PICTORIAL VERSION OF THE PAL SUBTREE TRANSFORMATIONAL GRAMMAR

```
     let     =>      gamma   |       where   =>      gamma
     / \             /   \   |        / \            /  \
    =   P         lambda E   |       P   =       lambda  E
   / \             /   \     |          / \       /  \
  X   E           X    P     |         X   E     X    P

     tau     =>    ++gamma   |         ->    =>      gamma
      |              /  \    |       / | \           /   \
     E++          gamma  E   |      B  T  E       gamma nil
                   /  \      |                    /   \
                aug  .nil    |                gamma    lambda
                             |                /  \        /  \
                             |             gamma lambda  ()   E
                             |             /   \   /  \
                             |          Cond    B ()   T

     not     =>      gamma   |       neg     =>      gamma
      |              /  \    |        |              /  \
      E            not   E   |        E            neg   E

   within    =>        =     |       rec     =>        =
    /   \             /  \   |        |                / \
  =     =           X2  gamma|        =              X  gamma
 / \   / \                /  |       / \                /   \
X1 E1 X2 E2        lambda  E1 |      X   E             Ystar  lambda
                    /  \      |                                /  \
                  X1   E2     |                               X    E

   fcn_form  =>      =        |      lambda   =>      lambda
   /   |  \        /   \      |       /  \            /  \
  P   V++  E       P ++lambda|      ,    E         Temp  ++gamma
                       /  \   |      |                  /   \
                      V   .E  |     X++i                lambda  gamma
                              |                        /  \     /  \
                              |                      X.i  .E Temp   \
                              |                                <INTEGER:i>

     lambda  =>    ++lambda   |       Op      =>      gamma
     /   \           /     \  |      / \             /    \
    V++   E         V    .E   |     E1  E2         gamma  E2
                              |                    /  \
                              |                   Op  E1

     and     =>      =        |        @      =>      gamma
      |             / \       |      / | \           /  \
     =++           ,  tau     |    E1 Op E2       gamma  E2
     / \           |   |      |                   /  \
    X   E         X++  E++    |                  Op   E1
                              |  [where Op =
                              |        aug or & + - / ** gr ...]
```