A test automation architecture is the same as a software architecture but built in the language of the test tool. But what is a software architecture?

What is an architecture?

“The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them” –Software Architecture in Practice, Bass et al. (Addison-Wesley 1997)

So, a software architecture is the structure of the source code and the relationship between the components. In civil engineering, the architecture is the overall structure of the building. The architecture provides the space for the functional use of the building and ultimately is the major factor affecting how successful it is at fulfilling its purpose.

The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.

A test automation architecture is the same as a software architecture but built in the language of the test tool. But what is a software architecture?

What is an architecture?

“The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them” –Software Architecture in Practice, Bass et al. (Addison-Wesley 1997)

So, a software architecture is the structure of the source code and the relationship between the components. In civil engineering, the architecture is the overall structure of the building. The architecture provides the space for the functional use of the building and ultimately is the major factor affecting how successful it is at fulfilling its purpose.

The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.

A test automation architecture is the same as a software architecture but built in the language of the test tool. But what is a software architecture?

What is an architecture?

“The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them” –Software Architecture in Practice, Bass et al. (Addison-Wesley 1997)

So, a software architecture is the structure of the source code and the relationship between the components. In civil engineering, the architecture is the overall structure of the building. The architecture provides the space for the functional use of the building and ultimately is the major factor affecting how successful it is at fulfilling its purpose.

The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.

A test automation architecture is the same as a software architecture but built in the language of the test tool. But what is a software architecture?

What is an architecture?

“The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them” –Software Architecture in Practice, Bass et al. (Addison-Wesley 1997)

So, a software architecture is the structure of the source code and the relationship between the components. In civil engineering, the architecture is the overall structure of the building. The architecture provides the space for the functional use of the building and ultimately is the major factor affecting how successful it is at fulfilling its purpose.

The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.

A test automation architecture is the same as a software architecture but built in the language of the test tool. But what is a software architecture?

What is an architecture?

“The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them” –Software Architecture in Practice, Bass et al. (Addison-Wesley 1997)

So, a software architecture is the structure of the source code and the relationship between the components. In civil engineering, the architecture is the overall structure of the building. The architecture provides the space for the functional use of the building and ultimately is the major factor affecting how successful it is at fulfilling its purpose.

The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.

A test automation architecture is the same as a software architecture but built in the language of the test tool. But what is a software architecture?

What is an architecture?

“The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them” –Software Architecture in Practice, Bass et al. (Addison-Wesley 1997)

So, a software architecture is the structure of the source code and the relationship between the components. In civil engineering, the architecture is the overall structure of the building. The architecture provides the space for the functional use of the building and ultimately is the major factor affecting how successful it is at fulfilling its purpose.

The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.

A test automation architecture is the same as a software architecture but built in the language of the test tool. But what is a software architecture?

What is an architecture?

“The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them” –Software Architecture in Practice, Bass et al. (Addison-Wesley 1997)

So, a software architecture is the structure of the source code and the relationship between the components. In civil engineering, the architecture is the overall structure of the building. The architecture provides the space for the functional use of the building and ultimately is the major factor affecting how successful it is at fulfilling its purpose.

The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.
required to add a stock item. These functions are what are known as *wrappers*.

**Automation wrappers**

Wrapper is an OO term that means some software that is used to interface with an object. In programming terms you call the wrapper if you wish to use the object. You can’t call the object directly.

In test automation, wrappers are programs or functions written in the language of the test tool which perform discreet automation tasks as instructed by the test data and actions. They provide the interface between the test and the system under test’s user interface. In our previous example there are four wrappers – Stock_Item_Add, Loc_Add, Supplier_Add and Stock_Loc_Add. These are business level wrappers – they ‘wrap’ business functions and were written by the test programmer (scripter). You will be able to see that the example in figure 2 is similar to the data driven approach but the first column in the test data of each line tells the automation which wrapper to call.

There are two distinct types of advanced architecture. Figure 2 shows a *business object level architecture*. In this type of automation architecture, one automation function or program is written in the test tool for each type of business task. These functions are the wrappers for the business tasks.

Usually a functional decomposition of the system is the first step in building this type of architecture. In a functional decomposition, the basic business functions of the system are defined. Then the wrappers for each business task are programmed (scripted) and the data format for each task is created.

Test data in business object level architectures is at the business language level and therefore understandable by end users, which is a great advantage.

**Screen/window level architectures**

In this architecture there is one test program that deals with each screen/window in the system—it acts as a wrapper for that screen/window. It handles all of the input and output (getting expected results) for its window or screen. See figure 3 for an example of the data. Again lines with a hash in the left hand side are comments used to show the test analyst the format of the data for that screen/window. The other lines are the test data which is passed to the automation wrappers for that particular screen. The first column of the test data represents the screen that the data refers to and thus the format of the line is dependent upon what the screen name is in column one. Also note that, as this a test of a GUI system, for each user interface object there is an action and a data field. Thus the test analyst can specify actions like ‘Check’ that the object contains the data equal to ‘Smith’ or even ‘CheckEnabled’.

One of the biggest advantages of screen/window level architectures is that all of the user interface objects can be made available to the test analyst in Excel and thus the analyst has complete control over what the navigation and actions should be, rather than being dependent upon what the test automation programmer (scripter) has put into the wrapper—as with business object level architectures.

**Next issue: New ways to create automation code**