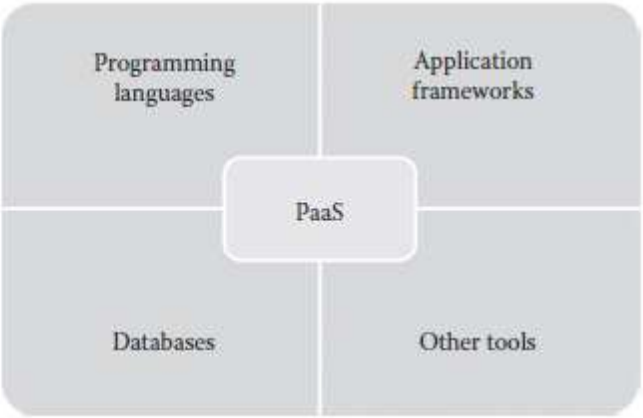

FIGURE 7 Services provided by PaaS providers.

3.1 Characteristics of PaaS

PaaS development platforms are different from the traditional application development platforms. The following are the essential characteristics that make PaaS unique from traditional development platforms:

1. All in one: Most of the PaaS providers offer services to develop, test, deploy, host, and maintain applications in the same IDE.

2. Web access to the development platform: A typical development platform uses any IDEs for developing applications. Typically, the IDE will be installed in the developer's machines. But, PaaS provides web access to the development platform. Using web UI, any developer can get access to the development platform. The web-based UI helps the developers create, modify, test, and deploy different applications on the same platform.

3. Offline access: A developer may not be able to connect to the Internet for a whole day to access the PaaS services. When there is no Internet connectivity, the developers should be allowed to work offline. To enable offline development, some of the PaaS providers allow the developer to synchronize their local IDE with the PaaS services. The developers can develop an application locally and deploy it online whenever they are connected to the Internet.

4. Built-in scalability: Scalability is an important requirement for the new-generation web or SaaS applications. It is very difficult to enable the dynamic scalability for any application developed using traditional development platforms. But, PaaS services provide built-in scalability to an application that is developed using any particular PaaS. This ensures that the application is capable of handling varying loads efficiently.

5. Collaborative platform: Nowadays, the development team consists of developers who are working from different places. There is a need for a common platform where the developers can collaboratively work together on the same project. Most of the PaaS services provide support for collaborative development. To enable collaboration among developers, most of the PaaS providers provide tools for project planning and communication.

3.2 Pros and Cons of PaaS

The main advantage of using PaaS is that it hides the complexity of maintaining the platform and underlying infrastructure. This allows the developers to work more on implementing the important functionalities of the application. Apart from this, the PaaS has the following benefits:

1. Quick development and deployment: PaaS provides all the required development and testing tools to develop, test, and deploy the software in one place. Most of the PaaS services automate the testing and deployment process as soon as the developer completes the development. This speeds up application development and deployment than traditional development platforms.

2. Reduces TCO: The developers need not buy licensed development and testing tools if PaaS services are selected. Most of the traditional development platforms requires high-end infrastructure for its working, which increases the TCO of the application development company. But, PaaS allows the developers to rent the software, development platforms, and testing tools to develop, build, and deploy the application. PaaS does not require high-end infrastructure also to develop the application, thus reducing the TCO of the development company.

3. Supports agile software development: Nowadays, most of the new-generation applications are developed using agile methodologies. Many ISVs and SaaS development companies started adopting agile methodologies for application development. PaaS services support agile methodologies that the ISVs and other development companies are looking for.

4. Different teams can work together: The traditional development platform does not have extensive support for collaborative development.

6. Diverse client tools: To make the development easier, PaaS providers provide a wide variety of client tools to help the developer. The client tools include CLI, web CLI, web UI, REST API, and IDE. The developers can choose any tools of their choice. These client tools are also capable of handling billing and subscription management.

PaaS services support developers from different places to work together on the same project. This is possible because of the online common development platform provided by PaaS providers.

5. Ease of use: The traditional development platform uses any one of CLI- or IDE-based interfaces for development. Some developers may not be familiar with the interfaces provided by the application development platform. This makes the development job a little bit difficult. But, PaaS provides a wide variety of client tools such as CLI, web CLI, web UI, APIs, and IDEs. The developers are free to choose any client tools of their choice. Especially, the web UI–based PaaS services increase the usability of the development platform for all types of developers.

6. Less maintenance overhead: In on-premise applications, the development company or software vendor is responsible for maintaining the underlying hardware. They need to recruit skilled administrators to maintain the servers. This overhead is eliminated by the PaaS services as the underlying infrastructure is maintained by the infrastructure providers. This gives freedom to developers to work on the application development.

7. Produces scalable applications: Most of the applications developed using PaaS services are web application or SaaS application. These applications require better scalability on the extra

load. For handling extra load, the software vendors need to maintain an additional server. It is very difficult for a new start-up company to provide extra servers based on the additional load. But, PaaS services are providing built-in scalability to the application that is developed using the PaaS platform.

PaaS provides a lot of benefits to developers when compared to the traditional development environment. On the other hand, it contains drawbacks, which are described in the following:

1. Vendor lock-in: The major drawback with PaaS providers are vendor lock-in. The main reason for vendor lock-in is lack of standards. There are no common standards followed among the different PaaS providers. The other reason for vendor lock-in is proprietary technologies used by PaaS providers. Most of the PaaS vendors use the proprietary technologies that are not compatible with the other PaaS providers. The vendor lock-in problem of PaaS services does not allow the applications to be migrated from one PaaS provider to the other.

2. Security issues: Like in the other cloud services, security is one of the major issues in PaaS services. Since data are stored in off-premise third-party servers, many developers are afraid to go for PaaS services. Of course, many PaaS providers provide mechanisms to protect the user data, and it is not sufficient to feel the safety of on-premise deployment. When selecting the PaaS provider, the developer should review the regulatory, compliance, and security policies of the PaaS provider with their own security requirements. If not properly reviewed, the developers or users are at the risk of data security breach.

3. Less flexibility: PaaS providers do not give much freedom for the developers to define their own application stack. Most of the PaaS providers provide many programming languages, databases, and other development tools. But, it is not extensive and does not satisfy all developer needs. Only some of the PaaS providers allow developers to extend the PaaS tools with the custom or new programming languages. Still most of the PaaS providers do not provide flexibility to the developers.

4. Depends on Internet connection: Since the PaaS services are delivered over the Internet, the developers should depend on Internet connectivity for developing the application. Even though some of the providers allow offline access, most of the PaaS providers do not allow offline access. With slow Internet connection, the usability and efficiency of the PaaS platform do not satisfy the developer requirements.

4 Software as a Service

SaaS changes the way the software is delivered to the customers. In the traditional software model, the software is delivered as a license-based product that needs to be installed in the end user device. Since SaaS is delivered as an on-demand service over the Internet, there is no need to install the software to the end user’s devices. SaaS services can be accessed or disconnected at any time based on the end user’s needs. SaaS services can be accessed from any lightweight web browsers on any devices such as laptops, tablets, and smartphones. Some of the SaaS services can be accessed from a thin client that does not contain much storage space and cannot run much software like the traditional desktop PCs. The important benefits of using thin clients for accessing the SaaS application are as follows: it is less vulnerable to attack, has a longer life cycle, consumes less power, and is less expensive. A typical SaaS provider may provide business services, social networks, document management, and mail services as shown in Figure 8:

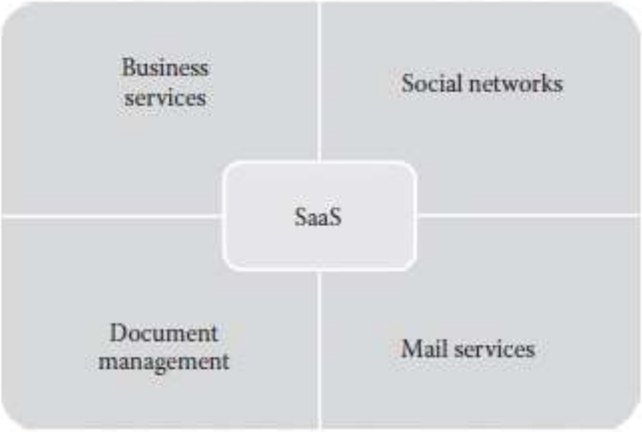

FIGURE 8 Services provided by SaaS Providers.

1. Business services: Most of the SaaS providers started providing a variety of business services that attract start-up companies. The business SaaS services include ERP, CRM, billing, sales, and human resources.

2. Social networks: Since social networking sites are extensively used by the general public, many social networking service providers adopted SaaS for their sustainability. Since the number of users of the social networking sites is increasing exponentially, cloud computing is the perfect match for handling the variable load.

3. Document management: Since most of the enterprises extensively use electronic documents, most of the SaaS providers started providing services that are used to create, manage, and track electronic documents.

4. Mail services: E-mail services are currently used by many people. The future growth in e-mail usage is unpredictable. To handle the unpredictable number of users and the load on e-mail services, most of the e-mail providers started offering their services as SaaS services.

4.1 Characteristics of SaaS

SaaS services are different and give more benefits to end users than the traditional software. The following are the essential characteristics of SaaS services that make it unique from traditional software:

1. One to many: SaaS services are delivered as a one-to-many model where a single instance of the application can be shared by multiple tenants or customers.

2. Web access: SaaS services provide web access to the software. It allows the end user to access the application from any location if the device is connected to the Internet.

3. Centralized management: Since SaaS services are hosted and managed from the central location, management of the SaaS application becomes easier. Normally, the SaaS providers will perform the automatic updates that ensure that each tenant is accessing the most recent version of the application without any user-side updates.

4. Multidevice support: SaaS services can be accessed from any end user devices such as desktops, laptops, tablets, smartphones, and thin clients.

5. Better scalability: Since most of the SaaS services leverage PaaS and IaaS for its development and deployment, it ensures a better scalability than the traditional software. The dynamic scaling of underlying cloud resources makes SaaS applications work efficiently even with varying loads.

6. High availability: SaaS services ensure the 99.99% availability of user data as proper backup and recovery mechanisms are implemented at the back end.

7. API integration: SaaS services have the capability of integrating with other software or service through standard APIs.

4.2 Pros and Cons of SaaS

SaaS applications are used by a wide range of individuals and start-up industries for its cost-related benefits. Apart from the cost-related benefits, SaaS services provide the following benefits:

1. No client-side installation: SaaS services do not require client-side installation of the software. The end users can access the services directly from the service provider data center without any installation. There is no need of high-end hardware to consume SaaS services. It can be accessed from thin clients or any handheld devices, thus reducing the initial expenditure on buying high-end hardware.

2. Cost savings: Since SaaS services follow the utility-based billing or pay-as-you-go billing, it demands the end users to pay for what they have used. Most of the SaaS providers offer different subscription plans to benefit different customers. Sometimes, the generic SaaS services such as word processors are given for free to the end users.

3. Less maintenance: SaaS services eliminate the additional overhead of maintaining the software from the client side. For example, in the traditional software, the end user is responsible for performing bulk updates. But in SaaS, the service provider itself maintains the automatic updates, monitoring, and other maintenance activities of the applications.

4. Ease of access: SaaS services can be accessed from any devices if it is connected to the Internet. Accessibility of SaaS services is not restricted to any particular devices. It is adaptable to all the devices as it uses the responsive web UI.

5. Dynamic scaling: SaaS services are popularly known for elastic dynamic scaling. It is very difficult for on-premise software to provide dynamic scaling capability as it requires additional hardware. Since the SaaS services leverage elastic resources provided by cloud computing, it can handle any type of varying loads without disrupting the normal behavior of the application.

6. Disaster recovery: With proper backup and recovery mechanisms, replicas are maintained for every SaaS services. The replicas are distributed across many servers. If any server fails, the end user can access the SaaS from other servers. It eliminates the problem of single point of failure. It also ensures the high availability of the application.

7. Multitenancy: Multitenancy is the ability given to the end users to share a single instance of the application. Multitenancy increases resource utilization from the service provider side.

Even though SaaS services are used by many individuals and start-up industries, the adoption from the large industries is very low. The major problem with SaaS services is security to the data. All companies are worried about the security of their data that are hosted in the service provider data center. The following are the major problems with SaaS services:

1. Security: Security is the major concern in migrating to SaaS application. Since the SaaS application is shared between many end users, there is a possibility of data leakage. Here, the data are stored in the service provider data center. We cannot simply trust some third-party service provider to store our company-sensitive and confidential data. The end user should be careful while selecting the SaaS provider to avoid unnecessary data loss.

2. Connectivity requirements: SaaS applications require Internet connectivity for accessing it. Sometimes, the end user's Internet connectivity might be very slow. In such situations, the user cannot access the services with ease. The dependency on high-speed Internet connection is a major problem in SaaS applications.

3. Loss of control: Since the data are stored in a third-party and off-premise location, the end user does not have any control over the data. The degree of control over the SaaS application and data is lesser th