

Improving App Inventor Usability via Conversion Between Blocks and Text

Karishma Chadha, MIT Lincoln Laboratory
Franklyn Turbak, Wellesley College

Problem

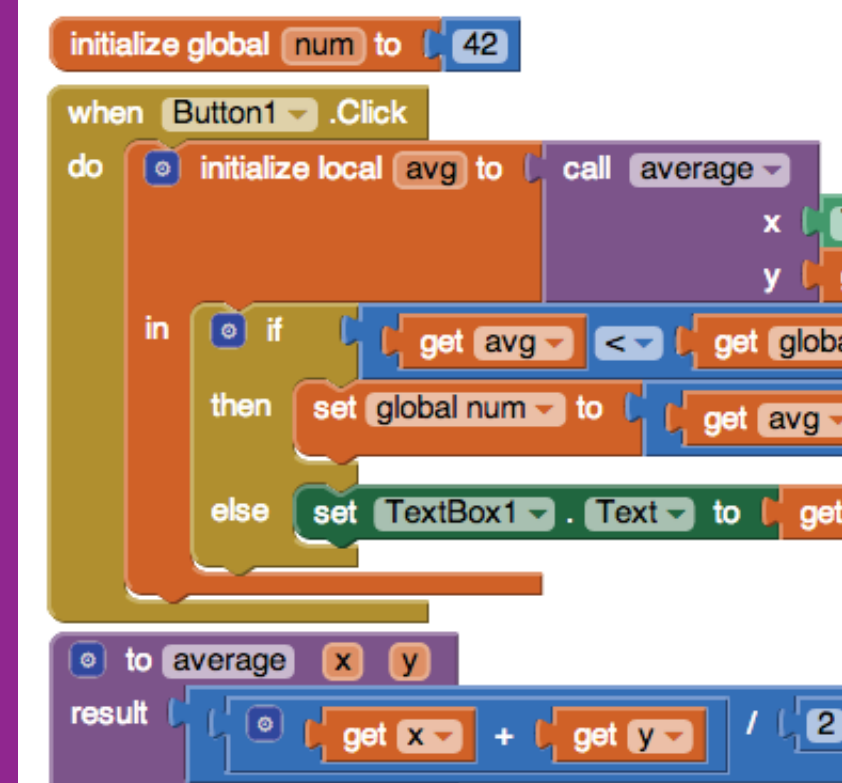
MIT App Inventor 2 (AI2), a popular online environment for Android app development, democratizes programming through its easy-to-use blocks language. While simple blocks programs are easy to read and write, complex ones become overwhelming. Creating and navigating nontrivial blocks programs is tedious, and AI2's current inability to copy blocks between projects inhibits reusing code between projects and programmers.

Solution

TAIL (Textual App Inventor Language)

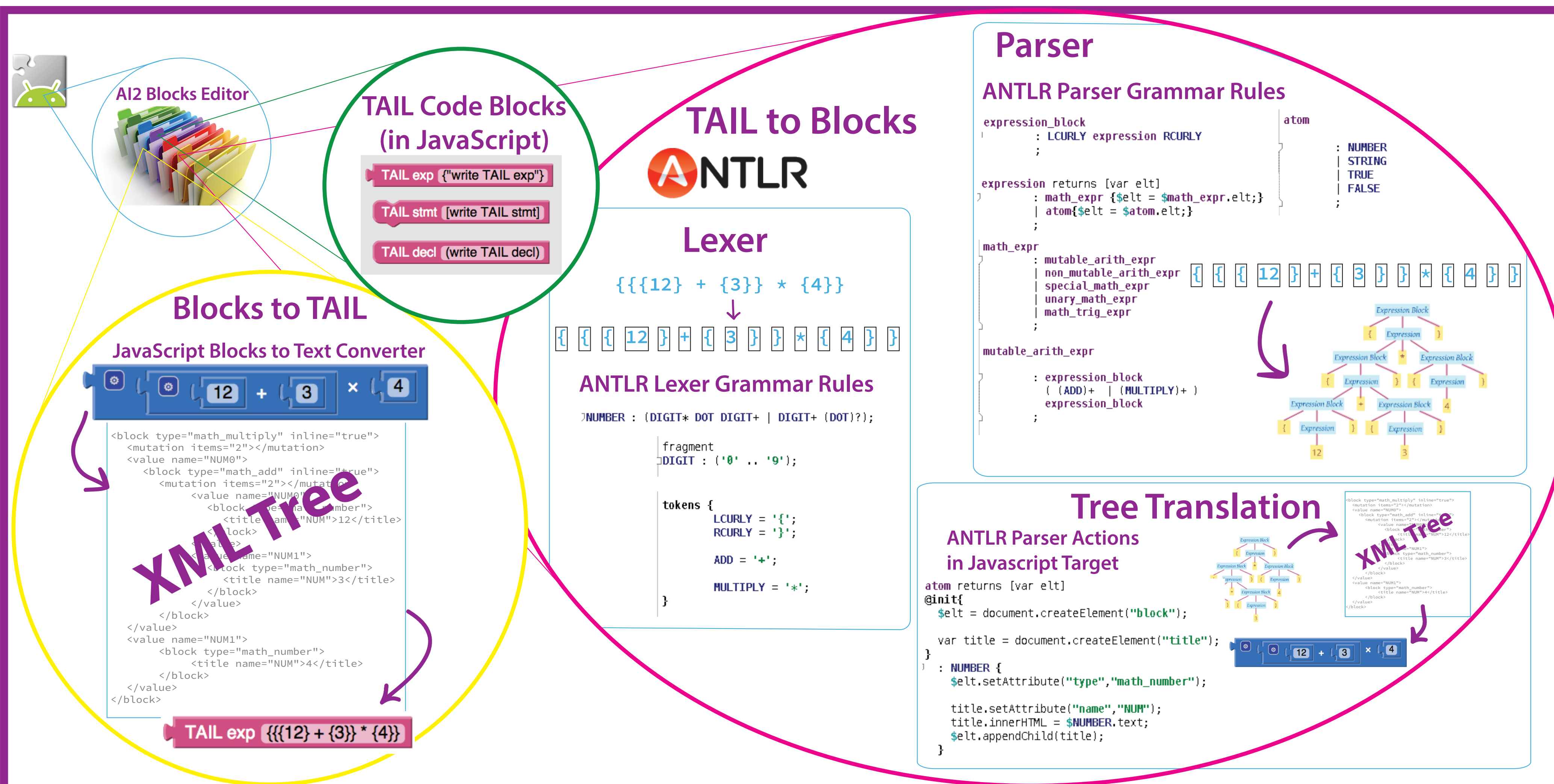
To address these issues, we have created a new textual language, TAIL, that is **isomorphic** to AI2's blocks language and provided a means for converting between regular blocks and code blocks that contain TAIL. This project aims to (1) increase AI2's usability by providing an efficient means for reading, constructing, and reusing programs, and (2) ease users' transitions from blocks programming to text programming.

TAIL Language Design

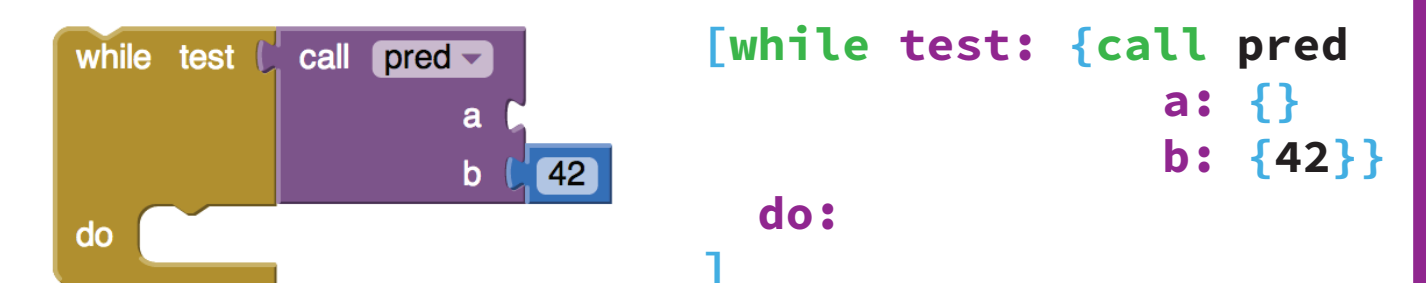


```
(initialize_global <num> to: {42})
(when Button1.Click
do: [initialize_local <avg>
to: {call average
x: {TextBox1.Text}
y: {get global num}}
in: [if {{get avg} < {get global num}}
then: [set global num to: {{get avg} / {2}}]
else: [set TextBox1.Text to: {get avg} ]]])
(to <average> <x> <y>
result: {{{get x} + {get y}} / {2}})
```

Conversion Between Blocks and TAIL: The Details



Empty Sockets



Blocks \longleftrightarrow Text Conversion

