

第8回ソフトウェア翻訳士認定試験

<一次試験> 4月6(日) 10:00~15:00

問題1・2の両方について解答のこと。選択ではありません。

<問題1> 下線部を訳して提出してください。

Good design may reflect the zeitgeist, but designers also try to lead the charge in advancing the culture in which we live. So the advent of 'web 2.0', which is increasingly permitting users to design the look and feel of their own content --- is challenging the role of the interface designer like never before.

Blogs and social networks are having a massive impact on the type of design people expect. A prime example is the increasing popularity of right-hand navigation, due to the various template-driven blogging tools available. Web users have grown accustomed to seeing navigation on the right-hand side of the screen, and as 1.024 x 768 resolution monitors become the norm, designers are shifting their navigation to the right as well. Social networks such as MySpace have arguably made users accustomed to busy, ugly websites. On the other hand, the personal nature of social networking seems to be contributing to warmer, more organic design. More and more, designers are using textured backgrounds, moving away from white space and using less austere fonts.

Consolidate your vision

Today's designers need to think beyond 2D and create in space (3D) and time (4D). We also need to consider how the front end works. When you hand over your design to the team that will code the website, make sure they fully appreciate your vision of how it should work, especially with regard to the transitions and visible behaviours newer technologies support (fades, zooms, blurs, and so on). Squeeze a bit more time out of your project manager so you can design storyboards for your Flash people, rather than static Photoshop screens. Work as a team, not in isolation.

If you're creating a pop-up slideshow, set the colour of that alpha-transparency overlay, don't just accept the semi-opaque black. Alternatively, tell the interface developer that you want a shorter transition back to the web page when the slideshow overlay is closed. If you've designed a site that's edgy and punchy you don't want the effect ruined by a serene two-second fade-up/down.

As designers, we can only maintain the integrity of our work by insisting on aesthetically consistent behaviours between static and interactive elements. To please our potential audience we have to keep progressive enhancement and graceful degradation at the forefront of our minds. Don't forget that some users have JavaScript turned off, so you need to be aware of how the design will degrade. That cool pop-up slideshow with the alpha-transparency behind it should be reduced to an image library of thumbnails with the option to click through to a page containing a full-size image --- you need to design that too.

<問題 2> 全文を訳して提出してください。

Querying Data from Distributed Heterogeneous Database Systems through Web Services

Web services have become an important research topic in recent years, mostly because it promises interoperability between loosely-coupled applications across distributed heterogeneous computing environments. The provision of data and application logic can be exposed by a system through Web services. Some of the most common Web services found on the Internet simply provide data such as XMethod's stock quote service, Google and Amazon's Web service APIs, and eBay's price watcher. These websites and many other web applications like them are driven by database systems, and the Web services that they provide can be viewed as an interface to transparently access these database systems.

A typical Java implementation of a Database Web Service Provider is portrayed in Figure 1. At the forefront of the service architecture is the SOAP engine for handling incoming SOAP request messages. Apache Axis can be used to demarshal SOAP messages, and execute corresponding methods in Java classes in the middleware layer that access a database system using a JDBC driver. After executing an SQL statement through JDBC, the data returned in a JDBC ResultSet object can be passed back to AXIS from the middleware application, to where the data is marshalled into a SOAP response message and returned back to the Web service requestor.

Conversely, the notion of pushing data to a Web service for processing is also applicable under the database Web service requestor model. By decentralizing the procedural logic, intensive computations can take place on a dedicated server provided by a Web service, therefore distributing the computational load amongst many computers. The resulting computations can then be retrieved and inserted into the calling database.

出展 : <http://ausweb.scu.edu.au/aw04/papers/refereed/carroll/paper.html>