EDUCATION & COMPETITIVENESS ON LABOR MARKET
DEVELOPING NEW SKILLS WITH VISUALBUILDER

CRISTINA CHELARU[1]

Abstract

In a dynamic world in which "today" differs to a large extend from "yesterday" and in which organizations are living organisms trying to adapt, develop and became more and more competitive each day, the place and role of the employee is also going through dramatic changes. We will try to emphasize the need for adaptability, flexibility and new computer skills within nowadays labor force's environment. As ITC is changing organizational structures it also impacts its members' work behavior. Using a computer is not the same it used to be. Basic computer assisted tasks today may comprise activities based for example on using simple programming languages. We will present VisualBuilder, a tool by which such a task can became really easy for a employee, whether or not he has an IT qualification. We will show it is possible by using visual programming instead of text programming languages. This is all about employee's competitiveness. And it is possible through continuous education**.**

Keywords: labor force, skills, competitiveness, efficiency, visual programming

Introduction

Motto: "In the New Economy, knowledge, rather than natural resources, is the raw material of business".[Center for Regional Studies, USA 2002].

In an economy driven by innovation and knowledge, in marketplaces engaged in intense competition and constant renewal, in a world of tremendous opportunities and risks, in a society facing complex business, political, scientific, technological, health and environmental challenges, and in diverse workplaces and communities that hinge on collaborative relationships and social networking, the ingenuity, agility and skills of people are crucial. Creating an 21st century public education system that prepares students, workers and citizens to triumph in the global skills race is the central economic competitiveness issue every nation aims for the next decade.

Over the last several decades, the industrial economy based on manufacturing has shifted to a service economy driven by information and innovation. All developed countries have made this shift to information products and services. Jobs also have shifted from manufacturing to services, particularly in higher paid information services. "Economic success is

[1] PhD Student, Doctoral School of University „Alexandru Ioan Cuza" of Iasi. E-mail: cristina.chelaru@redprojects.com

increasingly based on the effective utilization of intangible assets, such as knowledge, skills, and innovative potential, as the key resource for competitive advantage."

Companies also have changed how they are organized and the way they do business. Workers have now more responsibility and contribute more to productivity and innovation.

Economic development practitioners throughout the world have struggled with defining the specific skills available in their labor force. Educational coverage among populations living in poverty is inadequate and of poor quality, and this has negative consequences in terms of young people's labor competencies. Significant firm-level productivity increases when associated with changes in business practices, including reengineering, regular employee meetings, self-managed teams, up skilling of workers and computer use by front-line workers. An essential positive relationship in this respect is found between both information sharing and decentralized decision making and a company's innovativeness.

Information and communications technologies (ICT) often have supported changes in organizational structures and practices for communication, information sharing, analysis and simulation of business processes. Early studies of ICT use, showed little productivity gain from technology investments. However, later studies found significant productivity gains associated with specific ways that technology is used. For example, reviews of firm-level studies found that the greatest benefits are realized when ICT investments are accompanied by other organizational changes that ICT use makes possible, such as: new strategies, business processes and practices, and organizational structures.

New, Different Skill Demands at the Work Place

This is a seminal moment in the history for education and competitiveness. The fundamental shifts in the economy demand bold and creative labor work force policies. Formalizing the connection between education and competitiveness with an agenda focused on our century skills – which are widely acknowledged and supported by employers, educators, researchers and thought leaders – is the starting point.

Creating an aligned education system that prepares us to thrive, is the central competitiveness challenge of the next decade and an imperative requirement.

Advanced economies, innovative industries and firms, and high-growth jobs require more educated workers with the ability to respond flexibly to complex problems, communicate effectively, manage information, work in teams and produce new knowledge**.** Generally on labor market, tends to be little coordination between formal education, vocational training and the world of enterprises. More important than the amount of education are the kinds of skills required by the new economy.

A study from Massachusetts Institute of Technology found that, beginning in the 1970s, labor input of routine cognitive and manual tasks in the U.S. economy declined and labor input of non-routine analytic and interactive tasks rose. This finding was particularly pronounced for rapidly computerizing industries. As firms take up technology, computers substitute for workers who perform routine tasks – but they complement workers who perform non-routine problem solving. Repetitive, predictable tasks are readily automated. Hence, computerization of the workplace has raised demand for problem solving and communications tasks, such as responding to discrepancies, improving production processes, and coordinating and managing the activities of others. The net effect is that companies, particularly those with heavy ICT investments – are hiring workers with a higher skill set, particularly expert thinking and complex communications skills.

Advanced economies compete by producing "innovative products and services at the global technology frontier using the most advanced methods" [Porter, Ketels & Delgado, 2007]. High-income countries have a high capacity for innovation—and their strategies are global in scope, which requires a workforce with the skills to "translate business models and offerings to international marketplaces," offer "cross-border perspectives and solutions," and apply "tangible skills such as language proficiency" and "skills that are less tangible, including greater sensitivity to cultural differences, openness to new and different ideas, and the ability to adapt to change.

The level of education is also to increase significantly in the following years in order to assure workers the desire level of competitiveness on labor market. The U.S. Bureau of Labor Statistics for example identifies 271 jobs with high-growth potential over the next 10 years; are likely to increase by 20 percent or more during this period. All of these jobs require at least some college education; most require one or more college degrees. More important than the amount of education are the kinds of skills required by the new economy On Investments and Returns on Education and Skills "As the larger return to education and skill is likely the single greatest source of the long-term increase in inequality, policies that boost the investment in education and training can help reduce inequality while expanding economic opportunity." Ben Bernanke, chairman of the Federal Reserve, 2007.

Employers across cite professionalism or work ethic, oral and written communications, teamwork and collaboration, and critical thinking and problem solving as the most important skills that recently hired graduates from high school towards postsecondary institutions need, according to a nationwide U.S. survey of 400 employers**.** The United States leads the world in several high-growths, ICT-intensive industries, including technology, media and telecommunications. Fueling creativity, innovation and adaptability that are the hallmarks of competitive, high-growth and emerging industries, requires a highly skilled, creative and nimble workforce [Ewing Marion Kauffman Foundation, 2007].

On the subject related to nowadays competitiveness on labor market we might say that all of us as workers, not just an elite few, need new skill skills that will increase our marketability and employability. Some of these news one that seems to be essential could be:

- Thinking critically and making judgments about the barrage of information that comes their way every day—on the Web, in the media, in homes, workplaces and everywhere else. Critical thinking empowers employees to assess the credibility, accuracy and value of information, analyze and evaluate it, make reasoned decisions and take purposeful action.
- Solving complex, multidisciplinary, open-ended problems that all workers, in every kind of workplace, encounter routinely. The challenges workers face don't come in a multiple-choice format and typically don't have a single right answer. Nor can they be neatly categorized as "math problems," for example, or passed off to someone at a higher pay grade. Businesses expect employees at all levels to identify problems, think through solutions and alternatives, and explore new options if their approaches don't pan out. Often, this work involves groups of people with different knowledge and skills who, collectively, add value to their organizations.
- Creativity and entrepreneurial thinking – a skill set highly associated with job creation Many of the fastest-growing jobs and emerging industries rely on workers' creative capacity – the ability to think unconventionally, question the herd, imagine new scenarios and produce astonishing work. Likewise, Americans can create jobs for themselves and others with an entrepreneurial mindset – the ability to recognize and act on opportunities and the willingness to embrace risk and responsibility, for example.
- Making innovative use of knowledge, information and opportunities to create new services, processes and products. The global marketplace rewards organizations that rapidly and routinely find better ways of doing things. Companies want workers who can contribute in this environment.

Computer programming and programming languages as specific skills

Motto: "Programs must be written for people to read, and only incidentally for machines to execute"[Abelson & Sussman, SICP]

The good news is that the human being has an amazing capacity of remodeling, representing and solving various problems. This is possible by reconsidering and utilizing the language concept, not only for communication but also for knowledge. It is said that in our lifetime, we will change from 4 to 5 different types of jobs, doing different sorts of activities at our workplace. So no wonder if at some point in our professional careers we will be undertaking a task or work using things we would have never imagined before. One example could be the situation in which an end user or a worker from outside an IT department would have to use simple notions of programming as part of his work' tasks. Computer languages of the future will be more concerned with goals and less with procedures specified by the programmer. Languages are instruments of thinking. They are tools used for processing. Science itself represents a set of methods and cognitive instruments.

What is computer programming? Simply stated, is really just about solving problems. It turns out that a large number of the problems we encounter in the real world are really just special cases of a more general problem. Luckily for us, many of these problems have been studied by computer scientists for a very long time, sometimes leading to probably unbeatable solutions, or sometimes solutions which are "good enough" for every day needs. In short, learning a language gives you skills, but teaching data structures and algorithms shows you how to use these skills wisely.

Computer programming is the craft of writing useful, maintainable, and extensible instructions which can be interpreted by a computing system to perform a meaningful task.

Computer programming is one part of a much larger discipline known as "software engineering" which includes several different aspects of making software including design, construction and quality control. The current definition of software engineering is still being debated by practitioners today as they struggle to come up with ways to produce software that is "cheaper, bigger, quicker".

Textual programming languages involve series of imperative commands sequentially executed. Opposed to them, visual programming languages is like an assembly line in which workers do their assigned tasks as they are required.

Visual Programming Concept was developed on some very simple motivations:1) Many people think in pictures. 2) Textual programming languages have proven to be difficult for many people to learn to use effectively. 3) Some applications are very well suited to graphical development approaches. Visual programming is basically the use of visual expressions in the process of programming. They might be: graphical interfaces for textual programming languages, graphical presentation of the behavior or structure of a program or syntax of new visual programming languages. Definitions of VPL:1). A programming language that uses a visual representation. 2) A visual language that manipulates visual information or supports visual interaction, or allows programming with visual expressions [Golin, 90]. 3) Any system where the user writes a program using two or more dimensions [Myers, 90]

The advantages of using a visual programming language instead of the text one are clear, but we will state only a couple of the most significant ones: fewer programming concepts; concreteness; explicit depiction of relationships; immediate visual feedback; not necessary to visualize a program in a sequential manner; eliminating an intermediate step in the process of creating a program; less emphasis on syntax; navigable program structure executable; partially specified programs; integration of pictorial clues

VisualBuilder - an exemple of developing new skills

Motto: “If there is ever a science of programming language design, it will probably consist largely of matching languages to the design methods they support” [Robert Floyd, 1979, p. 455]

Fig 1. *VisualBuilder’s logo*

Teaching all students to think and to be curious is much more than a technical problem for which educators, alone, are accountable. And more professional development for teachers and better textbooks and tests, though necessary, are insufficient as solutions. The same goes with employees, their skills, competences and work place requirements.

Are visual programming languages the way to go?

Reuse-in-the-small (libraries of subroutines) began nearly 50 years ago and is a well-solved problem. Reuse-in-the-large (components) remains a mostly unsolved problem, even though everyone agrees it is important and desirable.

We believe that visual programming will catch on in time and will provide a quantum leap in productivity. Just as assembly programming still exists today but is related to a very small, though important, niche, so too imperative text-based programming may find itself off the main track in the future, giving way to sleek and efficient visual systems that will allow many more people to tell the computer what they want done by plugging together components from a library, or pulling up a template and altering it visually on screen. Today's powerful personal computers, advanced graphical operating systems, and visually oriented tools make it possible to deploy a true visual programming language.

Why? Because people think in images. Many applications can be developed more effectively using graphical development approaches. And not finally because visual programming is more suited to development of a variety of concurrent or distributed processing scenarios.

What is VisualBuilder?

VisualBuilder is a revolutionary product, a visual programming language and development environment. VisualBuilder literally allows you to draw your application, not code it. You drag a component to the canvas, and wire it to the other components in the diagram. Then you hit the run button. No compile, no link, none of that. Just Run.

Who is it for?

Anyone who needs to solve a problem fast. VisualBuilder is for professionals that aren't programmers, but need to throw a script together now and then to get their job done. It will also impress professional programmers with its power and capabilities. System managers and IS pros will appreciate its easy reuse that makes the second and third programs faster and easier as you build up your own library of visual components. VisualBuilder is a revolutionary new programming tool. Unlike many tools that have been called “visual” but still require conventional programming, VisualBuilder is fully and truly visual – you don’t need to “write code” to create programs. This technological breakthrough allows both programmers and non-programmers to construct computer applications quickly. With VisualBuilder, you create applications simply by drawing pictures. Your mouse does most of the work. Main steps: 1.Locate each component you want to use in the palette.2. Drag each component out of the palette and drop it onto a canvas.3. Link the components. The VisualBuilder palette includes over 200 assembled components you can use immediately to build programs.

Why use VisualBuilder?

The conventional approach to programming black boxes is to write subroutines, or functions that have well defined input and output. To connect a set of related procedures together, the programmer would have to write a main program to call the relevant functions in the desired order and make sure that at each stage the data is passed to the next function correctly. This code would then have to be compiled, linked with the other functions and debugged before it could be tested.

Consequently, even though the result may not be satisfactory, the programmer would have to put a reasonable amount of effort in writing the program. For example, he/she would have to make sure that the correct number of parameters were being sent to the appropriate functions and each parameter contained correct type and dimensionality of data. This process is extremely time consuming as one mistake along the process chain can lead to unexpected results and even program crashes. This can be avoided by using a robust standard that defines the input and output more rigorously. VisualBuilder takes exactly this approach to linking a set of components.

Each component within VisualBuilder only needs to be written and compiled once. When the component is written, the programmer specifies the type of information that the component can receive and the type of information that it can output. Thereafter, when the end-user wants to connect several tools together he/she simply connects the components together by dragging a cable between them with the mouse. VisualBuilder then checks to see if the connection is valid (i.e. the type of information being sent is receivable by the receiving unit). If it is valid, then the connection takes place, otherwise a message is displayed saying why the connection was not possible.

Once the programmer connects the desired topology of components he/she hits the start button to see the result. Furthermore, since the data types that are passed between the components contain the parameters associated with the particular data (e.g. a sample set contains the sample data within an array, the number of samples and the sample rate) then each component knows exactly how to deal with the data object whatever it contains. It is therefore virtually impossible to crash a VisualBuilder network.

In conclusion it is simply to lear and use, it has a platform independent development and deployment (developed in JawaA, written once, run everywhere, it can be used on mobile phones, desktops or on servers) it is very extensible (it can integrate existing or new components or it can be extended to target new platforms. It can be an educational tool for beginners not only on companies but also in schools.

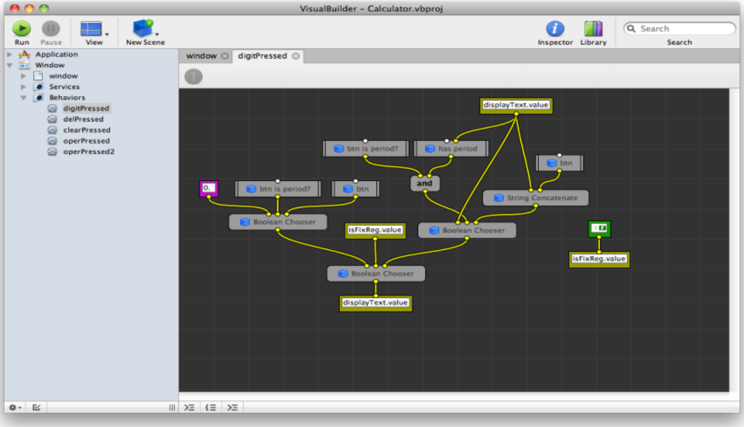

Figure 2. *A simple calculator application – diagram editor*

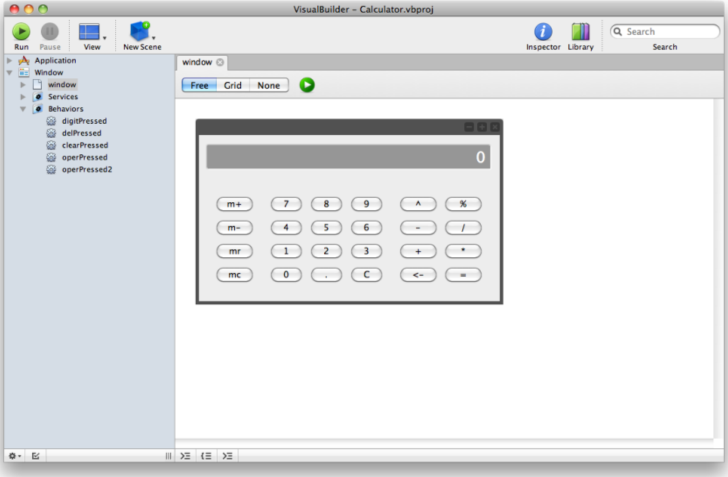

Figure 3. *A simple calculator application - user-interface editor view*

VisualBuilder will be available on 3 editions: Professional, Starter and Enterprise. The Professional Edition will be distributed free of charge trough universities and other educational institutions.

Services

Services are user-generated components. Services perform operations. For example, a service might read data from a database, update data in a database, or execute a web service. A service may also call Java or JavaScript code. Services usually require input parameters and provide output results.

The services available in VisualBuilder include:

- Database services – database services provide create-read-update-delete operations on a database. They are created by the user by importing a database schema.
- Query services - query services provide customized queries on a database.

Web services - execute SOAP, REST, RSS, or ATOM web services. They are created by importing a web service description or a RSS feed.

Java services - execute Java code. They are created by importing external Java libraries.

- JavaScript services - executes JavaScript code to perform client operations.

What types of applications can be created?

VisualBuilder can generate the following types of applications:

- Rich-client applications – used on desktop computers

Applications interacting with Microsoft Office Applications. The ability to read type libraries gives you access to any application that supports a Microsoft Object Model.

- Server applications – used on server computers
- Web applications – used on server computers in the cloud
- Enterprise applications – by wiring together web services with built-in components

VisualBuilder can generate other types of applications by adding specialized modules that extend the functionality of the product.

Conclusions

As new skills are starting to be required on our work environment, we have the VisuaBuilder's example of a programming language, aimed to ease everyone's work and to enable better workflows within organizations. Not only a tool for our work tasks but also a learning platform that will allow us to understand, learn and utilize the concepts of Object Oriented Programming and also Component Oriented Programming.

Acknowlegments

The present study was possible due to the support of RED Management Capital and Solusis System SRL as they provided their own research and conclusions on the visual programming languages and allowed the presentation of the VisualBuilder programming language.

REFERENCES

www. virtuallearning.ning.com
www.ittc.ku.edu/~rsass/rcreading/johnston04.pdf
www.unige.ch/eao/www/Visual/Visual.Programming.biblio.html
www.cs.washington.edu/homes/jpower/vpl/vpl_home.html
http://www.wi.leidenuniv.nl/CS/SEIS/vislang/VLcourse.html
http://www-lsi.upc.es/~rbaeza/sem/vp/todo.html
http://www.cs.berkeley.edu/~maratb/cs263/
http://www.pitt.edu/~korfhage/is2650.html
http://www.taylor-design.com/DataLink/Home.htm
http://hudson.idt.net/~jvn/ch2/ch2.htm
http://www.ntt.ca/prograph.html
http://www.cy-soft.com/prographe.html
http://www.mactech.com/articles/mactech/Vol.10/10.11/PrographCPXTutorial
Partnership for 21st Century Skills, The Conference Board, Corporate Voices for Working Families & the Society for Human Resource Professionals, 2006.
Vlada, M., 2005, Conferinta Nationala de Invatamint Virtual, CNIV.
Whitley, N.,1997,Visual Programming An Outlook for the Academia and Industry, pp.180-208.