When to Use Custom, Proprietary, Open-Source or Community Source Software in the Cloud

Massimiliano Claps, Andrea Di Maio

Cloud computing has created new delivery models that governments need to consider when acquiring software, such as using infrastructure as a service to test new applications or using software as a service for enterprise and domain-specific applications. This research provides CIOs, enterprise architects, and application and sourcing managers an overview of use cases where cloud-computing offerings could be used to acquire, develop and deploy software.

Key Findings

- Government IT executives that need to replace or acquire new software must consider two axes of analysis: acquisition models and delivery models.
- Acquisition models are not limited to the traditional build-versus-buy paradigm, but they include custom-built, transfer, community source, open source and proprietary off-the-shelf.
- Infrastructure utility services, and in particular cloud services, introduce the additional variable of deployment and delivery models, which create a range of new options to acquire software. However, not all acquisition models can be used in the cloud.

Recommendations

Government CIOs, enterprise architects and sourcing managers acquiring new software should:

- Evaluate what acquisition model best matches the organization's functional requirements, technical architecture, sourcing attitude, availability of skills, potential for cooperation, and appetite for risks, as well as the maturity of external implementation and maintenance services.
- Evaluate the suitability of cloud-based services for specific uses cases, such as testing a custom-built application on a cloud infrastructure service or delivering application access for a community source application.
- Carefully time the adoption of cloud services, based on the product and service market maturity for open-source and commercial off-the-shelf (COTS) domain-specific systems.
ANALYSIS

This research is part of a series of three notes that includes a taxonomy of cloud services for governments and criteria for evaluating cloud offerings.

In the past several years, the increased maturity of off-premises application hosting and, more recently, the advent of infrastructure utility and cloud-computing offerings (see "Comparing Cloud Computing and Infrastructure Utility") have created new options for how government agencies can source their software applications.

Traditionally, system integrators would be involved in helping with development and deployment (such as design, implementation, testing and configuration); some system integrators and outsourcers would provide on-site application management or host the application in a dedicated server in a remote data center. Today, several government clients are exploring ways to leverage cloud-based services to support their application needs (see Note 1).

In order to evaluate the impact of cloud computing on software application sourcing, it is important to consider both the acquisition and the cloud delivery models.

Acquisition Models

Acquisition models include (see Note 2):

- **Custom.** A software application is developed by internal resources, by a contractor, or by a combination thereof, according to specific requirements.

- **Transfer or reuse.** A custom software application or its components (for example, requirements, algorithms, functions, business rules, architecture, source code, test cases, input data and scripts) are reused by a different government organization than the one that was originally responsible for its acquisition (see Note 3).

- **Community source.** This model aims to coordinate the work of different user IT organizations sharing the same purpose and a set of common requirements. Participating organizations leverage internal application and development resources, with the expectations that others will do the same, to contribute to an outcome rather than take responsibility for it.

- **Open source.** The application is available through an open-source license and can be supported by internal or external resources.

- **Proprietary commercial off-the-shelf.** The application is licensed by an independent software vendor (ISV). It can undergo configuration and personalization that can be performed by the ISV, by a system integrator or by internal resources.

Cloud Delivery Models

Cloud delivery models include (see "Helping Governments Cut Through the Definitional Cloud"):  

- System infrastructure services (corresponding to infrastructure as a service [IaaS] in the U.S. National Institute of Standards and Technology [NIST] definition)

- Application infrastructure services (corresponding to platform as a service [PaaS] in the NIST definition)
• Application services (corresponding to software as a service [SaaS] in the NIST definition)

Software Acquisition Models in the Cloud

Although conversations about applications and cloud computing in government tend to focus on software as a service, cloud computing can be an effective delivery model in several respects. Table 1 illustrates the most likely use cases at the intersection between software acquisition models and cloud delivery models. Specifics are detailed as follows:

• **Custom.** The characteristics of cloud computing that assume shared resources do not fit the need of custom-developed applications. Nonetheless, the developers could use computing power and tools from cloud service providers to develop and test custom application software.

• **Transfer.** Similarly, a solution that is developed by an agency and then transferred to other agencies is unlikely to be delivered through a cloud service as a fully standard application. In fact, the transfer can consist of only some parts of the source code, so that it would not be a real multitenant system. Furthermore, the limited number of agencies that usually acquire a transfer solution prevents the achievement of the economies of scale that enable scalability and elasticity, which are proper for cloud computing. The only exception could be large agencies that already offer shared services to smaller bureaus and commissions, which could rearchitect applications to deliver them through a private cloud platform.

• **Community source.** This software, because it is natively developed by several organizations, could not only leverage infrastructure and platform services, but also have the developer community set up an application service accessible by all users. The critical mass of the user community and its stability (frequent increases and decreases in the number of users could challenge the economics of a cloud service with finite elasticity) will drive decisions about the opportunity to invest in setting up a cloud service.

• **Open source.** This software has a more standardized architecture across vast groups of users compared to community source. Therefore, it is more likely to provide an incentive to government and public cloud service providers to offer it as an application service; the maturity of those services will depend on how quickly open-source software (OSS) communities will rearchitect their solutions for the cloud and on how quickly service vendors create cloud-based service offerings for OSS products, beyond the traditional implementation capabilities that they offer already.

• **Commercial off-the-shelf.** These software products have the most standardized architecture across vast groups of users. Therefore, they are the most likely to provide an incentive to government and public cloud service providers to offer them as cloud application services; the maturity of those services will depend on how quickly software product vendors will invest in rearchitecting their solutions and on how product and service vendors partner to redefine business models from the traditional resale of license and maintenance to usage-based fees.
Table 1. Use Cases for Cloud Computing

<table>
<thead>
<tr>
<th>System Infrastructure Services</th>
<th>Application Infrastructure Services</th>
<th>Application Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom</td>
<td>The agency programmers could use remote hardware resources to develop, prototype and test the custom application.</td>
<td>The agency programmers could use remote application development workbenches to develop code for the custom application. The agency enterprise architects and/or programmers could use remote application integration services to interface the custom application with other systems.</td>
</tr>
<tr>
<td>Transfer</td>
<td>The agency reusing the software could use remote hardware resources to develop, prototype and test the custom application. The agency that originates the software for reuse could use remote computing resources to establish a repository of source code components.</td>
<td>The agency reusing the software could use remote application development workbenches to adapt the code to their business requirements. The agency reusing the software could use remote application integration services to interface the custom application with other systems.</td>
</tr>
<tr>
<td>Community source</td>
<td>Community programmers could use remote hardware resources to develop, prototype and test the community source code.</td>
<td>Community programmers could use remote application development workbenches to develop the software functionality, as well as develop and deliver APIs.</td>
</tr>
<tr>
<td>Open source</td>
<td>The OSS community could offer (or partner with external service providers) to host the agency's software in the cloud.</td>
<td>The agency could use cloud-based application integration and application service orchestration.</td>
</tr>
<tr>
<td>Proprietary commercial</td>
<td>The software vendor could offer (or partner with external service providers) to host the agency's software in the cloud.</td>
<td>The agency could use cloud-based application integration and application service orchestration.</td>
</tr>
</tbody>
</table>

Source: Gartner (May 2010)
It is important to notice that there are at least two other variables that influence the applicability of cloud services use cases.

The first element that government agencies need to consider when evaluating cloud-based software services, in particular cloud-based application services, is that many business processes are supported by domain-specific software, such as licensing and permitting, tax collection, grant management, welfare payments, and social service planning and provision. As the market for domain-specific, open-source and proprietary off-the-shelf products is still maturing (see "COTS Is Becoming a Viable Option for Tax and Revenue Agencies' Application Modernization" and "Critical Capabilities for Grantor Management Software"), in most of those domains, government agencies should expect that cloud-based domain-specific application services will not be available in the next 12 to 18 months, with only some exceptions. Therefore, government agencies should start to pilot and then implement cloud-based services for horizontal software solutions, such as e-mail, financials and procurement.

The second element that influences the applicability of cloud-based services is at the federal/national government level. The complexity of business and legal requirements, as well as governance models of federal agencies and departments, has constrained the applicability of cloud services at that level of government compared to state and local governments; however, there are a few examples that show increased interest for public and private cloud services at the federal/national government level too.

**RECOMMENDED READING**

"When to Use Custom, Proprietary, Open-Source or Community-Source Software"

"Cloud Computing in Government: Private, Public, Both or None"

"Government in the Cloud: Much More than Computing"

"Cloud Computing Services: A Model for Categorizing and Characterizing Capabilities Delivered From the Cloud"

"Helping Governments Cut Through the Definitional Cloud"

**Note 1**

**Existing Examples of Cloud Services Used by Governments**

Examples of cloud system infrastructure services:

- USA.gov Web portal running on Terremark's Enterprise Cloud platform
- USASpending.gov running on NASA Nebula
- A U.K. government agency (that Gartner recently spoke with) that uses a cloud-based service to test new applications
- NASA cloud storage services through its Nebula platform ([http://nebula.nasa.gov/services/storage](http://nebula.nasa.gov/services/storage)).

Example of cloud application services:

Note 2  
Acquisition Models  

In 2007, Gartner analyzed various software acquisition options that government agencies could use: custom-built, community source, open source and proprietary COTS (see "When to Use Custom, Proprietary, Open-Source or Community-Source Software"). The research argued that beyond functional requirements, technical platform and cost considerations, the acquisition options should be evaluated based on sourcing attitude, availability of skills, maturity of services, risks and potential for cooperation.

Note 3  
Transfer  

This has proved popular in some domain-specific areas where immaturity of COTS applications, or lack of specialized knowledge by system integrators, made both custom-built and COTS options very expensive. For example, in welfare/human services in the U.S., some states acquired solutions from one another for case management systems in the past 10 years. The Australian Capital Territory government developed a solution that enables police and car inspectors to retrieve vehicle data via handheld devices; other governments in the region reused the source code to develop similar systems. More recently, Gartner spoke with a Canadian federal government agency that was negotiating with a provincial government in Quebec to transfer a grant management application. Transfer/reuse of solutions has not experienced widespread adoption because of a number of challenges:

- COTS components that were part of the architecture and were not easy to transfer unless a new license agreement was signed with the vendor
- Turf battles, where many government agencies wanted to "sell" their solutions and few wanted to "buy/reuse"
- Lack of capabilities from the originating agencies to offer documentation, training, after-implementation support and a consistent solution development road map (The acquiring agency quickly found itself with a system that resembled more of a custom-built solution than an off-the-shelf product.)