

Software as a Service Decision Guide and Best Practices

Purpose of this document

Software as a Service (SaaS) is software owned, delivered and managed remotely by one or more providers [Gartner, SaaS Hype Cycle 2009]. SaaS applications are usually:

- Accessed via the Internet
- Multi-tenant
- Paid for on a subscription basis

For more information, see the SaaS Toolkit on the Inweb.

Software as a Service (SaaS) has definite advantages and disadvantages. SaaS applications may suit a given business need based on factors including economics, flexibility, security, control of the data and stability. Firm rules do not exist, and SaaS applications are morphing rapidly as the market evolves. This document provides a set of best practices to guide you when considering a SaaS solution and vendor.

When to consider a SaaS as a solution

The following table describes general factors to take into account when considering a SaaS solution.

Factor	SaaS	On Premises
Cost	SaaS applications may be more expensive over the long term, Gartner Group estimates with a break even versus on premises somewhere between 3 to 5 years, depending on the nature of the application.	On premises applications require upfront investment, but maintenance costs drop. A long-term “set & forget” application that requires minimal upgrades is likely to be less expensive.
Customization	Business needs require minimal customization from the vendor’s standard configuration and functionality.	The application requires extensive customization in order to work properly with a City business process.
Functionality	The functionality the application provides is rapidly evolving – if you want to evolve in that direction. SaaS vendors don’t ask before changing applications.	On premises applications offer more staged matching of functionality to business processes.
Market direction	The vendor space for the type of application is evolving rapidly, so a long-term direction is hard to predict. Note that some market segments lend themselves to SaaS.	Evolution of the application market segment is predictable or complex.

Factor	SaaS	On Premises
Complexity	SaaS favors application areas that are more self contained and web-based like online job application or electronic contract bidding.	On premises favors complex, highly integrated applications or application suites, such as Enterprise Resource Planning applications.
Flexibility	The need exists to scale the number of users or amount of data up or down rapidly. SaaS resources are typically more flexible.	Numbers of users and data are more predictable and stable, allowing for planned expansion or contraction.
Expected longevity of the business need	The business process the application supports is likely to change significantly, or no longer be needed within several years.	Long term business needs.
Funding	Though SaaS contracts are typically shorter term with easier termination, you must budget for the subscription fee for as long as you intend to use the application.	Ongoing costs to maintain the hosting environment and vendor software maintenance costs need to be budgeted for in addition to acquisition costs. On premise often offers future cost flexibility by not renewing software maintenance, or deferring hardware upgrades.
Maintenance	Adequate staff or infrastructure resources are not available to adequately install and support the application in-house.	Adequate staff and infrastructure resources exist in house.
Skills	The application requires skills not found in the City or outside the City's core competence.	Skills are within the City's core competencies.
Data	Accessibility to data can be interrupted by downtime and workarounds can be established to retain data over the long term.	The application is mission critical and the data needs to be reconstructed rapidly via other methods for a workaround should the application be unavailable.
Security	Security offered by the vendor is adequate and verifiable.	The City data has security or retention requirements that are higher than the typical SaaS customer's data.
Downtime	Downtime stated in the vendor contract is adequate.	The business process demands zero application downtime.
Integration	The application is stand alone, or does not require significant integration with other systems.	Significant integration with other applications or vendor platforms is required.

Best practices for SaaS

Following best practices can help increase the likelihood of a successful application implementation. The following are best practices that are important in the SaaS world. While they aren't unique to it, the nature of SaaS increases their importance.

Best Practice: SaaS is a service - don't consider it an acquisition.

SaaS is a subscription, not a purchase. What we are buying is the use of the application and the infrastructure that runs the application, not the application and infrastructure themselves. We don't typically take possession of software or the infrastructure that runs it. We also don't control upgrades or changes to the application, either in terms of functionality or timing. And most importantly, when the subscription stops, the use of the application and access to the data also stops.

Implication – A SaaS application typically allows for minimal or no customization as compared to an internally hosted Commercial Off-The-Shelf (COTS) application. There is also no guarantee any modifications made by the City will be usable in future versions.

- Keep modifications to a SaaS application to a minimum - adjust business processes before trying to customize the application. This reduces the likelihood that work spent aligning business process and the application will be lost in the next version.
- Monitor the vendor's direction for the application and look for areas of concern with respect to projected City functionality.

Questions to ask

- How often does the vendor do releases? Are they scheduled?
- How far in advance of a migration to a new version will users typically have to evaluate the new version and determine needed modifications?
- Do we have any latitude when we are converted to a new version? Is a new version of the application typically run in parallel to the old version during the migration period, or are all users automatically switched at a cutover date?

Best Practice: stay in the mainstream of the market.

SaaS vendors operate through economies of scale by providing functionality that is of value to a high percentage of their current and targeted customer base. Features and functions that aren't used by many customers are at risk of being dropped in future versions due to costs to maintain.

Implication – If you are dependent on functionality the vendor doesn't consider core to its product, you are at higher risk of losing that functionality in future releases. This could have a sudden negative impact on your business' ability to get work done. When selecting a vendor, consider not just if the SaaS solution meets the current requirements, but how your requirements fit into the vendor's planned roadmap for their SaaS offering.

Questions to ask

- Is the City planning on using the application in a way that is significantly different than the way other customers are using it?
- Are other vendors in the application space moving away from providing functionality the City considers important?

Best Practice: Avoid an “Out of sight, out of mind” approach.

An advantage of SaaS is in not needing to build, install, monitor, and support an application and its infrastructure. However, this can lead to a false sense of security, and less than due diligence when selecting a solution because it is assumed the vendor is adequately addressing everything.

Implication – It is still important to understand the architecture, tools, and processes that support the application in order to determine if the application is likely to perform as promised. This needs to be done when the application is selected, and followed up with appropriate contracting to make sure changes don't create problems over the life of the business relationship. Stay actively involved ensuring changes/upgrades won't negatively impact City business processes.

Questions to ask

- Can the vendor provide architecture diagrams or other documentation that can demonstrate the environment has the ability to scale while maintaining adequate performance, is secure, and is sufficiently fault tolerant?
- Does the vendor have other customers that are at least as big as the City in terms of amount of data and number of users? If the City is the biggest user, we might be concerned about future application performance or scalability if the vendor gains additional large customers.
- Does the vendor have documented processes for their internal operations?
- How current are the versions of other third party software that support the application?
- Does the vendor monitor any third party service provider they may use to ensure the third party stays current on appropriate certifications, e.g. Statement on Auditing Standards (SAS) 70 for Data Centers?
- Are provisions included in the contract to make sure that changes are communicated and reviewed?

Best Practice: Remember the Vendor hosts the data and application.

In a SaaS application, we don't host the application and we typically don't store the data. That is done by either the vendor, or by a third party providing hosting services to the vendor. We therefore don't have direct access to our data.

Implication – Understanding which functions and which data is mission critical to the City is important to reduce the chance of interruptions to City services, and to guard against data loss.

Questions to ask

- Do we have the ability to reconstruct data should the vendor cease operations and we are unable to recover our data from them?
- How long can the City be without the data before critical business processes are affected?
- Can we reconstruct the data in a timely manner?

- If we determine the data can't be reconstructed in a timely fashion, can the vendor provide the capability for frequent downloads of our data in a format we have the software to work with (e.g., Excel, etc)?
- Does the vendor understand any retention and/or disclosure requirements that apply to them as a result of storing City data?
- Do you know where the data is physically stored? If it is out of the country, are the privacy and disclosure laws adequate to protect confidential data? If the data is confidential in nature, does the vendor have adequate policies and processes in place to protect the confidentiality of the data?
- If there is a breach in security of the vendors system has a process been defined as to when we be notified. Are there criteria for resolving a security breach?
- Does the vendor host the application and data in a SAS 70 certified data center?

Best Practice: Budget accordingly.

The cost model is different for SaaS than an internally hosted application. SaaS upfront costs are typically lower than an acquisition, though the long-term costs may be higher. With an internally hosted application, you normally have the ability to reduce or eliminate ongoing vendor costs by dropping maintenance. With SaaS, you incur costs for as long as the service is in use.

Implication – Since SaaS is a subscription, it is likely to represent an operating cost rather than a capital expenditure, and ongoing budget impacts need to be considered. The nature and stability of the City's funding source also needs to be considered.

Questions to ask

- Does the business process the service supports have a known lifespan, or is the need ongoing with no foreseeable end?
- Is there a long term commitment to the funding, or is the funding one-time such as a grant?
- Have price increase limits been incorporated into the contract renewal terms?

Best Practice: The Service Level Agreement (SLA) is key.

In SaaS applications, we don't have control over the application or the infrastructure it runs on. That means we don't decide when the system is down or when hardware is upgraded, and we are not able to tune the application to improve performance.

Implication – A Service Level Agreement (SLA) that is negotiated with the vendor is the strongest control we have to ensure that the application is available and performing acceptably, and that any problems are being addressed by the vendor in a timely and appropriate manner.

Questions to ask

- Do we understand the degree to which the various functions of the application are mission critical to the City, and therefore understand what the required service levels for service performance and availability need to be?
- Does the vendor already have defined SLA metrics and targets for both service availability and performance? Do they meet City business requirements?
- Does the vendor have regularly scheduled downtime for system maintenance? Do the availability uptime targets include or exclude scheduled maintenance? Are those metrics and targets stated in the SLA?
- Are processes defined in the SLA for both planned and unplanned outage notification?

Best Practice: SaaS support models differ from internally hosted applications.

The differences in the control of the various aspects of a SaaS solution versus an internally hosted solution often equates to differences in who can best support those various aspects. There are no standards in the SaaS world for what the vendor is responsible for, so support responsibilities vary.

Implication – Work out the division of support responsibilities between the vendor and any internal staff that may be called upon to support parts of the system in advance. Since IT plays a lesser role in a SaaS implementation than an internally hosted implementation, IT may not be involved in vendor discussions, which can lead to gaps in communication. Don't assume anything.

Early on, contact and involve your desktop support person/TSP. Just because this is a SaaS doesn't necessarily mean there is no desktop software component to check and/or install. In addition, there may be required cookies, changes to IE settings, etc., that your desktop person can help coordinate/implement and test.

As soon as you know your go-live date (even a tentative one), get your application into the Citywide Change Mgt process. The Change Mgt committee can help identify the types of issues listed above, as well as any other dept-centric issues that may arise.

Questions to ask

- Have the various types of support needed for the system been explicitly defined, and have roles and responsibilities been divided between the vendor and the City? Has the appropriate City staff been made aware of their responsibilities?
- Has the vendor defined their problem escalation process, including defining problem severity levels and expected response time and outcomes for each level?
- Has the vendor provided a support staff contact list?
- How will trouble tickets and escalations of problems be handled? Are City Service Desk and Network roles defined versus vendor roles?

Best Practice: Choose your vendor carefully.

While this can be said for any acquisition, it is especially critical for SaaS solutions. The SaaS market is evolving and changing rapidly. If a SaaS vendor ceases or suspends operations, our access to both the application and our data is instantly cut off. The ability for a vendor to outsource their infrastructure and programming to third parties has allowed a proliferation of companies to enter the SaaS market.

Implication – A number of the newcomers to the SaaS market are relatively inexperienced in the processes and tools needed to run a high-availability, 24 x 7 production environment. Startup vendors may not be in a strong financial position. Selection and adoption of a business application usually requires significant investment on the part of the City. SaaS vendor evaluation requires a higher level of due diligence.

Conversely, a number of established COTS software vendors are in various stages of moving into the SaaS market, still trying to determine how a SaaS offering fits into their business model. Some of these transitions won't pan out and their future offerings may be significantly different or represent dead ends.

Questions to ask

- How strong is the vendor's commitment to SaaS? Do they have a clearly defined strategy for their SaaS solution? Is it their primary future direction? Have they invested resources at a level that confirms this?
- How long has the vendor had a SaaS offering and supported a high-availability, 24 x 7 production environment?
- Does the vendor have documentation that describes the tools and processes they use to proactively monitor system performance?
- How is the company funded? Are they financially sound?

Best Practice: Plan for the worst.

SaaS vendors have gone out of business, suffer catastrophic natural disasters, and change functionality so that applications no longer fit City business processes. It is important for the City to reduce the potential impacts of these risks.

Implication – Major risks for a planned implementation should be identified, and plans to mitigate those risks should be developed. The comprehensiveness of these plans should be based on the mission criticality of the system. For example, a highly mission critical system might require a formal Disaster Recovery/Business Continuity plan to be worked out between the vendor and the City. On the other hand, a SaaS service that merely aggregates already existing public data to allow public mashups probably would not likely require such a plan.

Questions to ask

- Does the vendor have a formalized Disaster Recovery/Business Continuity plan?
- Monitor vendor viability. This is typically not a formal process, but is instead just paying attention to red flags that might signal trouble. Things to look out for:
 - Has the vendor lost major clients to competitors?
 - Has the vendor gone through or announced significant layoffs?
 - Are you experiencing a change in service levels? Or is the vendor having trouble being responsive to your inquiries?