Analele Ştiinţifice ale Universităţii "Alexandru Ioan Cuza" din Iaşi
Informatică, Tomul XIV, 2004, 1-?
Scientific Annals of the "Alexandru Ioan Cuza" University of Iaşi
Computer Science Section, Tome XIV, 2004, 1-?

Partial of

The impact of the operations upon complexity of WEB applications

Ivan Ion
Alecu Felician
Popa Marius

Abstract

Place the text of the abstract here.

1 WEB Applications

WEB applications are composed by several WEB pages connected through hyperlinks. There are a lot of methods and instruments used to create WEB pages and also there are some rules to be followed by the WEB designers.

An efficient way to create WEB application is to use prototypes. Developing an application starting with a prototype means the following:

- lower amount of resources to be used
- high degree of reusability
- a better task allocation for the team members
- shorter development phase
- better customer reviews
- lower costs
- high quality of the WEB application (both for WEB designer, customers and the final users)

The data structures used in a WEB application can lead to an easy to use and easy to maintain final product. Two major qualities should be followed at the time when the WEB applications are projected: homogeny and symmetry.

A tree data structure is homogenous if the leaf nodes are located on the same level.

The symmetry supposes the existence of identical structures for the sub-trees corresponding to any node that is not a leaf. Two WEB structures are identical if they have the same links and the same number of paths in the tree. The indicator called degree of symmetry (S, $0 \le S \le 1$) can be used to measure if two structures are similar. If the structures are identical, the indicator will be equal with 1. If the structures are completely different, the indicator will be 0.

At the time when a WEB application is designed it should be taken into account the possibility to auto organize the data structures according with the number of visits for the corresponding pages. After auto organization, a common user will follow a shorter path to reach the desired information and the homogeneity level of the application should be increased.

2 The complexity of WEB structures

2.1 McCabe Complexity

We consider a WEB application having a graph assigned (G). The nodes correspond to WEB pages and the oriented arches represent the existing links. The graph $G(NO, AR)$ is defined by NO (nodes, n_n in total) and AR (arches, n_a – the total number of links).

The *McCabe* complexity of graph G, also known as the cyclomatic number of the graph, is given by the following expression:

$$C = n_a - n_n + 2$$

The Figure 1 shows a WEB application having a linear structure.

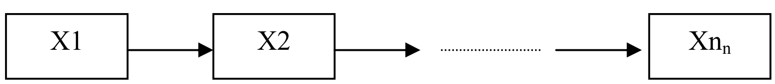

Figure 1 – WEB application having a linear structure

The graph has nodes $X1, X2, \ldots, Xn_n$ and the links $(X1, X2), (X2, X3), \ldots, (Xn_{n-1}, Xn_n)$. So, the linear structure has n_n nodes and n_n-1 links, so its complexity will be equal with

$$C = (n_n-1) - n_n + 2 = 1$$

This is why we can say with no doubt that any linear structure has its complexity equal with 1. If the linear structure is circular, the complexity will become 2.

For a double linked list (Figure 2), the number of links is double as for the linear structure but the number of nodes is the same.

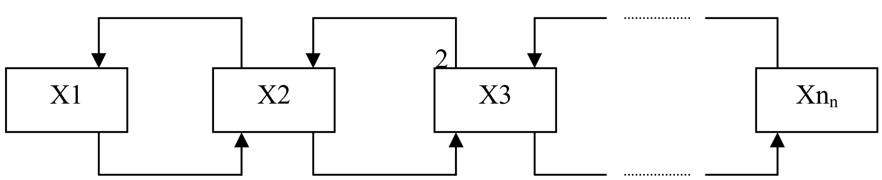

Figure 2 – Linear WEB structure with double links

The complexity of such a structure can be computes as following:

$$C = 2(n_n - 1) - n_n + 2 = n_n$$

The Figure 3 shows a perfectly balanced binary tree with k levels.

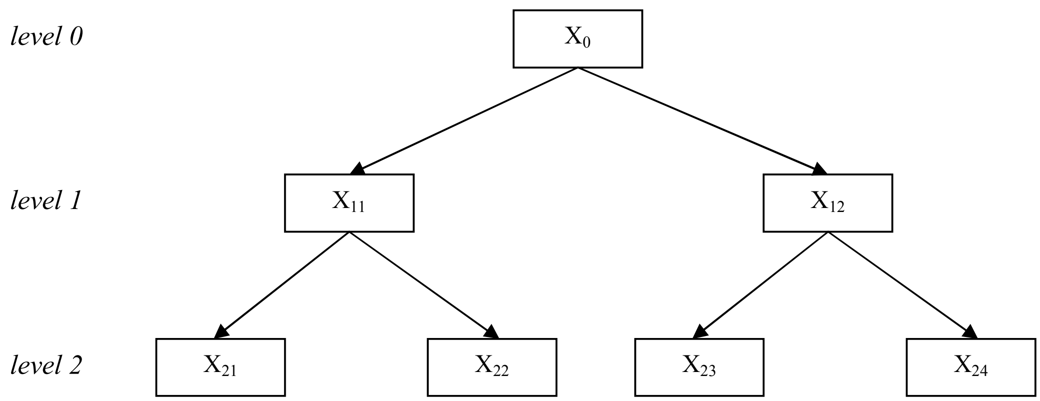

Figure 3 – perfectly balanced binary tree WEB structure

The complexity of such a structure will be:

$$C = (2^{k+1} - 2) - (2^{k+1} - 1) + 2 = 1$$

where

$n_n = 2^{k+1} - 1$ and

$n_a = n_n - 1$

The WEB structures contain many links, the most complex situation is the case when each node refers all the other. In such a structure, if the number of nodes is equal with n_n, there will be $n_n^2 - 1$ links and the maximum *McCabe* complexity will become

$$C = (n_n^2 - n_n) - n_n + 2 = n_n^2 - 2n_n + 2$$

2.2 Complexity indicators

In order to estimate the effort needed to create the links of the application, a complexity indicator can be used:

$$I_c = \frac{C}{C_{\max}} = \frac{C}{n_n^2 - 2n_n + 2}$$

The complexity of the structure typology level can be obtained by comparing the complexity of a specific WEB structure (C_c) with the complexity of the corresponding complete structure (C_p). The indicator of the relative complexity can be obtained in the following way:

$$I_r = \frac{\min\{C_p, C_c\}}{\max\{C_p, C_c\}}$$

The indicator will be equal with 1 for any linear WEB structures because the complexities (specific and complete) will be equal each other.

The next figure (Figure 4) shows a double linked list from which two arches are missing.

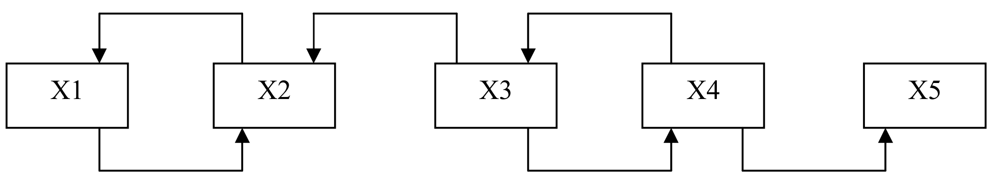

Figure 4 – Incomplete double linked list

The complexity of the complete double linked list is equal with n_n so the complexity of the incomplete list could be obtained in the following way:

$$C_p = 2(n_n - 1) – 2 – n_n + 2 = n_n – 2$$

The relative complexity indicator (I_r) becomes:

$$I_r = \frac{\min\{C_p, C_c\}}{\max\{C_p, C_c\}} = \frac{\min\{n_n - 2, n_n\}}{\max\{n_n - 2, n_n\}} = \frac{n_n - 2}{n_n}$$

We can clearly notice the complexity of a WEB application is changing when some modifications are performed over the initial structure.

The next figure (Figure 5) shows an incomplete binary tree organized on 3 levels.

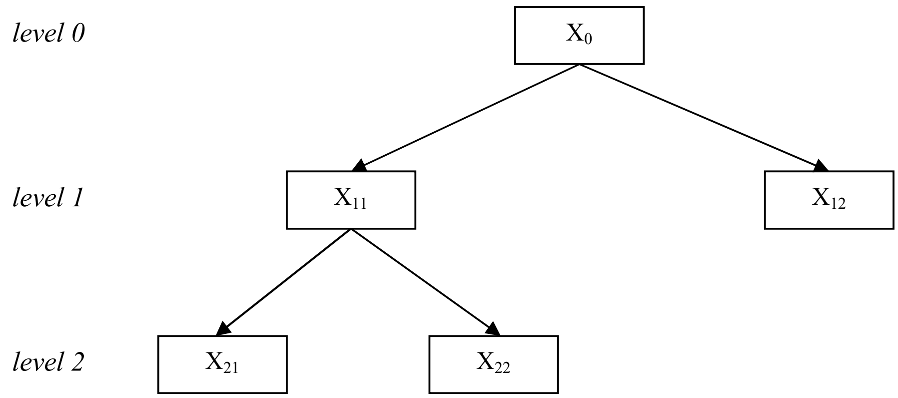

Figure 5 – Incomplete binary tree

This structure will be compared with the complete binary tree with 3 levels (Figure 3).

The complexities of these two data structures are:

$$C_p = (n_n - 1) - n_n + 2 = 1$$
$$C_c = (n_n - 1) - n_n + 2 = 1$$

And the relative complexity indicator value will become

$$I_r = \frac{\min\{C_p, C_c\}}{\max\{C_p, C_c\}} = \frac{\min\{1,1\}}{\max\{1,1\}} = \frac{1}{1} = 1$$

The complexity indicators allow to compare different WEB applications. The homogeneity of the structures of a WEB application generates variations of the complexity indicators.

References

[1] **Ivan, I., Alecu, F.**: *HTML Structures*, ASE Publishing House, 2005.