

Software as a Service

Software as a Service

- Cloud computing is an important trend that includes several categories of service, all offered on demand over the Internet in a pay-as-you-go model. Software as a service (SaaS) is one of these categories; others include platform as a service (PaaS) and infrastructure as a service (IaaS). See the sidebar “A Cloud Computing Taxonomy” on page 3 for more details.

Characteristics of Software as a Service (SaaS)

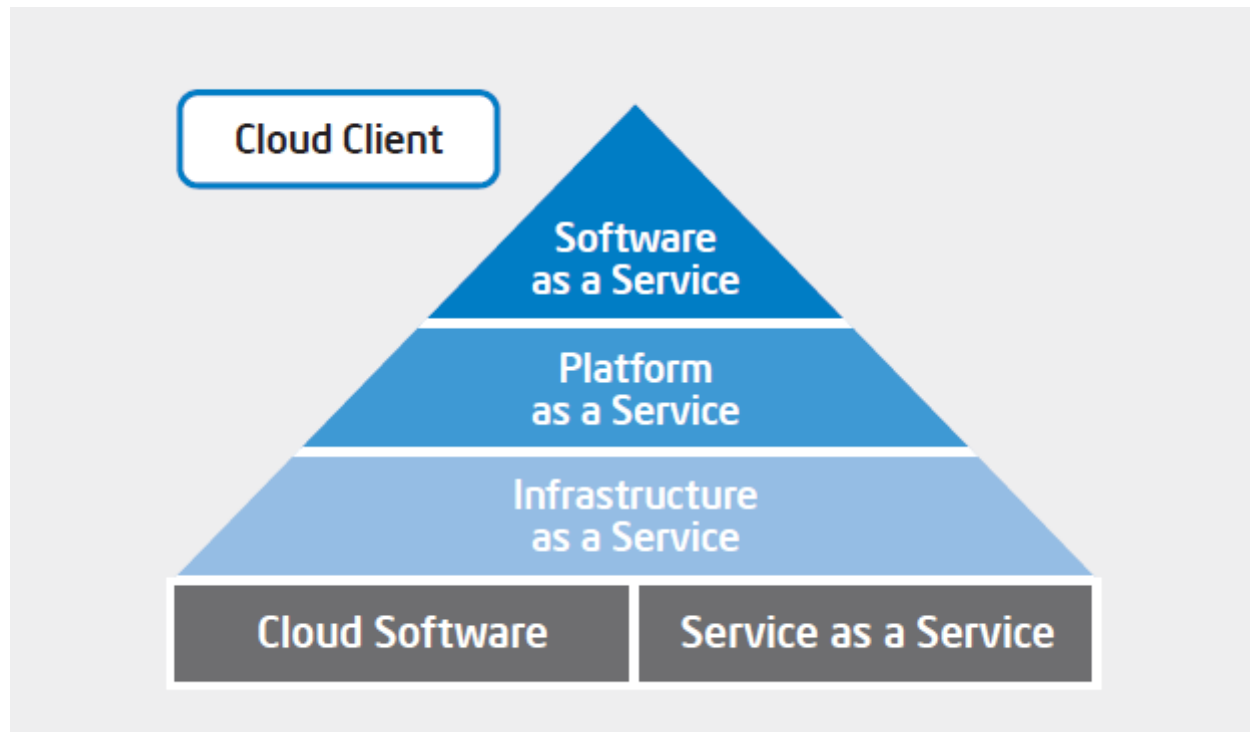
Application Categories	<ul style="list-style-type: none">▪ Good candidates for SaaS are applications with industry-standard workflows, which do not involve intellectual property or sensitive data, such as human resources management, employee stock options, medical benefits, and expense reports.
Users	<ul style="list-style-type: none">▪ The number of users varies—from participants in a small pilot project to a majority of Intel employees for several applications.▪ About half of SaaS applications are used globally and half are specific to the United States.
Business Drivers	<ul style="list-style-type: none">▪ Agility and time to market.▪ Lower cost: No need to develop and maintain internal expertise for commoditized capabilities with industry-standard workflows.
Costing Model	<ul style="list-style-type: none">▪ Most applications use subscription licensing, with a fee paid at regular intervals—yearly, quarterly, monthly.▪ A few applications are priced per transaction.
Platform	<ul style="list-style-type: none">▪ In about 30 percent of cases, suppliers provide a dedicated hardware platform for Intel's application; in the other 70 percent, the platform is shared. The application is typically not virtualized.▪ In about half of the cases, suppliers provide a dedicated application instance for Intel; the rest use a standard multi-tenant shared instance.
Disaster Recovery	<ul style="list-style-type: none">▪ There is a disaster recovery plan for most applications.
Security	<ul style="list-style-type: none">▪ All SaaS solutions have undergone a security risk assessment. Most data has a relatively low security rating.▪ Half of the SaaS applications use single sign-on (SSO); the rest use personal profiles.▪ Data may be encrypted in transit and at rest. Native Web applications use HTTPS/SSL to protect important data during transmission.
Monitoring	<ul style="list-style-type: none">▪ Typically, vendors monitor applications and infrastructure and send us selected or summary alerts.

A Cloud Computing Taxonomy

- Elements of the taxonomy include:
- **Software as a service (SaaS).** On-demand software applications. With SaaS, software applications are rented from a provider as opposed to purchasing them for enterprise installation and deployment. At the top of the pyramid, this is the most mature category of cloud service; a wide variety of applications are already available for enterprise use

- **Platform as a service (PaaS).** On-demand software development platforms.
- **Infrastructure as a service (IaaS).** On-demand computing infrastructure.
- **Cloud software.** Unique purchased/packaged software used to build and run cloud services.
- **Service as a service.** Horizontal service that is subscribed to and used as a component of SaaS, IaaS, or PaaS offerings. An example is a billing service.
- **Cloud client.** Client-centric services and run-

cloud computing taxonomy using input from existing works.



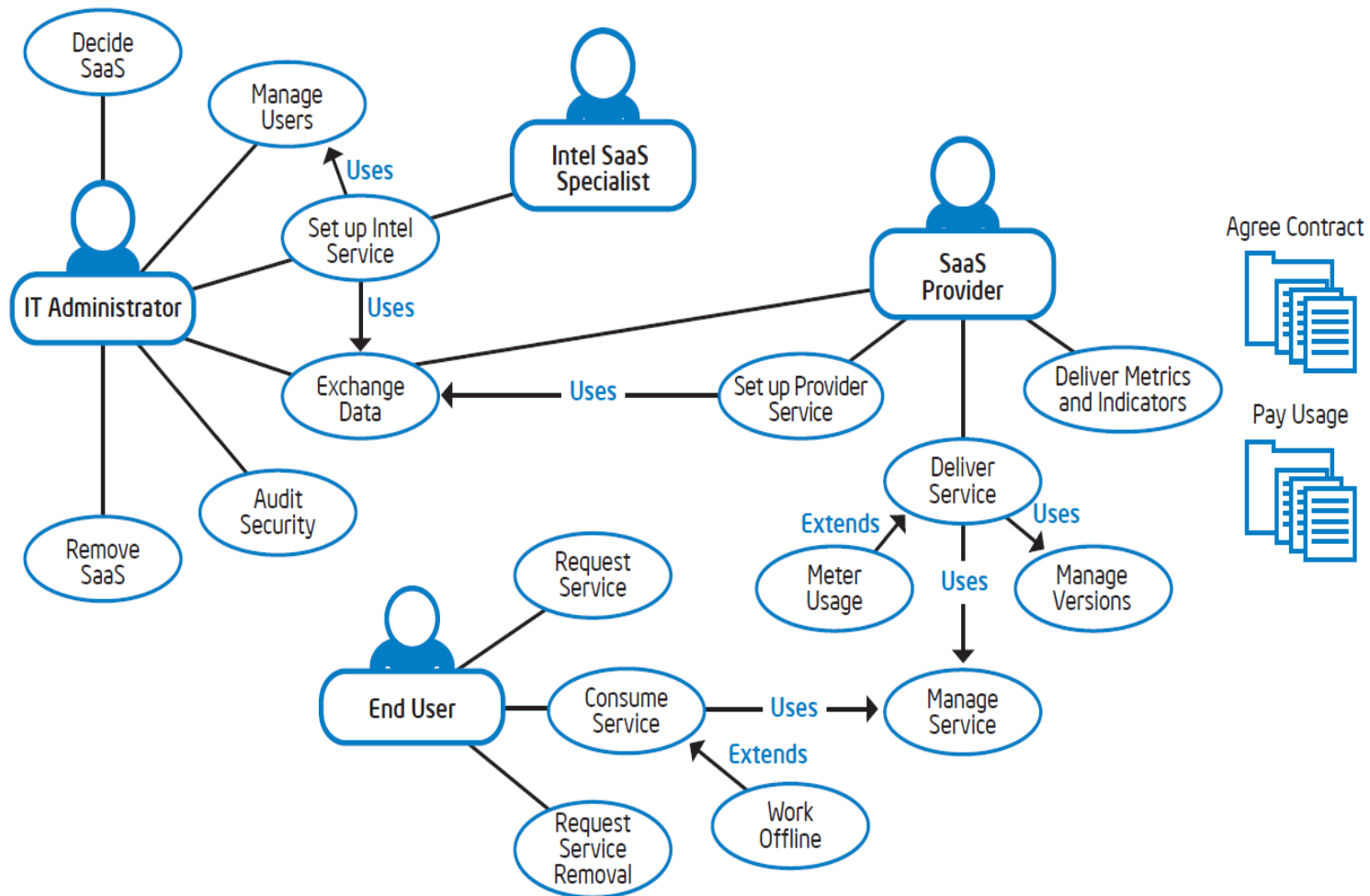


Figure 2. Software as a service (SaaS) architecture use-case model.

Use-Case Model

- There are three types of primary SaaS user roles within Intel.
- **IT users.** Primary IT user roles are applications administrators and SaaS specialists.
- **End users.** Primary end users are individual workers at Intel who use SaaS applications for job-related activities. Workers are located within the enterprise or connected to Intel while traveling or working from home.

Conceptual Architecture

- The conceptual architecture is intended to represent a three- to five-year vision of SaaS architecture, free of implementation technology details, and to establish common capability definitions.

KEY FEATURES

- **Multi-tenant efficient.** The design should support multiple tenants using a single instance of the application. The data must be segregated for each tenant.
- **Configurable.** The application can be configured to meet the needs of each tenant, using metadata and a metadata execution engine—also known as a business rules engine. Routine configuration changes should be possible without the need to coordinate

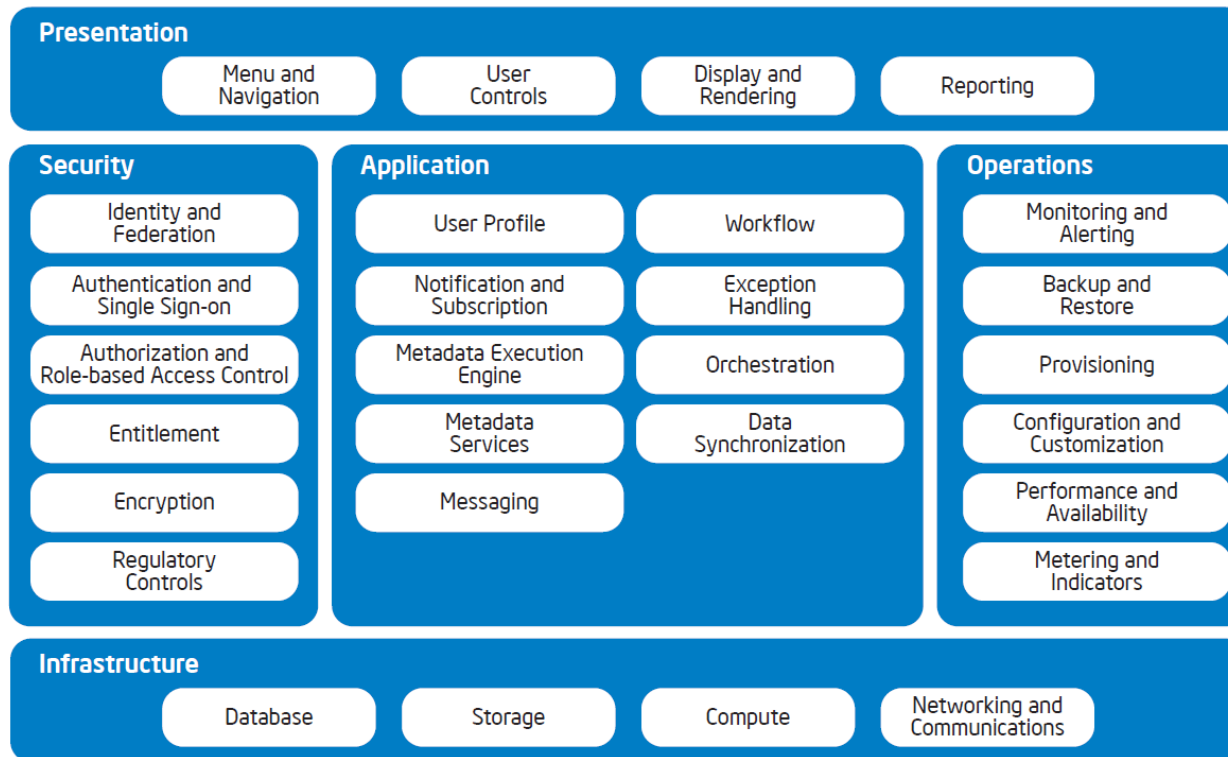
SAAS CAPABILITIES

- Many capabilities make up the SaaS conceptual architecture. We group these into presentation, security, application, operations, and infrastructure categories . This includes all capabilities exposed to the user, such as:
 - **Menu and navigation.** These provide access to the features and functionality within an application, organized in an intuitive way so that the user can select the desired function.

Security

- Security is one of the most important categories of SaaS capabilities, given that Intel's data and user accounts are typically hosted by the SaaS provider. We considered the following capabilities:
- **Identity and federation.**
- **Authentication and single sign-on (SSO).**
- **Authorization and role-based access control.**
- **● Entitlement.**

conceptual architecture for software as a service (SaaS)



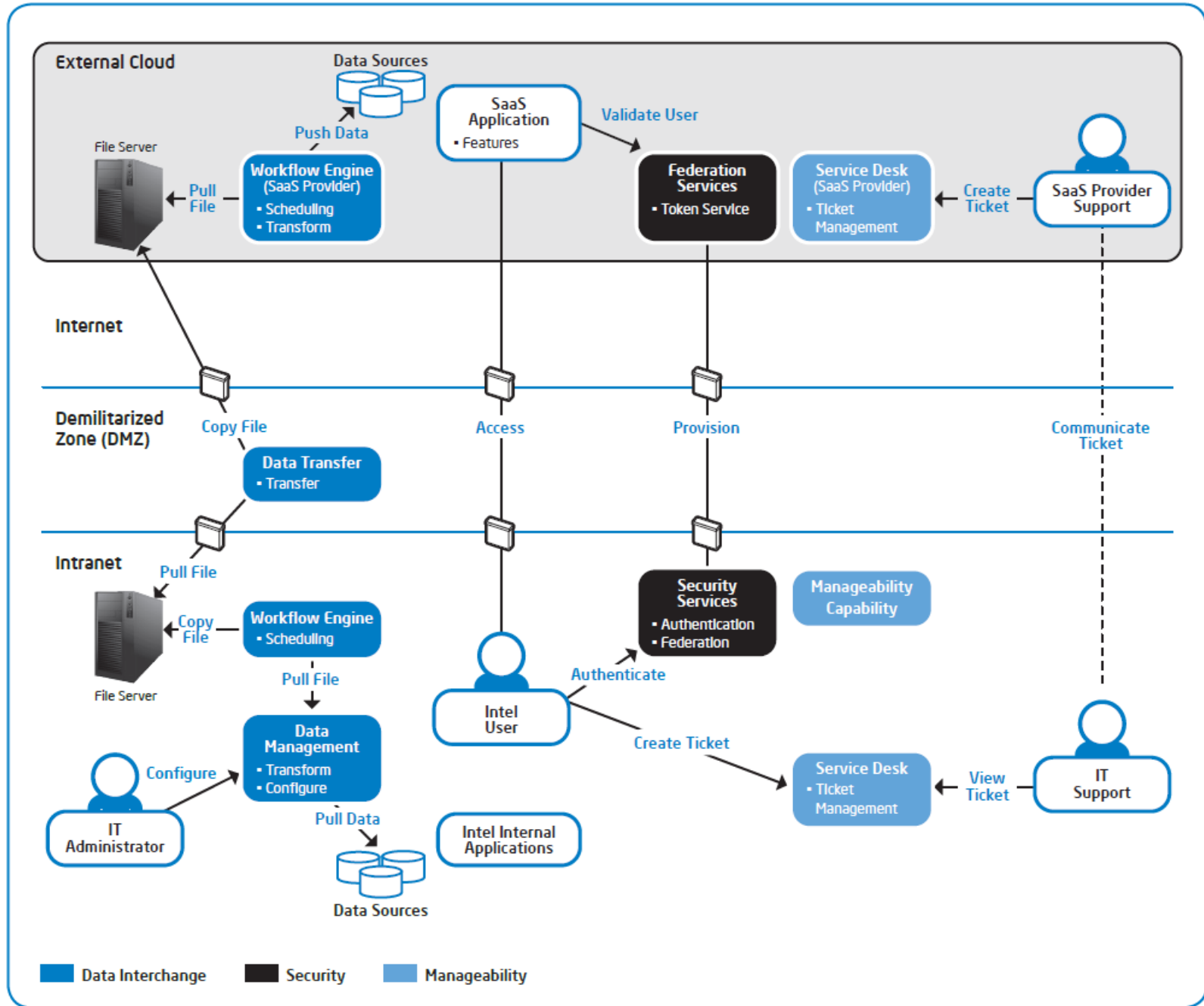


Figure 4. Intel IT's summary view of software as a service (SaaS) reference architecture.

Application

- These represent the typical business layer or middle tier of a SaaS application:
- **• User profile.** The attributes and information that describe a user, such as name, e-mail address, and role.
- **• Metadata execution engine.** Statements that define or constrain some aspect of the business. They are intended to assert business structure or to control or influence the behavior of the business.

- ● **Workflow.** The defined series of user-based tasks within a process to produce a final outcome. An example is creating a purchase order.
- ● **Exception handling.** The process of raising and managing exceptions within an application. This includes how application errors are exposed to the user and how error messages are logged.
- ● **Orchestration.** The series of technical tasks performed within a process to produce a final outcome. An example is an extract, transform, and load sequence to move data between

Operations

- These are the capabilities needed to efficiently keep the SaaS application running:
- **● Monitoring and alerting.** Polling application components, services, and infrastructure to detect failures. On detection, an alert is sent to the appropriate support group.
- **● Performance and availability.** Performance describes how the application performs under load, both in terms of the number of users and the transaction volume. In the context of

Infrastructure

- The underlying technical capabilities required for storing data and moving it around the network:
- **● Database.** In a multi-tenant data architecture, there could be one database per tenant or one database shared by multiple tenants with the data indexed by a specific tenant identification.
- **● Compute.** The physical clients, servers, or virtual machines that execute code.

Reference Architecture

- The purpose of the reference architecture is to provide a proven template solution that project teams can immediately apply to specific application domains.
- Accordingly, it includes only a subset of the capabilities described in the conceptual architecture and is more near-term in nature—one to two years.
- The reference architecture also provides a common vocabulary for discussing

DATA INTERCHANGE

- Intel SaaS project teams cite data interchange—moving employee data and other information between internal Intel systems and data stores hosted by the SaaS provider—as their biggest challenge.
- It's important to keep this data synchronized between internal and external systems, so data transfer may need to occur frequently, often on a scheduled basis.

SECURITY

- SaaS providers must comply with a number of security policies. We have done a good job of assessing providers in advance of implementation to help ensure they meet our requirements, but we will continue to move cautiously.
- Additional work is required to qualify an externally hosted SaaS solution, so it is important to identify whether SaaS is an option early in the life cycle of an application

From our environmental scan of existing deployments, we identified several other key elements needed for a successful project:

- Classification of the data by an IP attorney.
- Rigorous due diligence to help ensure the required controls are included in the contract.
- Completion of an information security risk assessment.
- Protection of data in transit and at rest.
- Making sure that suppliers provide satisfactory disaster recovery and business continuity plans.

MANAGEABILITY

- To date, automated manageability has not been a priority for us because we have had relatively few applications and they have generally not been business-critical.
- Intel relies largely on user feedback and SaaS provider data to help ensure the providers are meeting contractual obligations and addressing inadequacies.

- We plan to work with standards bodies to develop verifiable manageability standards and certifications for use by service providers. This will provide organizations consuming SaaS applications with a common set of metrics, which will eliminate many of the initial validation steps currently required.
- How much manageability data we will require from SaaS providers is still an open question. Our goal is to minimize introspection into incidents, focusing more on application performance, reliability, and common tracking of requirements.

CONCLUSION

- The success of SaaS applications at Intel to date, together with our industry analysis, suggests that adoption will continue to grow.
- Our goal is that our SaaS architecture enables Intel's use of SaaS to progress from organic growth to prescriptive deployment, with the reference architecture helping to drive consistent designs, quickly, for new SaaS projects.