavaScript

Interpreter example: JaRIKo

G R U P P O



JaRIKo: a **J**ava virtual machine **R**pg Interpreter written in **K**otlin.

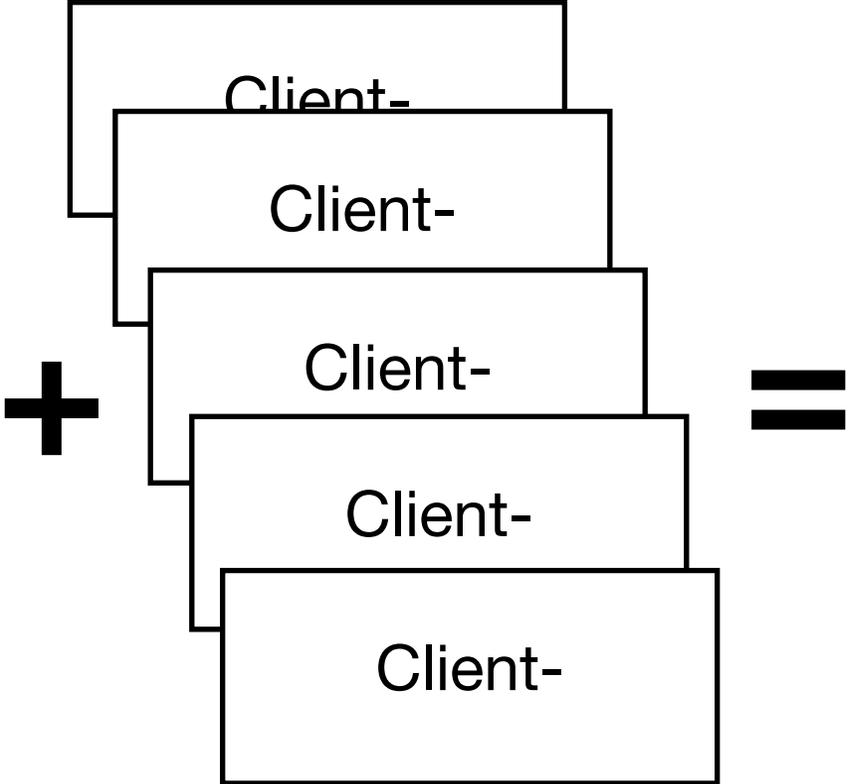
JaRIKo is an interpreter for the RPG programming language. It runs on the JVM since it's written in Kotlin.

We built the first version.

<https://github.com/smeup/jariko>

<https://tomasseti.me/jariko-an-rpg-interpreter-in-kotlin/>

Interpreter example: JaRIKo



**A very valuable system,
written in RPG**

Interpreter example: JaRIKo

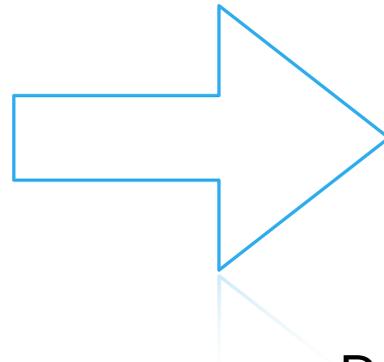
Without JaRIKO



Developers use
RPG



Consultants use
RPG



With JaRIKO



Developers use Java/Kotlin,
incrementally



Consultants use
RPG

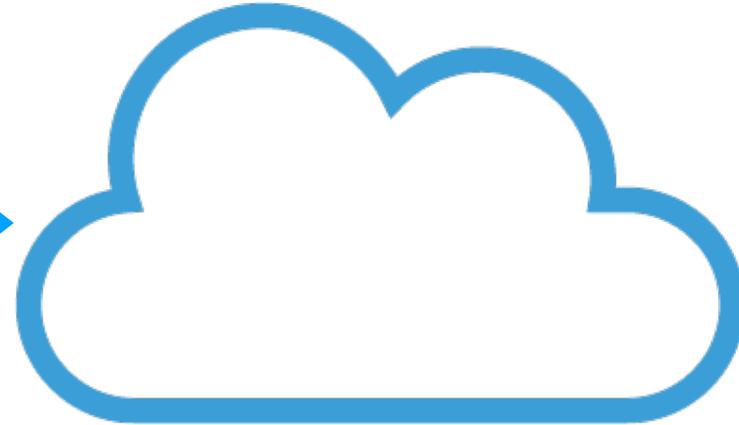
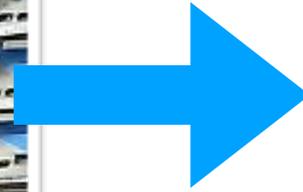
Why?



Currently RPG runs only on the IBM-i (new name of the AS/400 platform), so we are locked into that platform.

Is the IBM-i going to be available in the future?

Why?



Writing new code in RPG
and Java + Kotlin

Interpreter example: JaRIKo

Rpgle sources samples Hello world ↕

```
D Msg          S          50  
C              Eval      Msg = 'Hello World !'  
C              dsply     Msg  
C              SETON  
C                                     LR
```

Paste an rpgle source

Parameters (pipe separated)

Actions RUN

Execution time

Output

Interpreter example: JaRIKo

Why building something that runs RPG, such an old and niche programming language?

- RPG is widely common in the business and financial industry.
- There are tons of super-stable, highly-tested, mission-critical lines of RPG code running a big amount of businesses since tens of years.
- Big companies still rely on this programming language.

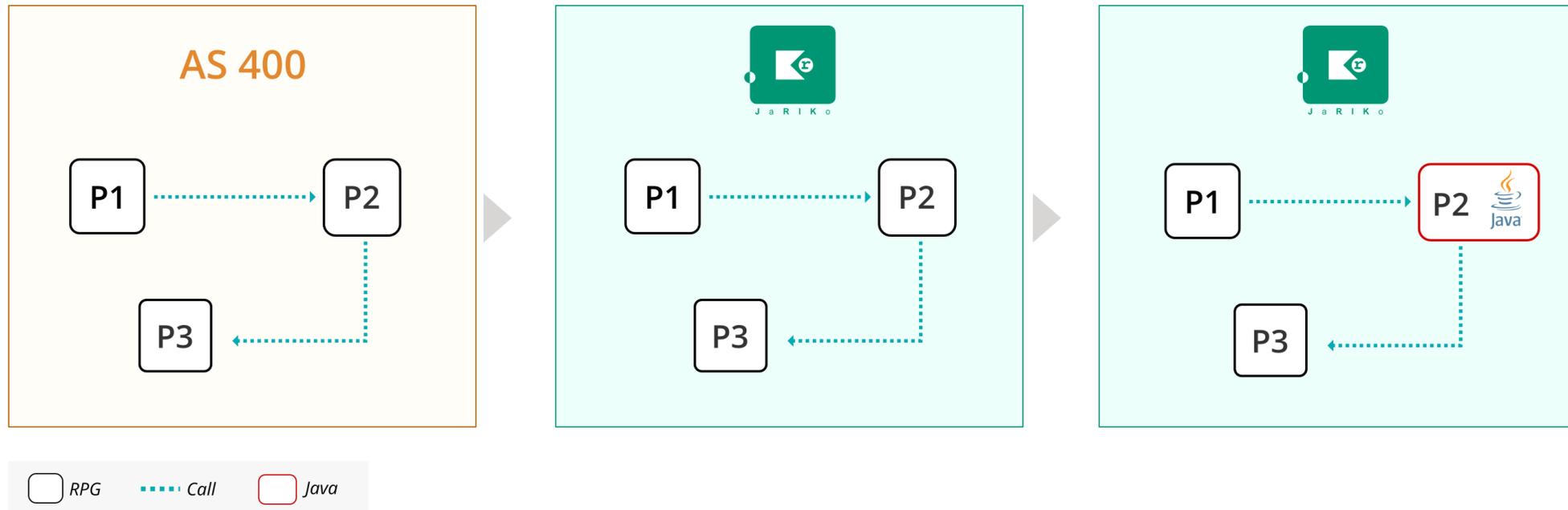
Interpreter example: JaRIKo

- First, RPG was designed for business, it is very simple, powerful for data manipulation, but extremely inadequate to solving technology issue (like threads, async, http calls, cryptography, and so forth). The developer doesn't need to be aware of the technical details. The system provides all this technology, making code run on top of a platform that solves those issues. This helped to create a generation of RPG programmers that are closer to business consultants than to developers, and this is very good for business applications development
- The second is that RPG only runs on IBMi, best known as AS/400, that was also designed for business, is very reliable, fast, well-supported, and stable. RPG leverages AS/400 architecture, they are the perfect couple

Interpreter example: JaRIKo

One of the core features of Jariko is the “doping” mechanism: once the code is taken over by the interpreter, every single program can be replaced at runtime. This allows to write a very flexible and polyglot software, choosing the right tool for the right job and having all the java (and JVM) power and ecosystem available.

Interpreter example: JaRIKo



Interpreter example: JaRIKo

```
C      PARM          $$$METO
C      PARM          $$$SVAR
C      EVAL      PRICE=$$$SVAR
C      CALC_VAT(PRICE)
```

In this example `CALC_VAT`, standing for "calculate vat" does not exist in RPG, but the interpreter can handle it as a language keyword, with type check, syntax highlighting, code suggestions, intellisense, errors and so on.

How does an interpreter work?

```
fun evaluate(expr) : Object {  
  if expr is StringLiteral  
    return expr.value  
  if expr is IntLiteral  
    return expr.value  
  if expr is SumExpr  
    return evaluate(expr.left) + evaluate(expr.right)  
  if expr is GreaterThanExpr  
    return evaluate(expr.left) > evaluate(expr.right)  
  ...  
}
```

How does an interpreter work?

```
fun execute(stmt) {  
    if stmt is PrintStmt  
        System.out.println(evaluate(stmt.argument))  
    if stmt is AssignStmt  
        symbolTable[stmt.var] = evaluate(stmt.value)  
    if stmt is CallFunctionStmt  
        argValues = stmts.args.map(evaluate(it))  
        executeFunction(stmt.function, argValues)  
    if stmt is IfStmt  
        conditionValue = evaluate(stmt.condition)  
        if (conditionValue)  
            execute(stmt.thenBranch)  
        else  
            execute(stmt.elseBranch)  
    ...  
}
```

How does an interpreter work?

```
fun execute(stmt) {  
    if stmt is PrintStmt  
        System.out.println(evaluate(stmt.argument))  
    if stmt is AssignStmt  
        symbolTable[stmt.var] = evaluate(stmt.value)  
    if stmt is CallFunctionStmt  
        argValues = stmts.args.map(evaluate(it))  
        javaClassForFunction = canFindJavaFunctionFor(stmt.function)  
        if (javaClassForFunction != null)  
            javaClassForFunction.invoke(argValues)  
        executeFunction(stmt.function, argValues)  
    if stmt is IfStmt  
        conditionValue = evaluate(stmt.condition)  
        if (conditionValue)  
            execute(stmt.thenBranch)  
        else  
            execute(stmt.elseBranch)  
    ...  
}
```

Resources

The ANTLR Mega Tutorial:

<https://tomassetti.me/antlr-mega-tutorial/>

For other articles visit:

<https://tomassetti.me>

Thinking of doing an industrial thesis?

Write at federico@strumenta.com and let's talk!

Questions?

Federico Tomassetti,
Language Architect at Strumenta