

It is interesting to see that (subject, of course, to the overriding disclaimer in the opening preamble) the Department and the ARB have been signed up to this non-binding “Framework” with the inclusion of references to: monitoring ARB’s activities on a regular basis, and informing ARB of relevant government policy (4.3); the Sustainable Buildings Division as the main source of advice to the responsible Minister in respect of ARB (4.3); the responsibility of the Registrar for safeguarding “public funds” (6.1) and “consulting CLG on matters of vision and strategy” (6.2); the role of the Board’s chairperson to lead in “representing the views of the Board to the architect’s profession and to the general public” (7.4); and so on.

All this may be helpful in boosting self-esteem and point-scoring for salary and fee scales in the usual departmental way, but if there is any reliable evidence to show that it helps architects eligible for Part 1 registration to earn a livelihood, let it be published to give the general public (which includes a considerable network of friends, relations and public spirited persons with an interest in the advancement of architecture) some assurance that this body is capable of providing (or in current jargon, “delivering”) a useful service, as distinct from an outstation of Whitehall style government.

And now – Why is “Framework” in the title? A simple definition (given in Collins Paperback Dictionary) is: “a particular set of beliefs, ideas or rules referred to in order to solve a problem”.

But there is another usage which seems surprisingly relevant, shown by this extract from Wikipedia’s “[Software Framework – 4: Architecture](#)”: “...software frameworks consist of *frozen spots* and *hot spots*. *Frozen spots* define the overall architecture of a software system, that is to say its basic components and the relationships between them. These remain unchanged (frozen) in any instantiation of the application framework. *Hot spots* represent those parts where the programmers using the framework add their own code to add the functionality specific to their own project. Software frameworks define the places in the architecture where application programmers may make adaptations for specific functionality—the hot spots. In an object-oriented environment, a framework consists of abstract and concrete classes. Instantiation of such a framework consists of composing and subclassing the existing classes. When developing a concrete software system with a software framework, developers utilize the hot spots according to the specific needs and requirements of the system. Software frameworks rely on the Hollywood Principle: “Don’t call us, we’ll call you.” This means that the user-defined classes (for example, new subclasses), receive messages from the predefined framework classes. Developers usually handle this by implementing superclass abstract methods.”

*Ed.* Another contributor, on reading this Note, recalls a stanza from Matthew Prior’s Ode: On Exodus III.14, hoping that the new Government will not be tempted to substitute for legislation where none is required:

*Man does with dangerous curiosity  
These unfathom’d wonders try:  
With fancied rules and arbitrary laws  
Matter and motion he restrains:  
And studied lines and fictitious circles draws:  
Then with imagined sovereignty  
Lord of his new hypothesis he reigns.  
He reigns; how long? till some usurper rise!  
And he, too, mighty thoughtful, mighty wise,  
Studies new lines, and other circles feigns.  
From this last toil again what knowledge flows?  
Just as much, perhaps, as shows  
That all his predecessor’s rules  
Were empty cant, all jargon of the schools:  
That he on t’other’s ruin rears his throne,  
And shows his friend’s mistake, and thence confirms his own.*