翻訳トライアル実践講座　第８回

＜問題１＞　全文を訳して提出してください。

A New Work Style Combining Social, Mobile, Analytics and Cloud

Today's customers and employees, particularly digital natives, are expecting a new style of commerce, content and collaboration that's social-, mobile-, analytics-, and cloud-enabled. They're looking for the same anytime, anywhere, and any-device convenience that they're familiar with in their personal lives through applications from companies such as Amazon and Facebook. In terms of device usage, the "mobile elite" in the workforce currently utilize three or more personal devices for work and this number will only increase as wearable devices such glasses and watches add to end user options.

In the SMAC era, the next generation of business applications needs to embrace this same approach to enhance the end user experience, and maximize convenience and productivity, as SMAC-enabled architectures become the preferred application paradigm and means of interaction.

As part of this new architecture, IT departments will need a capability equivalent to a "user experience engine" to provide the SMAC technology integration, management and personalization layer providing a contextually-relevant experience to end users and supporting their new work style. It's not just about mobile device management, mobile application management, and social platforms all in silos, but about integrating these capabilities into a seamless user experience.

＜問題 2＞　全文を訳して提出してください。

May Your Application R.I.P – Part 1

Didn't really mean that (R.I.P) way, Indeed. The objective behind this series of blogs is to represent some of the best practices, the how-tos and the whats/whys in relation with REST based integration of applications.

Before we shall talk on any related aspects of REST and how to design great RESTful services and, why go for RESTful design in the very first place, lets try and understand a little bit on what is REST and, "Resource Oriented Architecture (ROA)".

What is REST?

REST is an architectural style for distributed hypermedia systems like WWW. REST stands for REpresentational State Transfer. The term first originated in 2000 doctoral dissertation of Dr. Roy Fielding, one of the principal author of HTTP protocol. System or architecture or design that follow these REST principles can be termed as "RESTful". The key to REST architectural style are resources which are defined later in this blog. In RESTful architecture and design, an application can be split into resources which can be accessed using a URI. These resources in its current state can be accessed in form of information which is termed as representations. See the diagram below that represents a resource with multiple representations and different operations that can be applied on the resource. In other words, a representation is information about the state of the resource. For example, a book as a resource can have multiple representations. One can be a representation in form of name of the book, its author and pricing details. Other can be the content of the book. And, the recommended way to work with different representations of the same resource is to have different URIs.

Following is what takes place in REST (Representational State Transfer):
1.Client access the resource using URL.
2.Client retrieves the representation (information) of the resource.
3.Once retrieved, client is put into a new state.

4.URLs in the resource data (existing representation) can be used to retrieve new representations and transfer the client to a new state.

Thus, in brief, the state of client changes (transfer from one state to another) based on retrieval of resources' representations – "Representational State Transfer"

以上