CKY example

Based on these slides by Dávid Nemeskey

The input sentence

The dog bit John

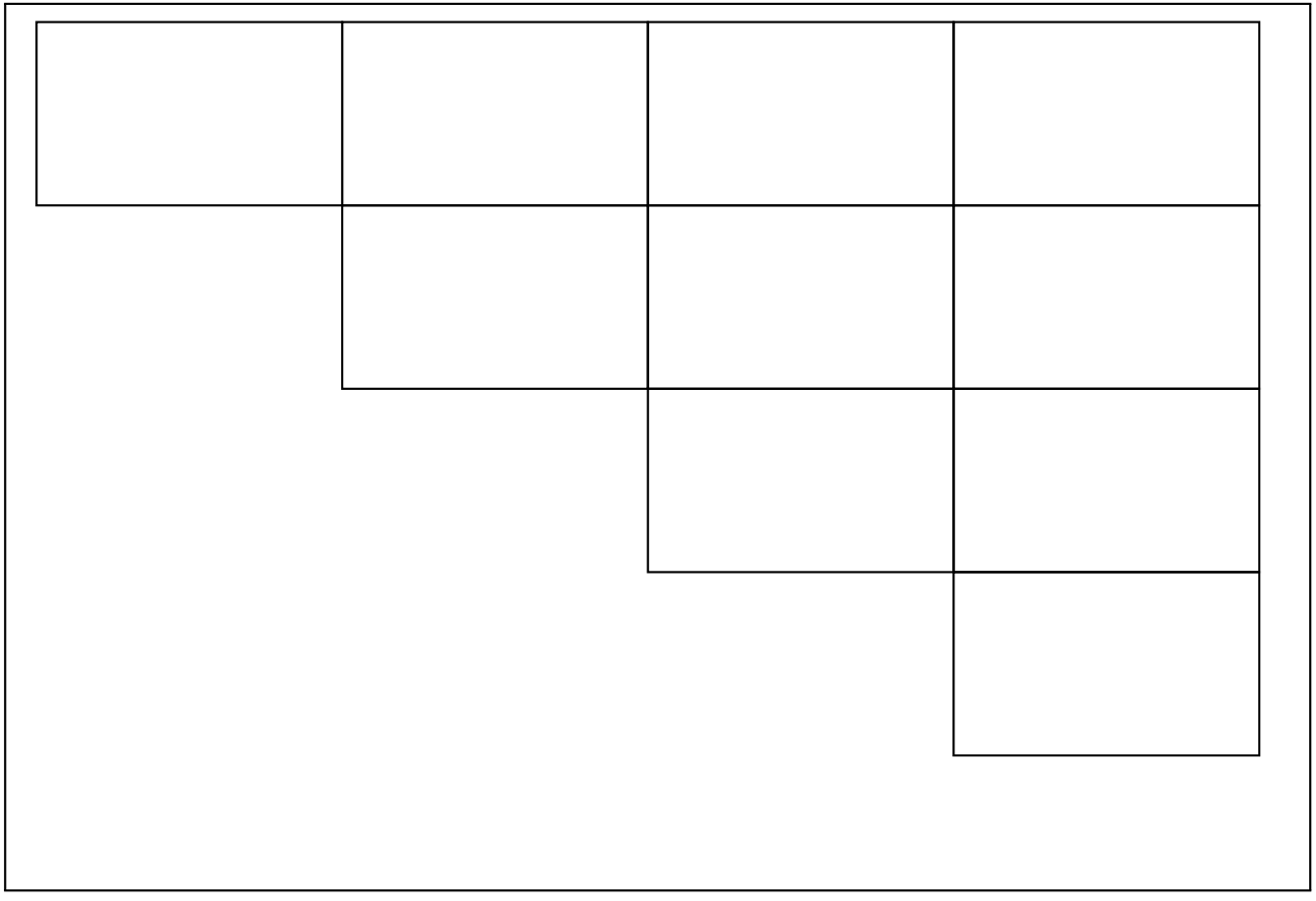
The grammar

```
S          →   NP VP
           |   ProperNoun VP
           |   NP Verb
           |   ProperNoun Verb

NP         →   Det Nominal
           |   Det Noun

Nominal    →   Nominal Noun
           |   Noun Noun

VP         →   Verb NP
           |   Verb ProperNoun
```

Example: CKY – The dog bit John.

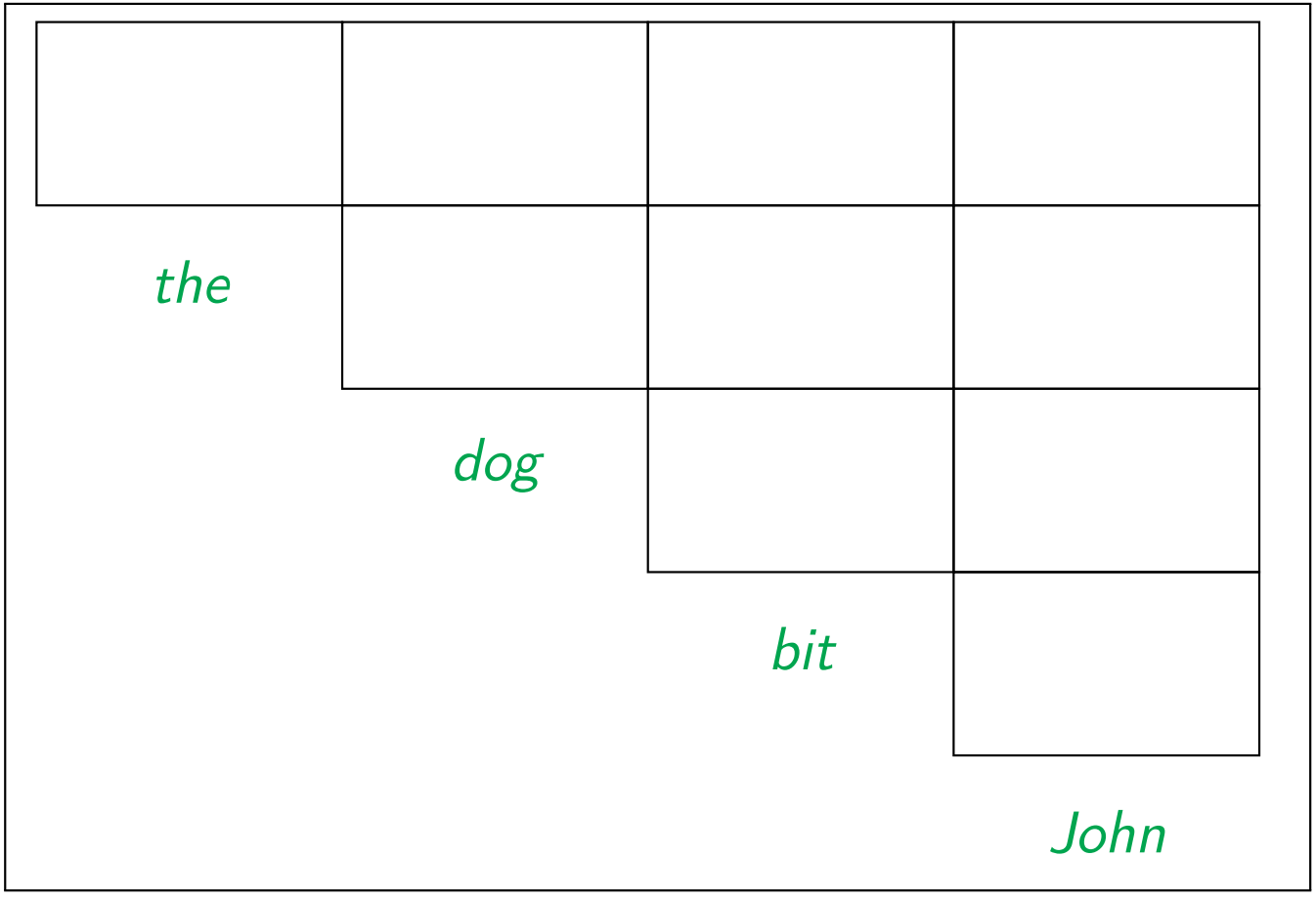
Now we can apply the CKY algorithm...

Example: CKY – The dog bit John.



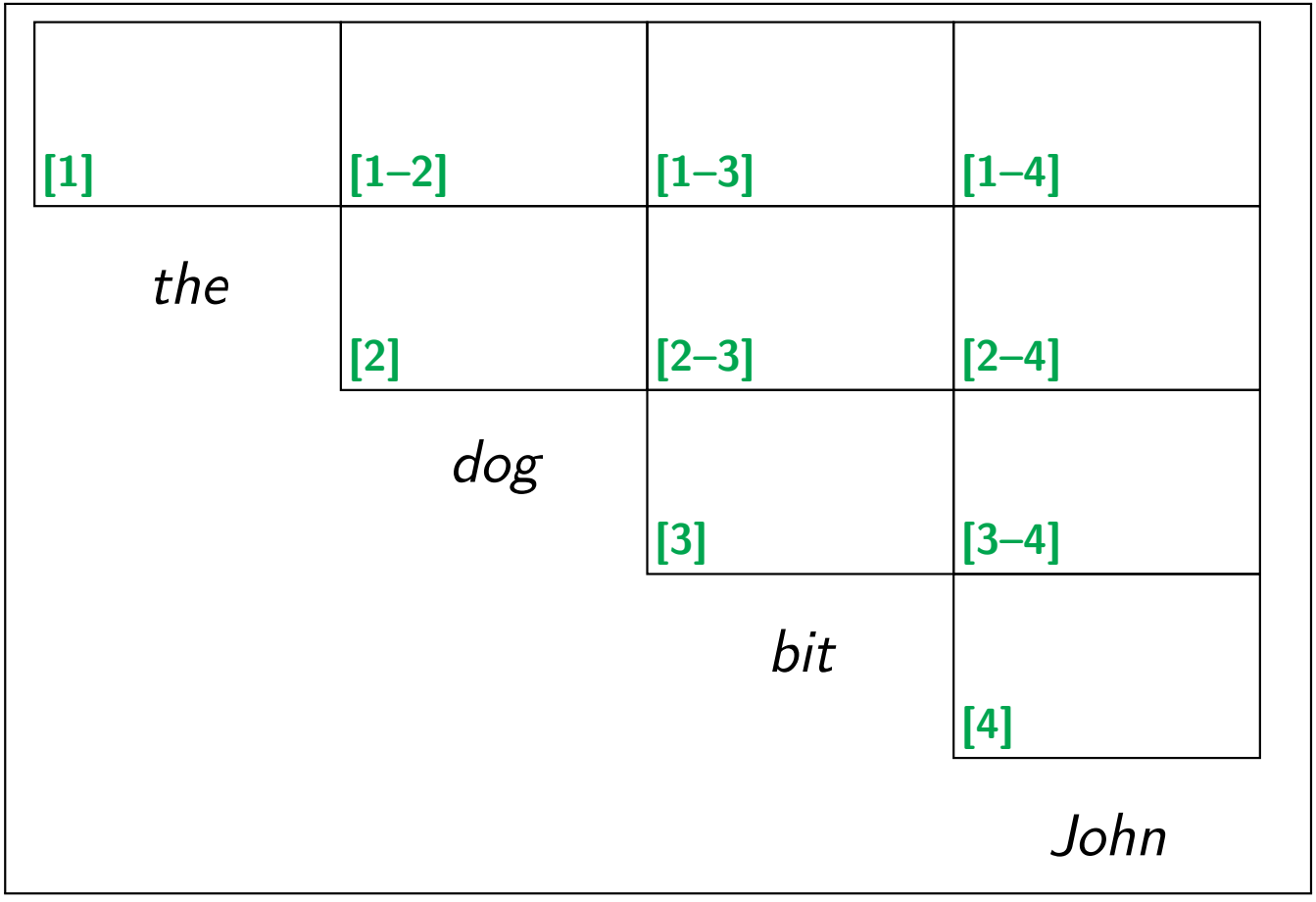
First, we start by drawing the upper triangular matrix

Example: CKY – The dog bit John.



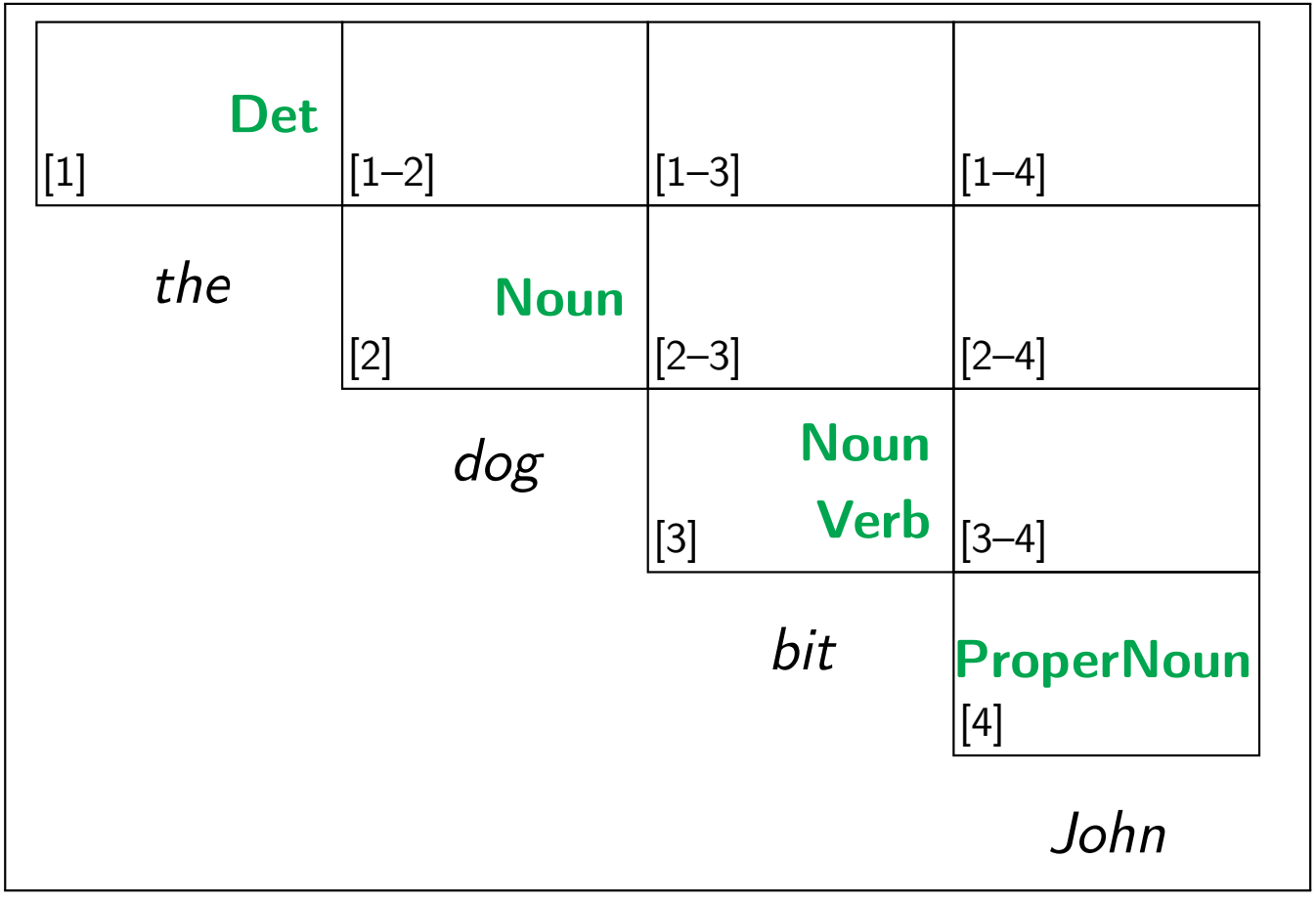
Each column is assigned to a word in the sentence

Example: CKY – The dog bit John.



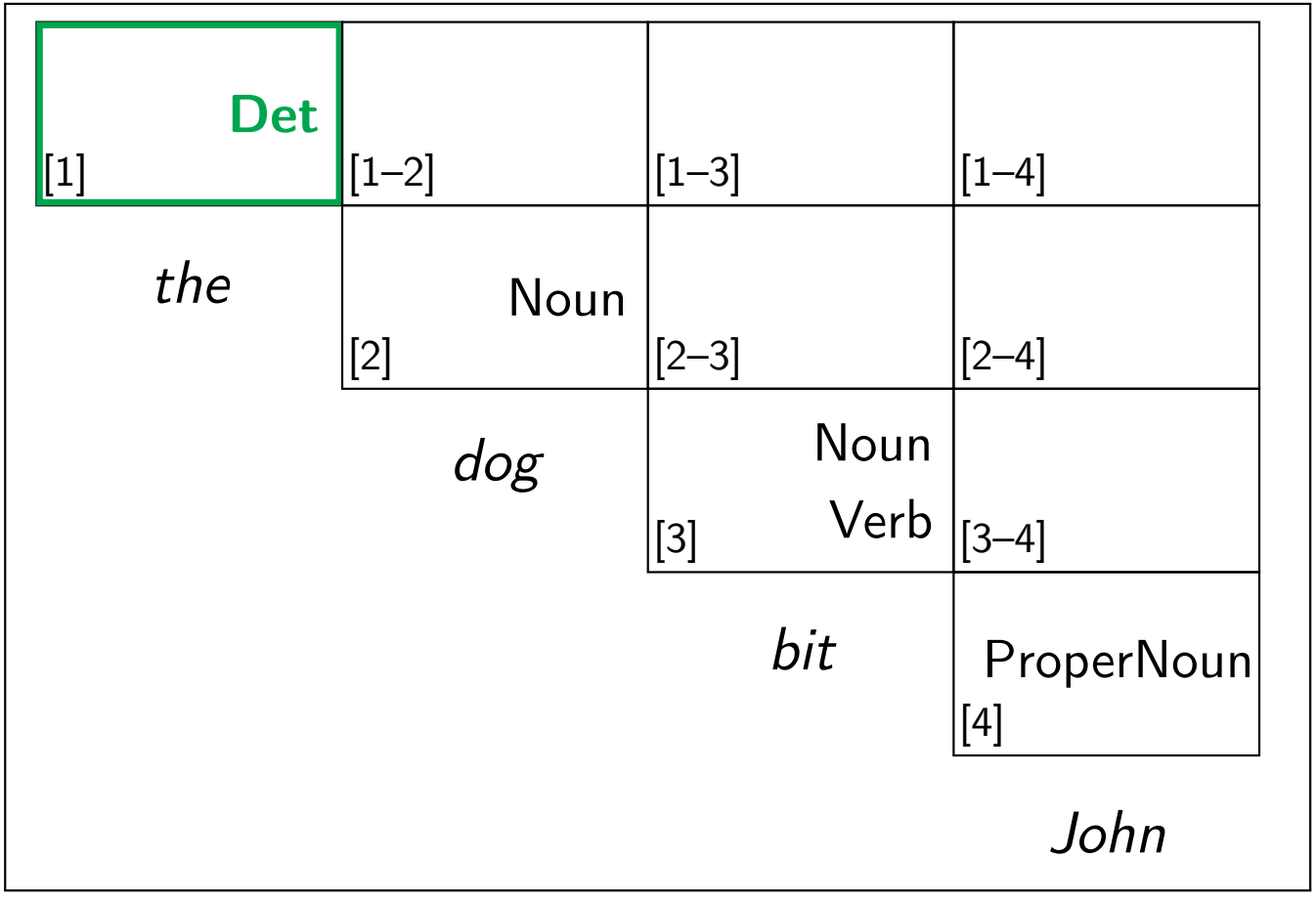
We note the word indices each cell corresponds to

Example: CKY – The dog bit John.



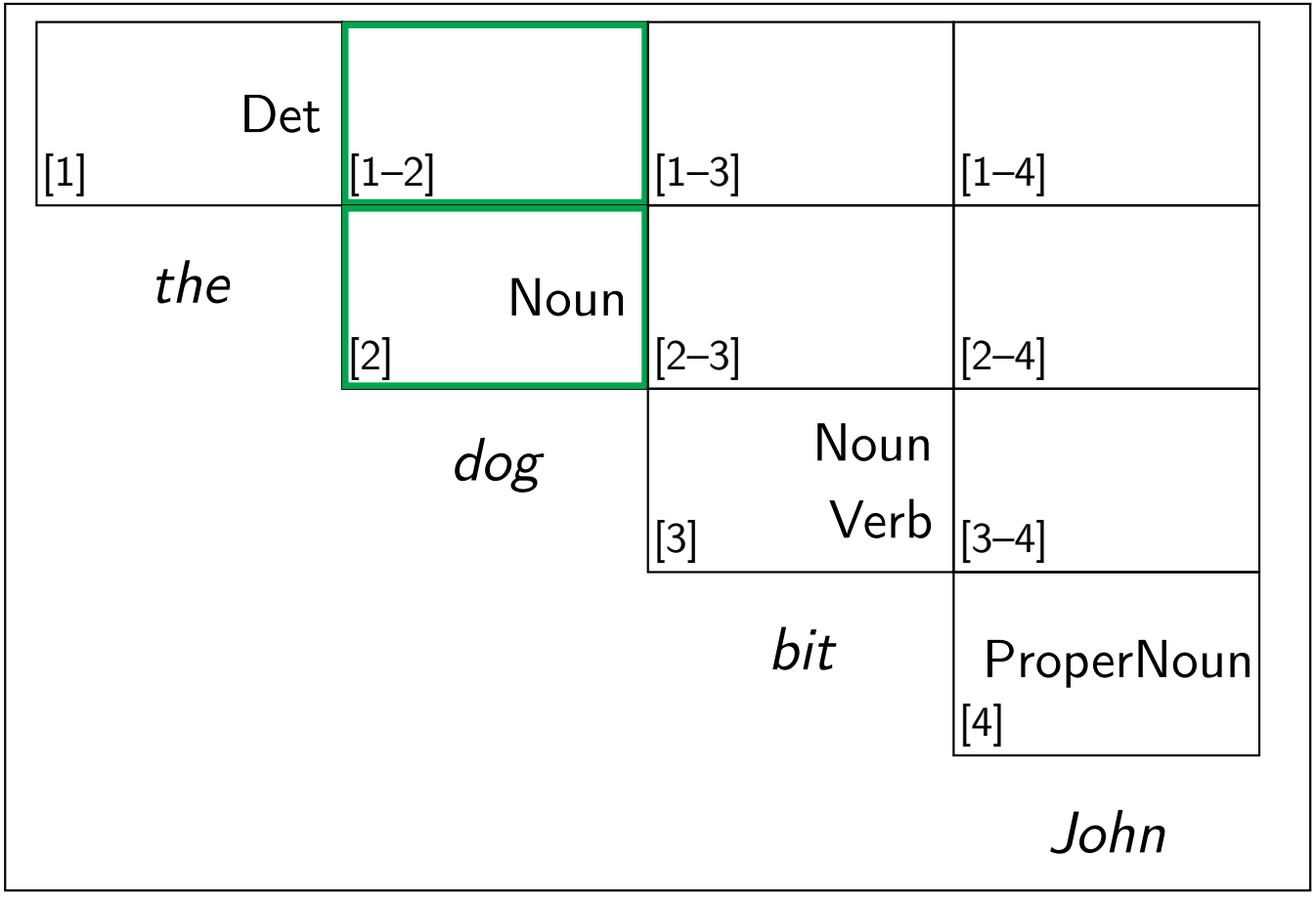
The diagonal records the terminal production rules (POS). Ambiguity: *bit* can also be a noun

Example: CKY – The dog bit John.



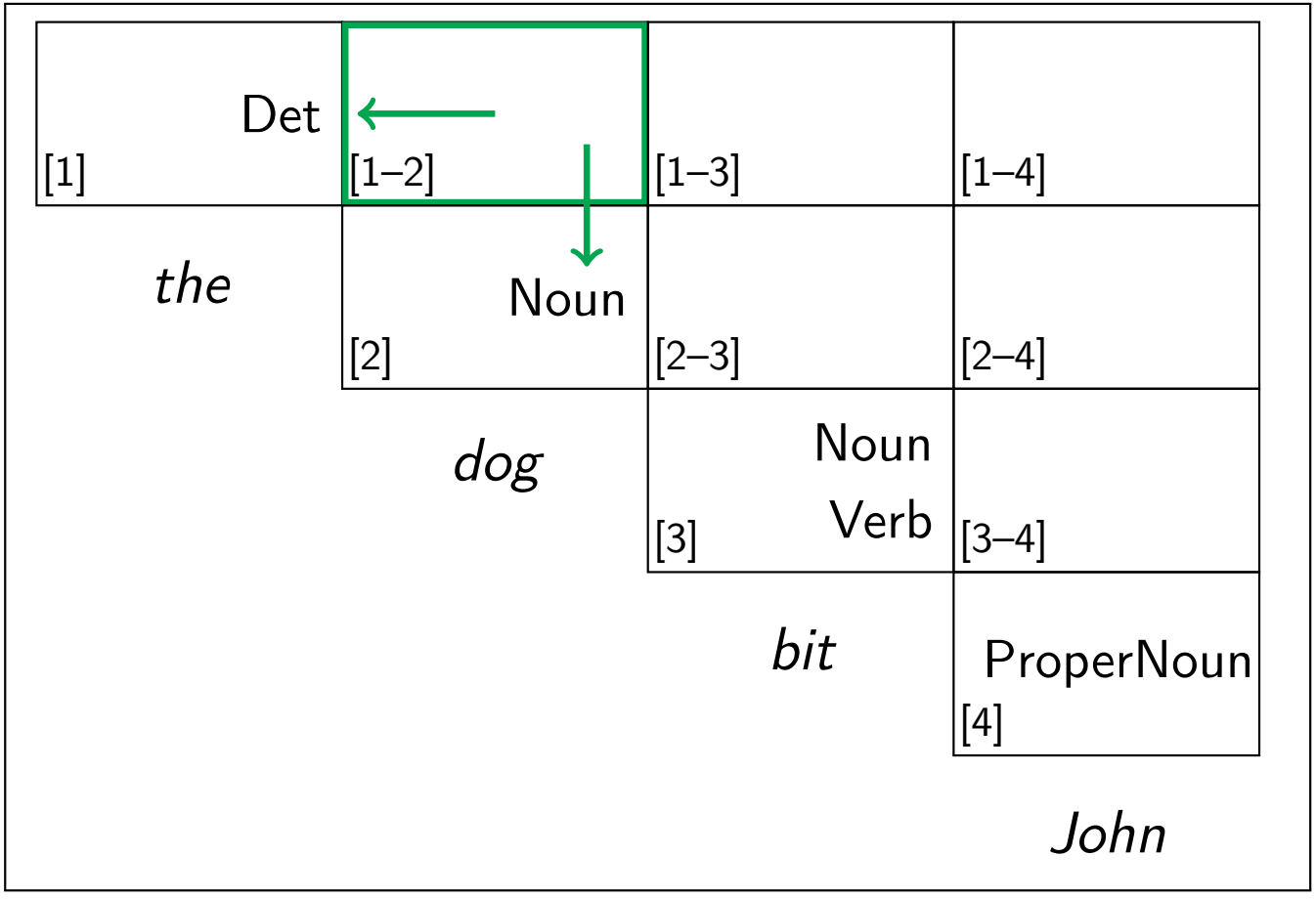
There is not much else to do in the first column

Example: CKY – The dog bit John.



On to the second column...

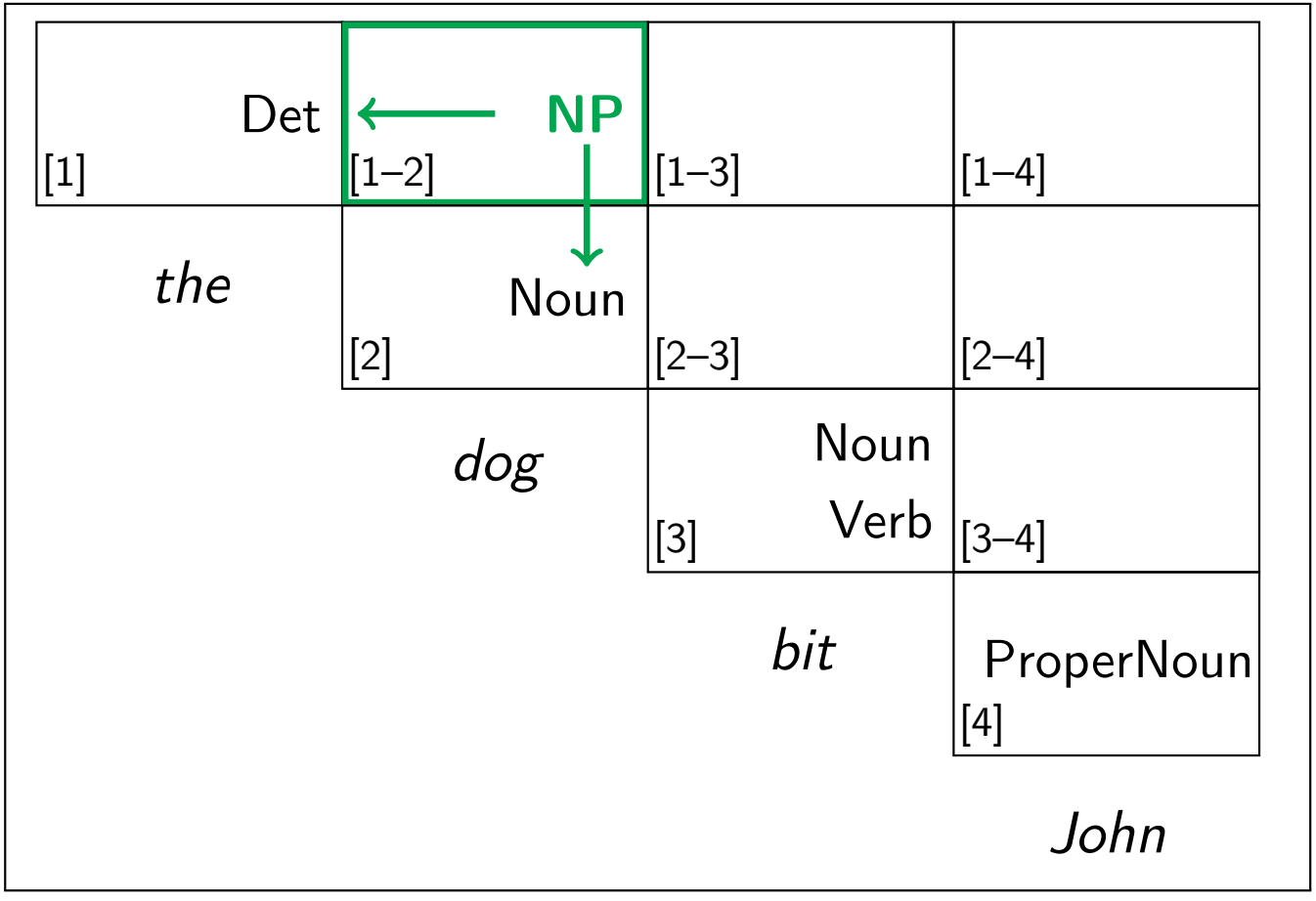
Example: CKY – The dog bit John.



Non-leaf cells correspond to binary rules:

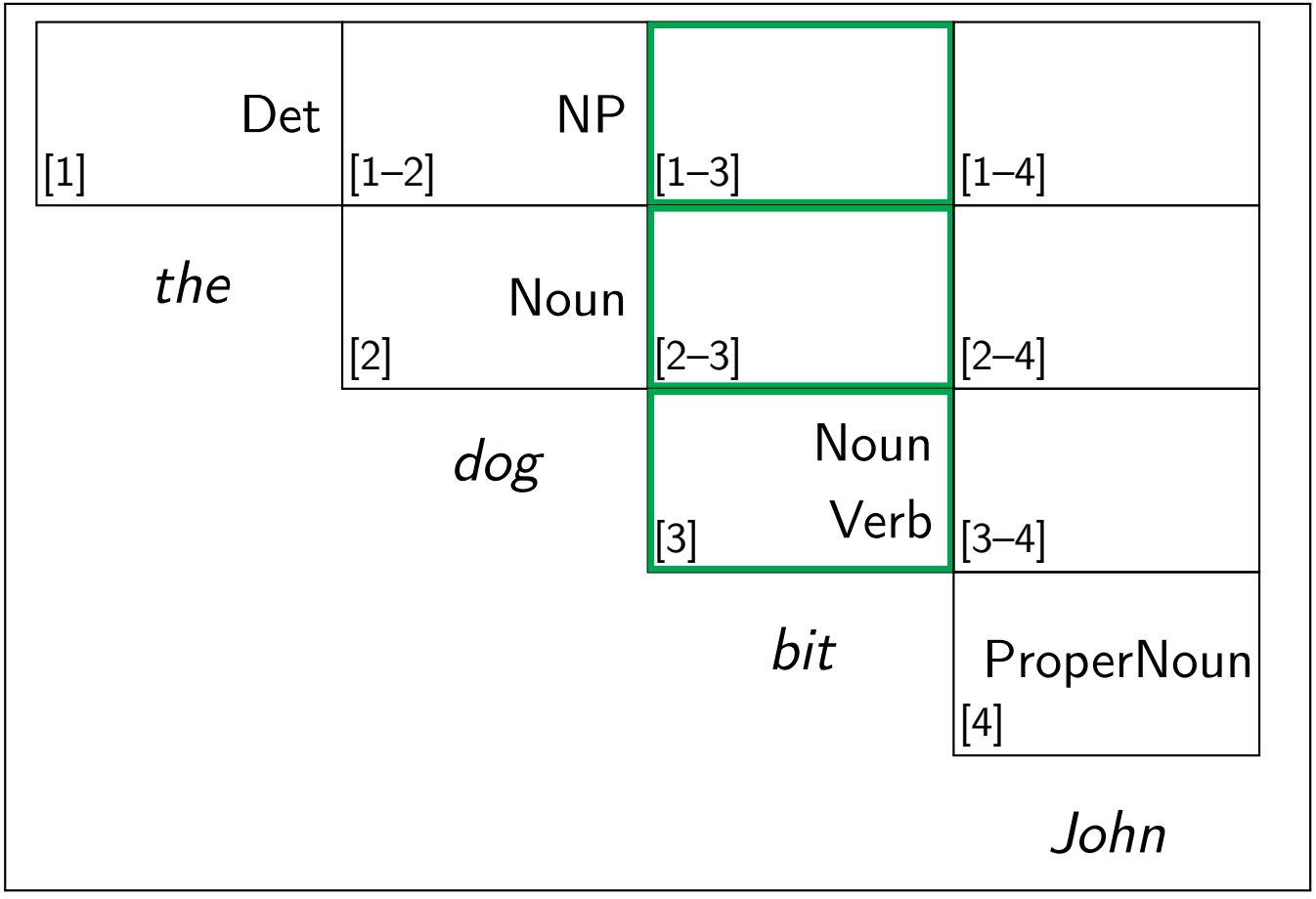
- the first constituent on the right side of the rule is to the *left*
- the second one is *down*

Example: CKY – The dog bit John.



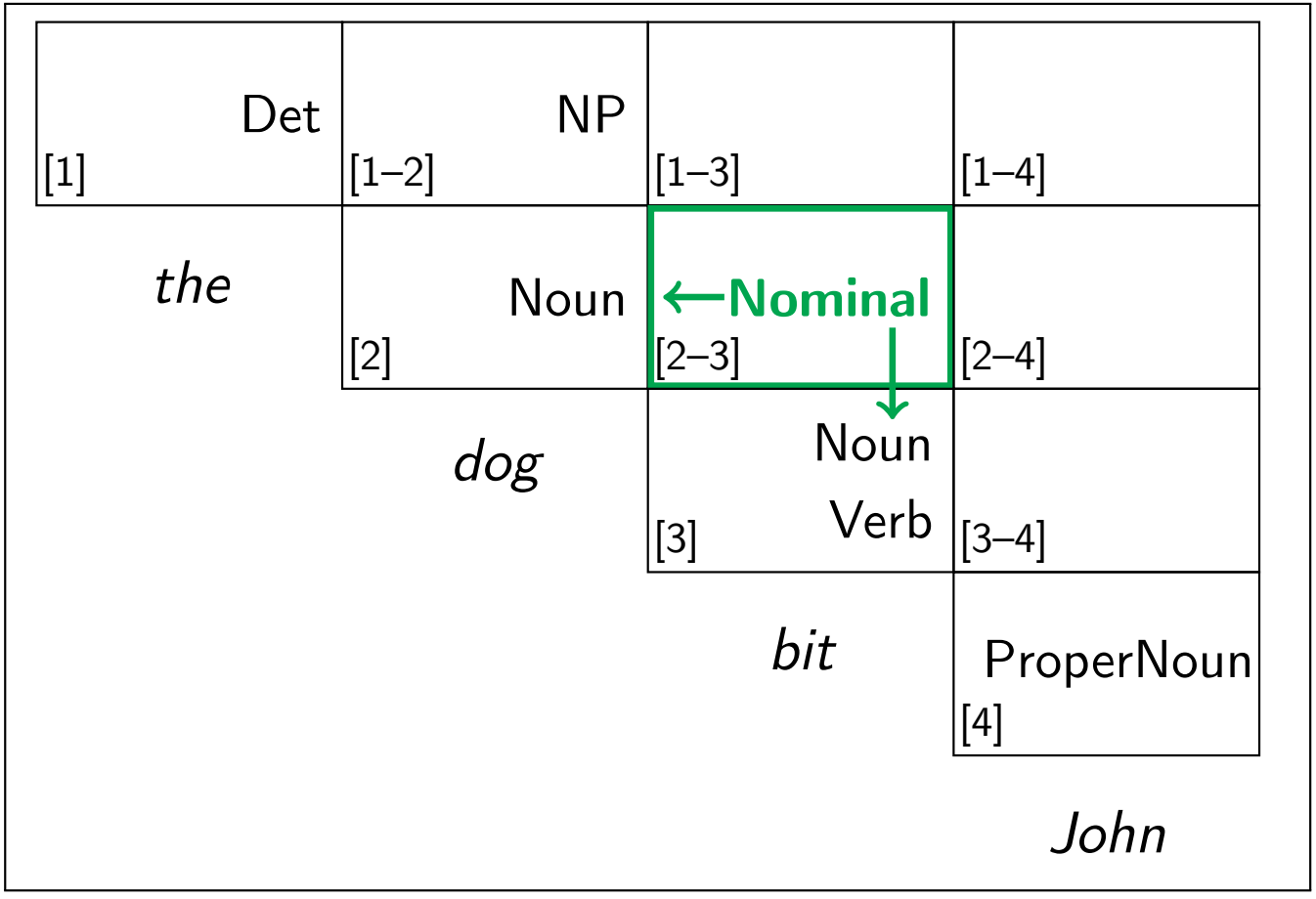
The first rule application: `NP` $\rightarrow$ `Det Noun`

Example: CKY – The dog bit John.



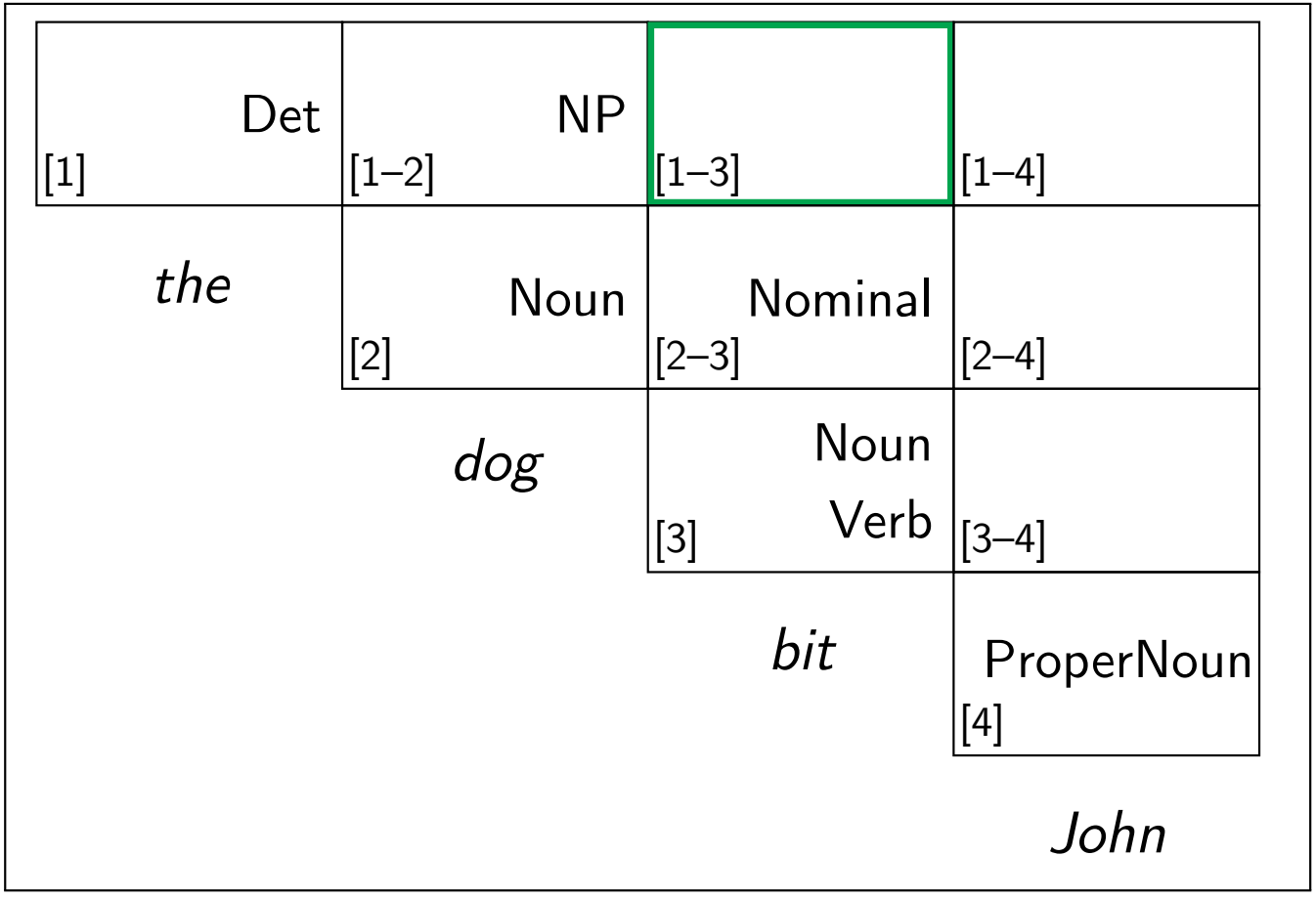
On to the third column...

Example: CKY – The dog bit John.



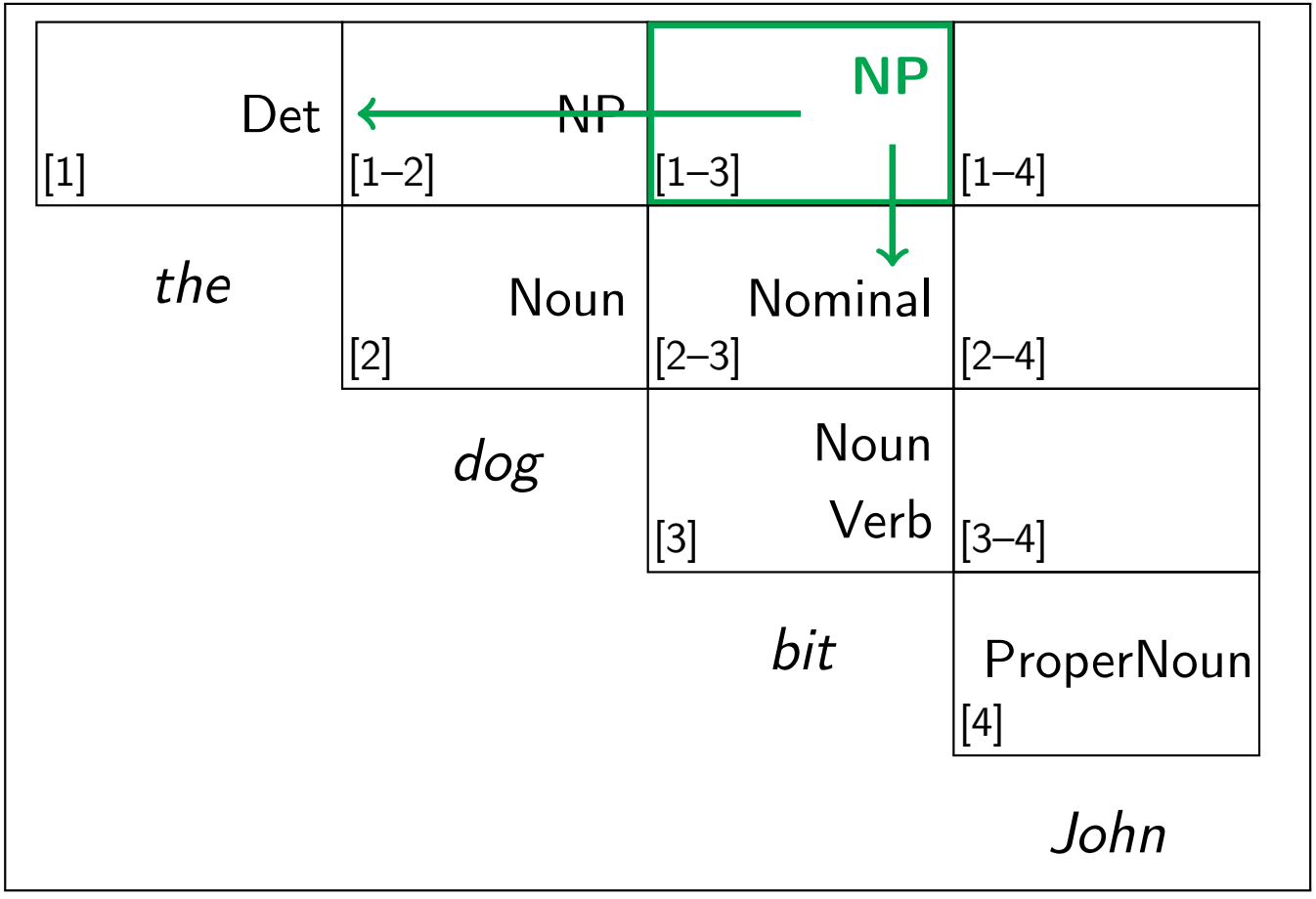
Rule application: `Nominal` → `Noun Noun`

Example: CKY – The dog bit John.



In general, non-leaf cells with N words can be split into two in $N - 1$ ways.

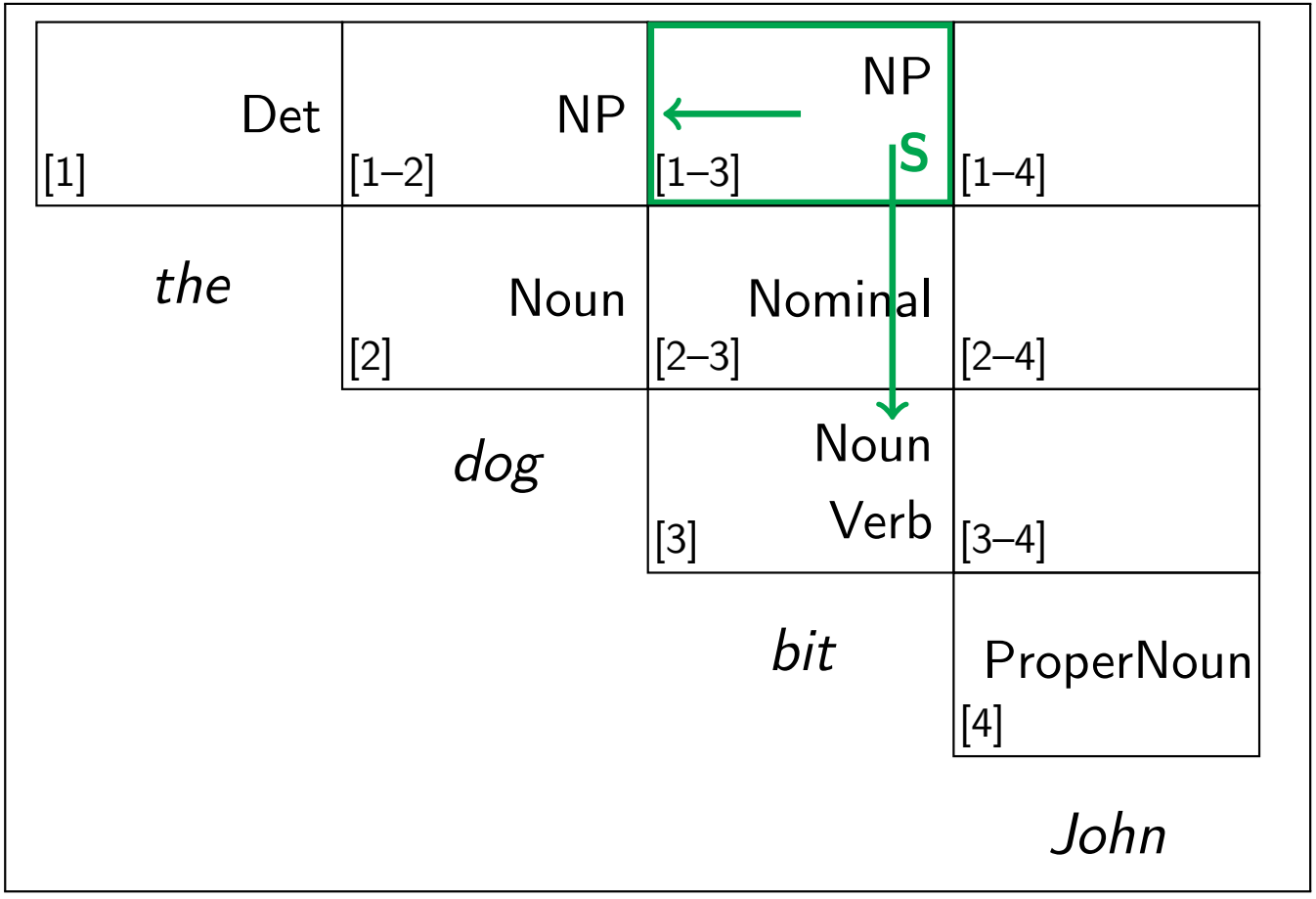
Example: CKY – The dog bit John.



In general, non-leaf cells with N words can be split into two in $N-1$ ways. When $N=3$:

- 2 to the left, 1 down: rule application `NP` $\rightarrow$ `Det Nominal`

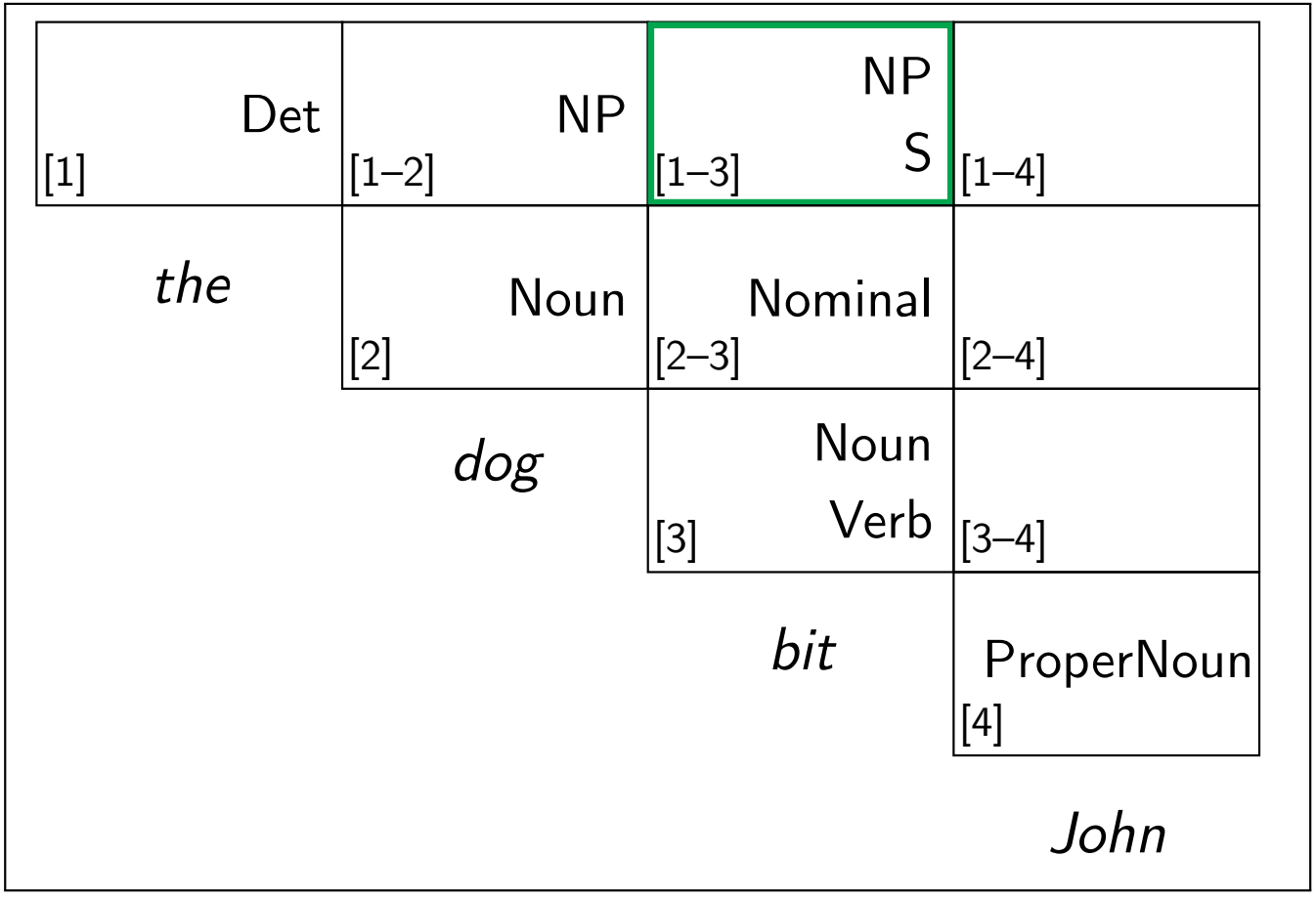
Example: CKY – The dog bit John.



In general, non-leaf cells with N words can be split into two in $N-1$ ways. When $N=3$:

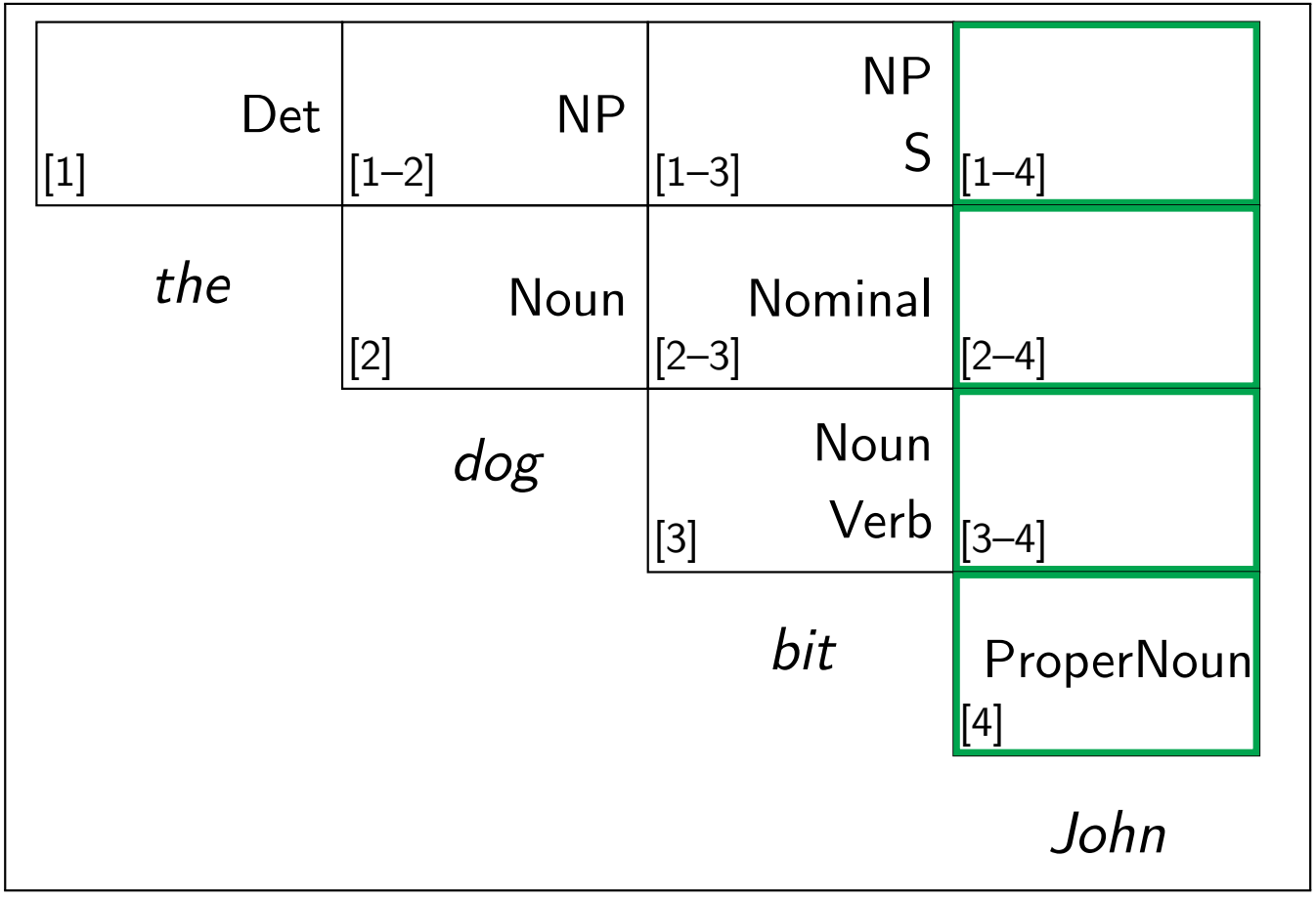
- 2 to the left, 1 down: rule application `NP` $\rightarrow$ `Det Nominal`
- 1 to the left, 2 down: rule application `S` $\rightarrow$ `NP Verb`

Example: CKY – The dog bit John.



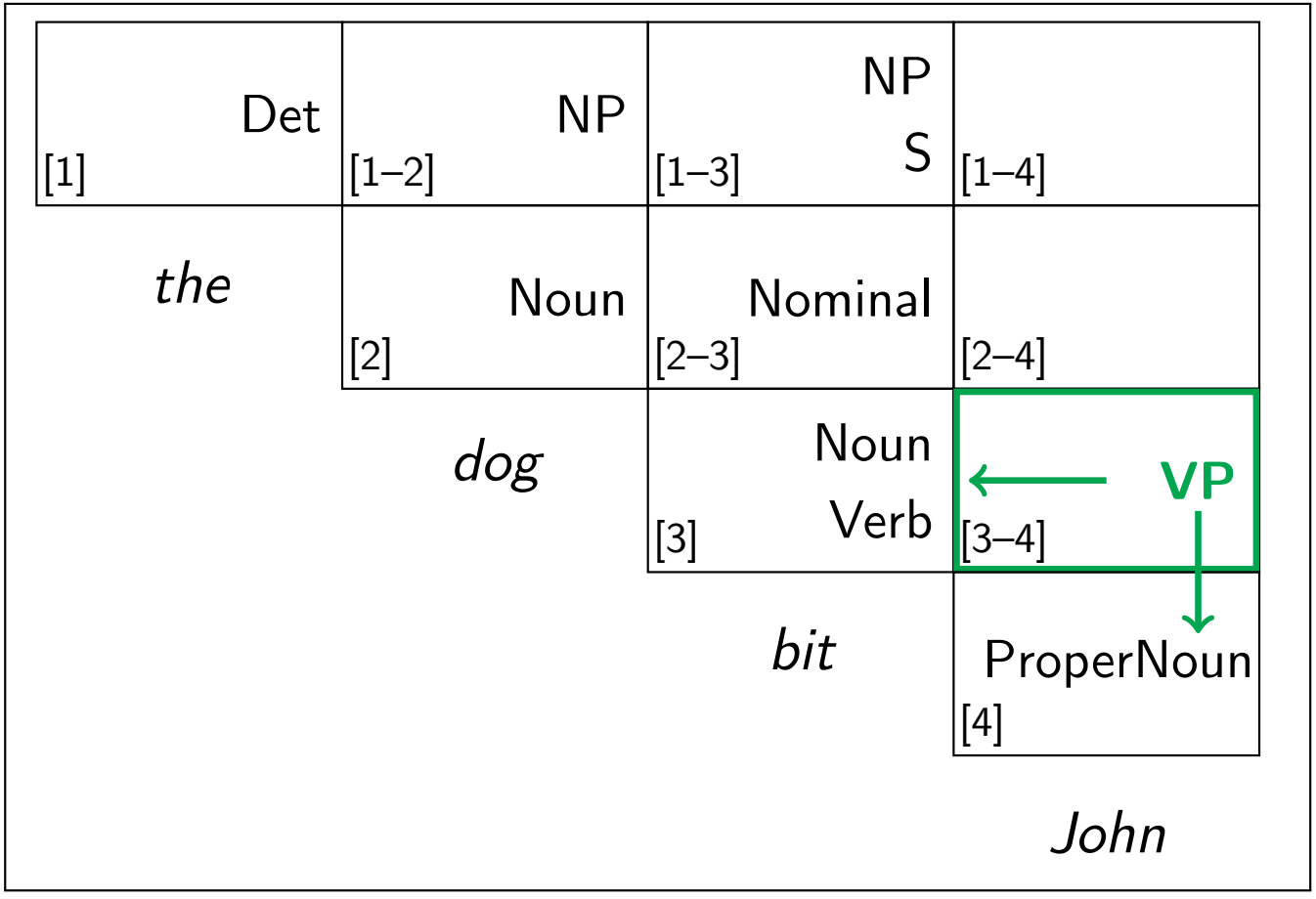
Here we found an S. However, it is not at the top right cell, so we are not done yet.

Example: CKY – The dog bit John.



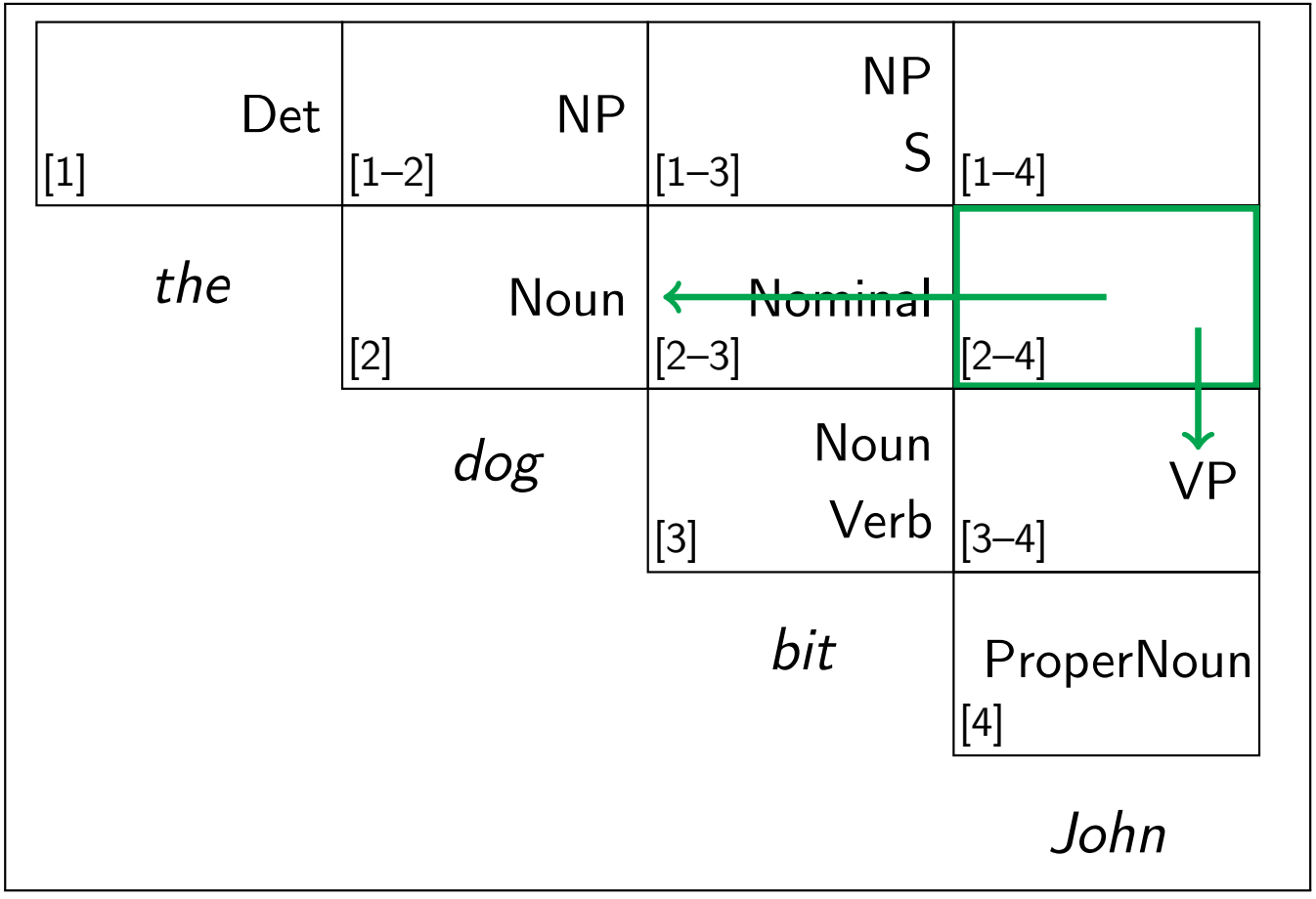
On to the fourth column...

Example: CKY – The dog bit John.



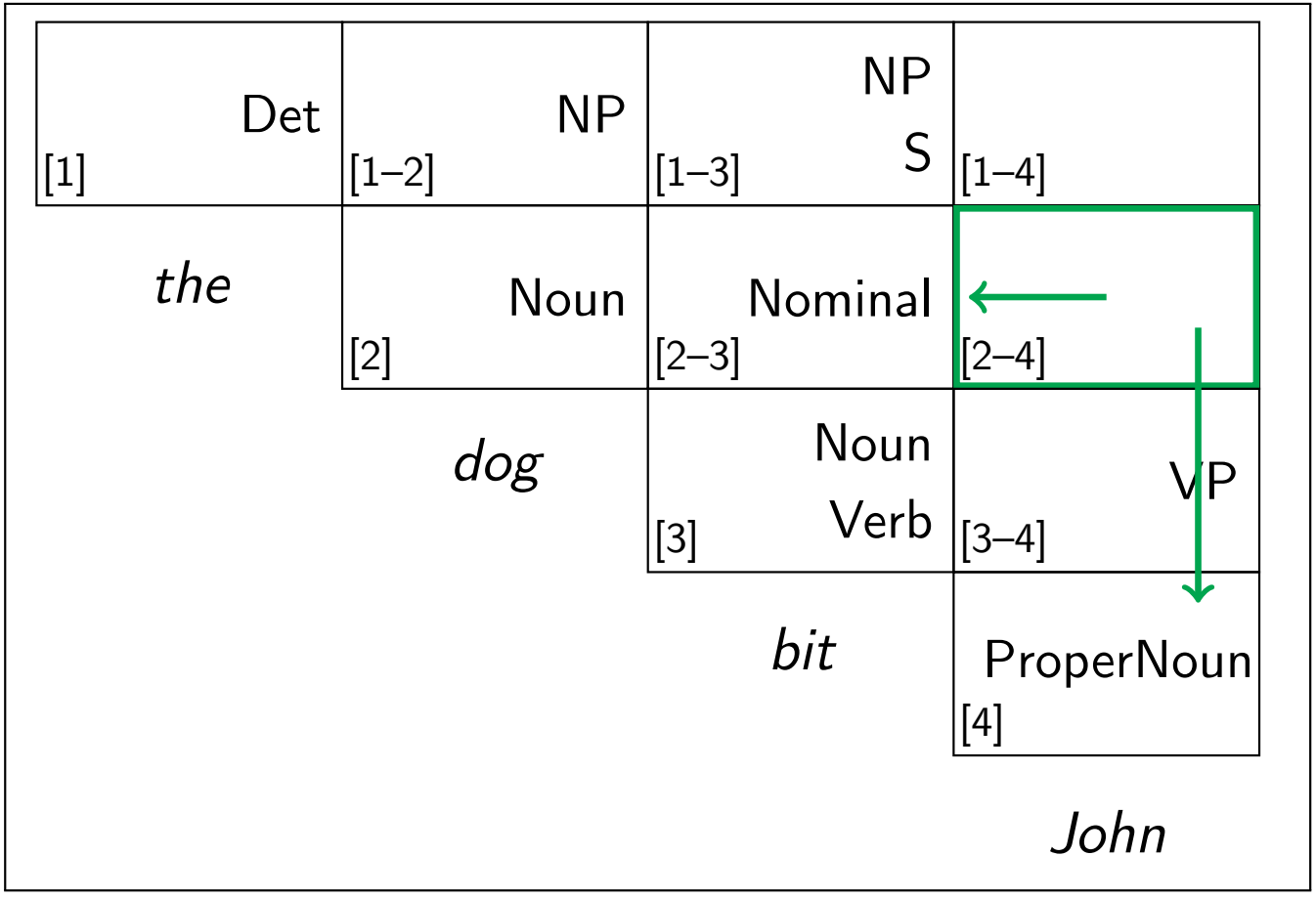
Rule application: `VP` → `Verb ProperNoun`

Example: CKY – The dog bit John.



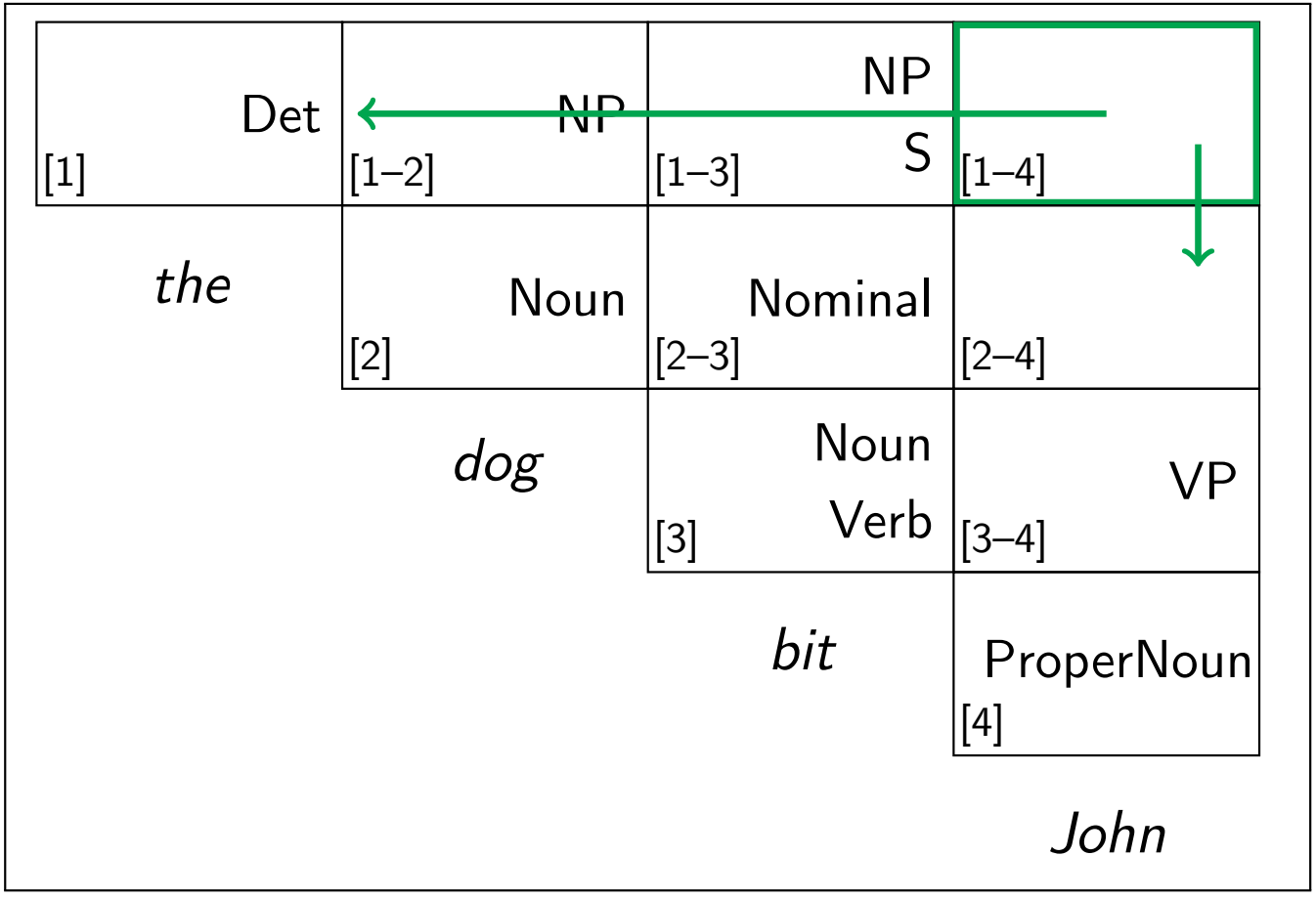
No applicable rules

Example: CKY – The dog bit John.



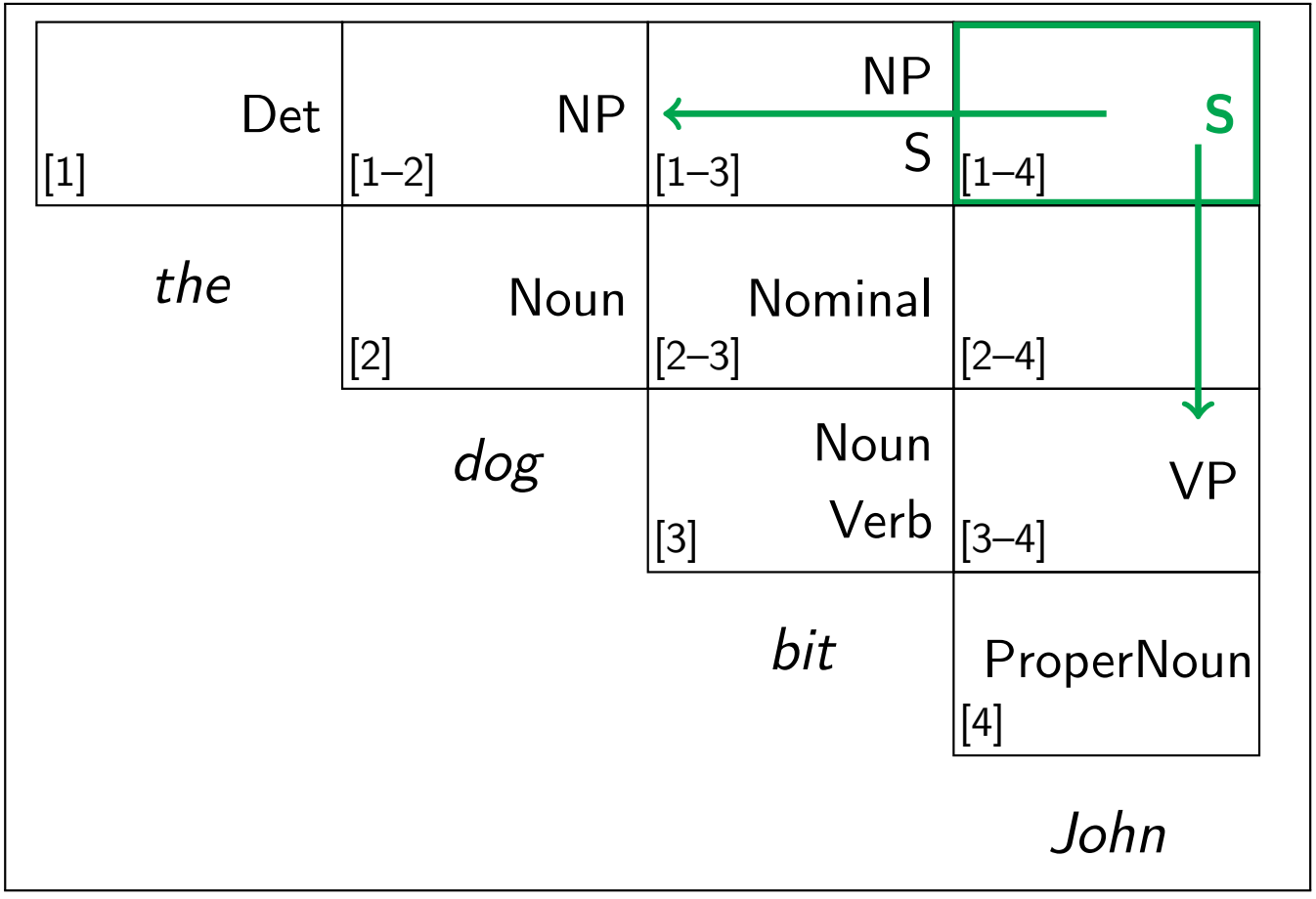
No applicable rules

Example: CKY – The dog bit John.



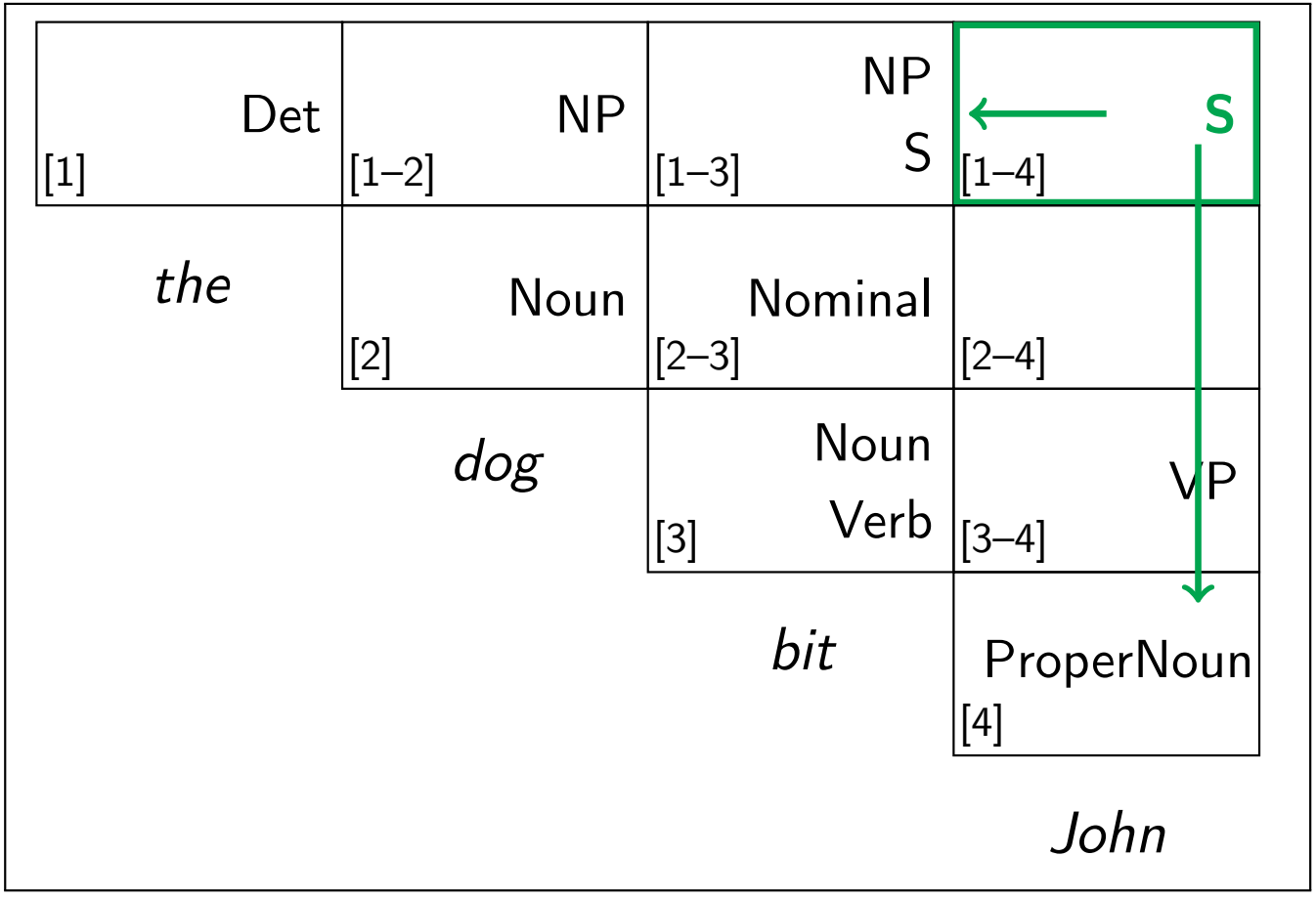
The top right cell represents the whole sentence.

Example: CKY – The dog bit John.

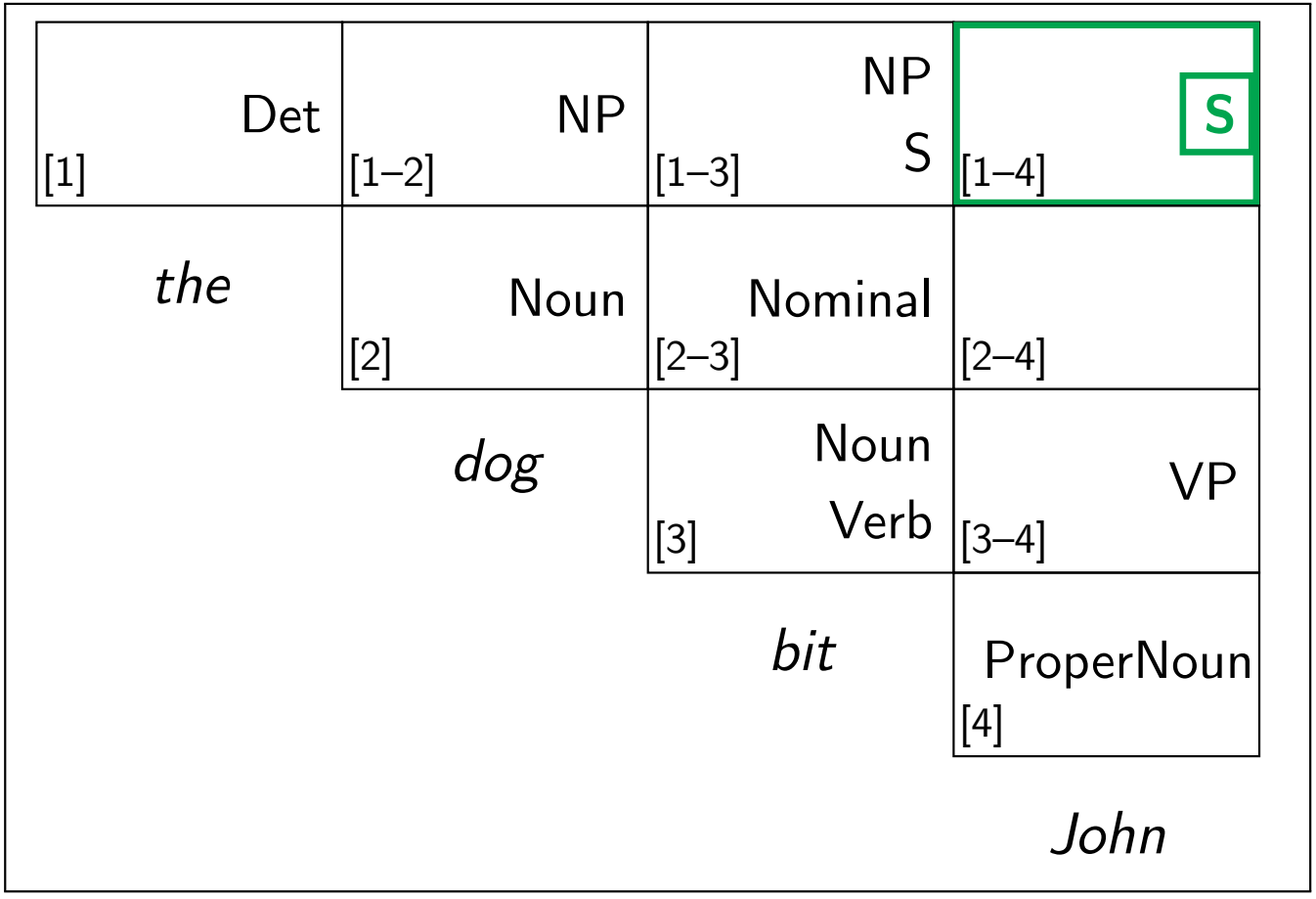


Rule application: S → NP VP

Example: CKY – The dog bit John.

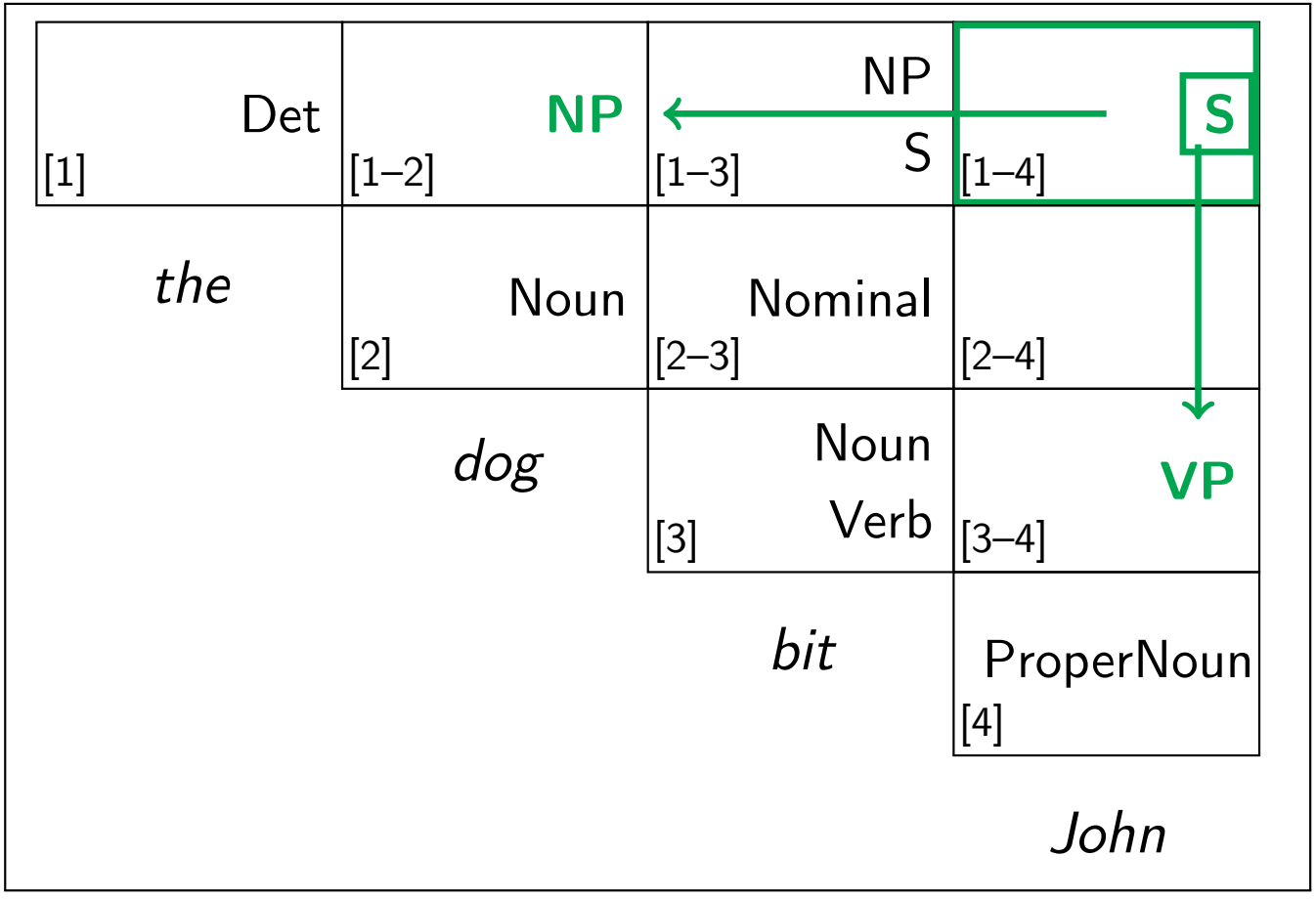


Example: CKY – The dog bit John.



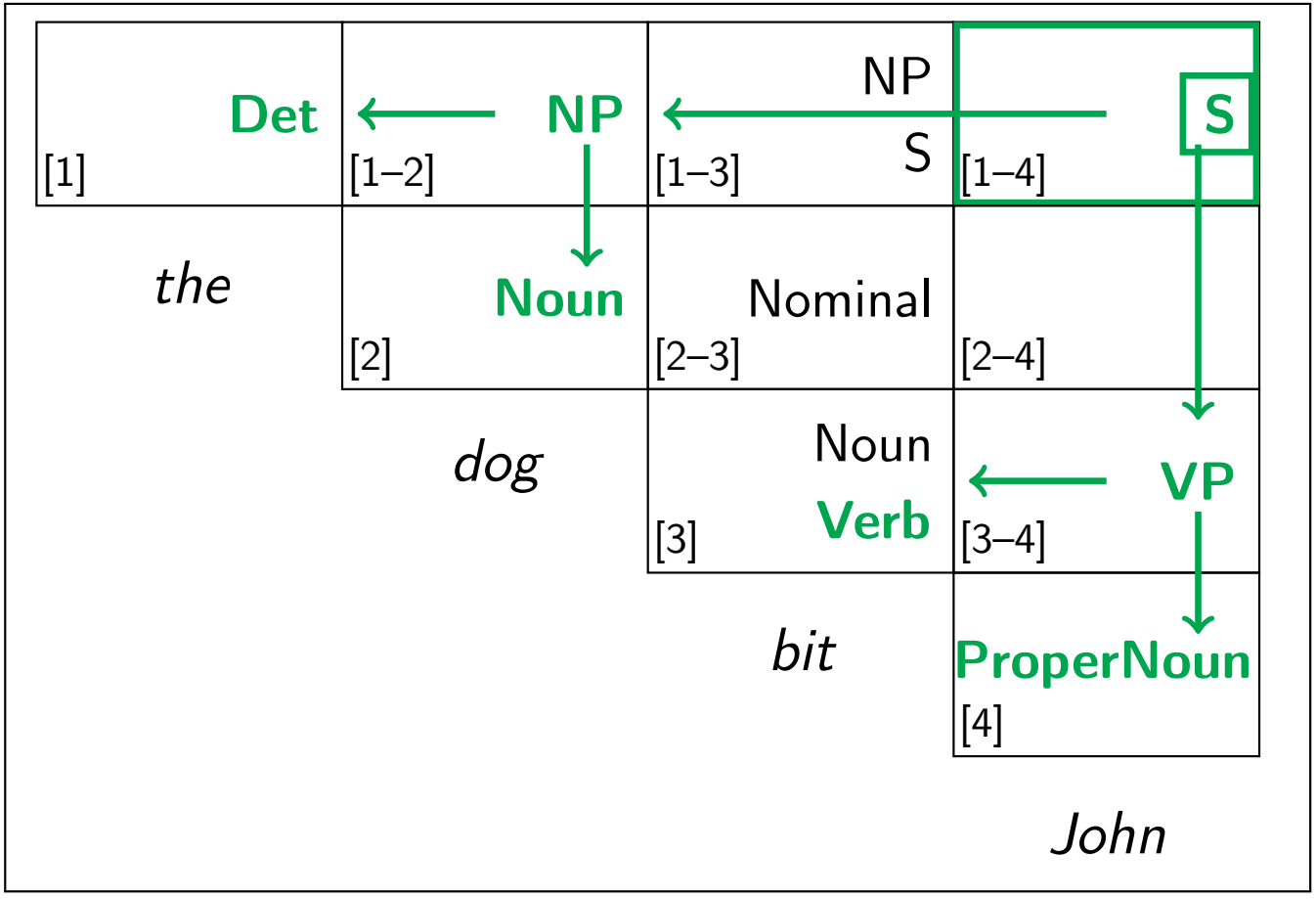
S in the top right cell: sentence accepted.

Example: CKY – The dog bit John.



Each nonterminal maintains backpointers to its children (here: NP and VP)

Example: CKY – The dog bit John.



The backpointers define the syntax tree.