Chapter 4
Constructing Green Software Services: From Service Models to Cloud-Based Architecture

Fei Li, Soheil Qanbari, Michael Vögler, and Schahram Dustdar

4.1 Introduction

In recent years, green software research is gaining momentum from the acute need for sustainable development as well as the far-reaching effect of ICT to our society. "[Green and] Sustainable Software is software, whose direct and indirect negative impacts on economy, society, human beings, and environment that result from development, deployment, and usage of the software are minimal and/or which have a positive effect on sustainable development" [4]. Based on this definition, green software research is growing in two directions. The first direction looks into the runtime energy consumption of software [15] and its engineering pro-aspects of our society and investigates how software can be used to improve the sustainability of a broader range of business, social, and individual activities [5]. This chapter is focused on the research and development in the second direction—to leverage software to solve sustainability problems on a wider scope.

The emergence of cloud computing and Internet of Things (IoT) makes software services further reach out to the physical world at a larger scale. Many existing business operations are being improved with respect to scalability and manageability through automation. Such new developments have strong implications to green software research since sustainability can be improved by applying software services based on these new computing paradigms. This view prompts us to rethink the delivery models and service scope of green software: when green software is delivered as online services, or green software services (GSS), a broader range of business, governments, and individual processes can more easily employ the services to reduce their energy consumption. Furthermore, more flexible business relationships can be established between different stakeholders so that the financial

F. Li • S. Qanbari (✉) • M. Vögler • S. Dustdar
Distributed System Group, Vienna University of Technology, Vienna, Austria
e-mail: Li@dsg.tuwien.ac.at; Qanbari@dsg.tuwien.ac.at; voegler@dsg.tuwien.ac.at; dustdar@dsg.tuwien.ac.at

C. Calero, M. Piattini (eds.), *Green in Software Engineering*,
DOI 10.1007/978-3-319-08581-4_4

and social interests of green software can in turn promote its research and development.

To this end, this chapter presents the core GSS constructs based on a cloud-based architecture. The research is aimed at providing a systematic, high-level view on four key elements in the development of GSS: stakeholders, their requirements, various business models, and corresponding software architecture. The stakeholders of GSS are detailed with the services they can provide and consume, thus clarifying their interests to GSS. Based on this analysis, we present the domain-independent core requirements to GSS that are considered by different stakeholders. Six business models are proposed to promote collaborations of stakeholders on the delivery of GSS. Each of the business models is then mapped to a scope of high-level components in the IoT PaaS architecture [11]. This chapter completes our previous work on business model analysis for GSS [6] by providing the software architecture for each model, which can serve as the reference for constructing GSS by service providers.[1] In the end, we will discuss how the business models are related to cloud services and the challenges of realizing a marketplace for GSS.

The remaining part of this chapter is structured as follows: Sect. 4.2 introduces the core stakeholders and their relationships in GSS. Then the requirements for GSS are analyzed in Sect. 4.3. Section 4.4 introduces the business canvas for describing business models and our past work on IoT PaaS as the basis for GSS architecture. Section 4.5 presents the business models and corresponding cloud services. Finally, Sect. 4.6 discusses the implications of these business models to the research and development of GSS, and the chapter is concluded in Sect. 4.7.

4.2 Stakeholders in GSS

In order to investigate the requirements for GSS, we first conduct a detailed analysis on the related stakeholders. Overall, the stakeholders are classified into core stakeholders and supportive stakeholders. Core stakeholders are those business entities that have direct business interests in providing GSS, whereas supportive stakeholders are those organizations or individuals who have financial or social interests in GSS but do not directly profit from delivering GSS. The stakeholders are illustrated in Fig. 4.1.

[1] Part of this chapter was published in [6], and this chapter extends the previous publication by adding the software architecture for each model.

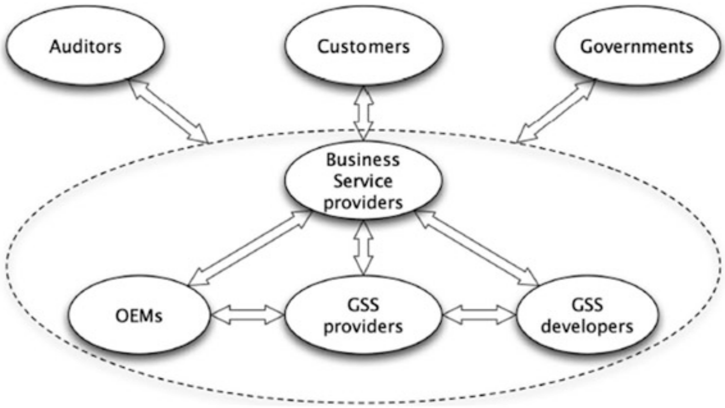

Fig. 4.1 Stakeholders of GSS

4.2.1 Core Stakeholders

- **Business service providers** are operating diverse businesses that might benefit from GSS, for example, building operators, transportation services, and data centers. They share one common objective of maximizing the sustainability of their businesses (by saving energy or reducing wastes). Since they have the direct financial incentives of reducing operational costs by applying GSS to their existing businesses, they are the main driving force for the development of GSS.
- **Original Equipment Manufacturer (OEM)** produce equipment that are the source of energy consumption. Their efforts alone on developing sustainable equipment can result in significant energy conservation (e.g., LED lights and energy-efficient chillers). Integrating, managing, and leveraging the energy-saving capabilities of OEM devices are one of the most important approaches of realizing GSS. More importantly, GSS is able to optimize complex systems that constitute a large number of OEM equipment [3].
- **GSS providers** offer GSS that are used by business service providers. The services are in diverse business domains and of various functions, such as home automation, facility management, offline analysis, and so on. The services provided by them can be realized by other stakeholders or domain experts. GSS providers retain the service interface and establish direct business relationships with customers who need GSS.
- **GSS developers** realize business logics and optimization methods in the target domain. Different implementations of the same business logic could have considerably different effects in terms of sustainability. Thus, domain knowledge is often required for GSS developers. Promoting a GSS developer community will help to leverage the growing amount of data available on the Internet and the increasingly connected devices [1] to create more diverse GSS and apply them to more business domains.

Table 4.1 Roles of stakeholders

Stakeholders	Services or information provided	Services or information consumed
Business service providers	• Domain-specific business service • Domain process optimization knowledge	• Optimization services • Auditing
GSS providers	• Device integration • Optimization services • Data acquisition • Analytics • Data visualization	• Domain-specific knowledge • Device connectivity • Application development
OEMs	• Devices • Device optimization knowledge	• Device integration
Application developers	• Business logic implementation • Optimization method implementations	• GSS platform services • Domain-specific knowledge • Device optimization knowledge
Governments	• Raising public awareness to GSS • Public information • Regulation and legislation • Policy enforcement	• Standardization • Auditing results
Auditors	• Auditing	• Data access • Process monitoring
Service consumers	• Usage feedbacks	• GSS • Business services

4.2.2 Supportive Stakeholders

- **Governments** are strong driving forces and important advocates for promoting sustainable development. Their activities such as policy-making, policy enforcement, legislation, and standard enactment are essential for the adoption and long-term growth of GSS. Governments are also important public information providers.
- **Auditors** systematically assess the performance of GSS. They provide a solid baseline for comparing and further improving GSS by applying standardized evaluation methods. Audit may be applied to any system components or domain-specific business services.
- **Service consumers** are, in most current GSS applications, passive stakeholders who benefit from business services at reduced costs. However, since the behavior of consumers is also decisive to the effects of GSS, the usage information and behavior patterns can be collected for designing better GSS.

Table 4.1 summarizes the main services provided and consumed by each stakeholder.

4.3 Requirements for GSS

Based on the previous analysis about different stakeholders and their involvement in GSS, this section presents the high-level requirements to GSS. These requirements are intended to be domain independent, as each of them addresses the needs of multiple stakeholders:

1. *Identifying core services (concerned stakeholders: all core stakeholders)*

   Given our perspective that green software is a software designed to improve the sustainability of other business, social, or individual activities, the easiness of engaging with various target domains is the key to the wide adoption of GSS. This prompts us to identify a set of core GSS capabilities that are independent from the domain specifics of target systems while making GSS easily adaptable to address domain requirements. Therefore, we regard the following core capabilities critical to the success of GSS. These core capabilities form the basic features for a domain-independent platform that can serve as the basis of green software services:

   - **Collecting and preparing data from target systems**: This means that GSS should have the capability to access and acquire raw data from diverse environments that are to be made 'greener', including physical environments, hardware, software, and information generated by humans. Thus, GSS should not be restricted to certain communication protocols or data exchange formats. Furthermore, for the data to be effectively utilized in GSS, GSS has to prepare the raw data at two levels. The basic level is syntactic preparation to normalize the presentation of the data. This task is covered in many domain-specific standardization efforts like oBIX (Open Building Information Exchange) [14]. The higher level of data preparation is in the ongoing research on semantic technologies, which aims at the semantic interoperability of systems [13].
   - **Customizing for different target systems**: Providing the GSS to a specific target system means that GSS should be tailored on provisioning. Such tailoring can either be physically separating system components, configuring them, and deploying only the necessary components for the target system or virtually excluding the target system from using other irrelevant system capabilities. Thus, extensibility and customizability are critical to GSS. This chapter will illustrate how such tailoring can result in various service models.
   - **Accommodating various scales**: The target systems of GSS may be of largely different scales, ranging from a single home to a large city.[2] They differ with each other in terms of numbers and types of equipment, data volume, and resource requirements for applications. Thus, GSS should be customized not only for specific functions but also customized in terms of the

[2] http://www.pacificcontrols.net/projects/ict-project.html

resources needed for each target system. This requires flexible allocation of computing resources and on-demand scaling of GSS.

2. *Supporting a broad range of process optimization and analytics methods (concerned stakeholders: GSS providers, business service providers, GSS developers)*

   Based on the data collected, GSS will support a variety of process optimization methods on the underlying resources they are running to ensure the reliability, sustainability, and cost-effectiveness of the semantics they have promised. GSS is not limited to a certain optimization method but has to decide on the exact method to be used according to the specific task at hand. The known methods for optimizing energy usage include offline data modeling and simulation [3, 17], context-aware controls (e.g., presence-based light control), and agent-based systems [9]. The growing amount of data further requires GSS to be employed for results in big data research. The implication of this requirement is that the optimization capability of GSS lies in the capabilities of handling various types of data formats, including time series data, well-structured data, unstructured data, or even natural language.
3. *Supporting realization and enforcement of sustainability policies (concerned stakeholders: GSS providers, business service providers, governments, auditors)*

   Supporting sustainability policies is one of the basic expectations from GSS. Multiple aspects need to be considered for the realization of sustainability policies. First of all, GSS is required to model and understand the sustainability policies of the target systems, for example, temperature limit for a green building. Second, GSS is required to find efficient ways of meeting the goals defined by the policies, for example, how to efficiently control the HVAC systems. Third, in realizing the policies, GSS themselves should be energy efficient, complying to sustainability policies for IT systems.
4. *Ensuring end-to-end privacy and security coverage (concerned stakeholders: business service providers, service consumers)*

   Given the sensitivity to private, commercial, and public data that might be used in GSS, data privacy, security, and confidentiality have to be incorporated into GSS from the beginning and visible during every stage of development of the system. Furthermore, since GSS usually need to apply certain controls or changes in order to change the energy consumption of target systems, such control capabilities are to be secured so that only the authorized software components and personnel can perform the allowed controls.
5. *Supporting collaborations between stakeholders (concerned stakeholders: all stakeholders)*

   As stated in the stakeholder analysis, successful GSS are not built by any single party but built by the collaborations of multiple stakeholders, each of which provides their knowledge and services. The collaborations are the basis for flexible business models. Thus, GSS need to support collaborations by providing interfaces for different stakeholders, for example, interfaces for third-party developers to develop new functions or interface for auditors to inspect the sustainability status. More importantly, such collaborations are not only reflected on system interfaces but also the capability of directly presenting and sharing business interests on serving certain needs of GSS stakeholders.

4.4 Background

The impetus behind business model development is how to create, deliver, and capture the values of a system. The fast pace of cloud innovation and increasing diversity of GSS, coupled with unpredictable and ever-changing business requirements, require flexible and adaptive business models built upon reliable framework to maximize GSS utilization. Before proceeding to the introduction of business models and corresponding architectures, this section will introduce the business model description approach and the IoT PaaS architecture.

4.4.1 Business Model Description

This chapter leverages an established business model framework [2] to describe the business models of GSS. The framework is illustrated in Fig. 4.2.

A business model is described in four areas—*finance, infrastructure, customer, and value proposition*. These four sections have strong and mutual interrelationships with each other that have to be taken into account in forming business models. Financial aspects aim at providing profitable and sustainable revenue streams. The cost structure in the financial area is directly related to the stakeholders who are providing resources and conducting service activities, whereas the revenue structure is related to customers who are interested in the specific services. The monetary flow of this cost and revenue streams are effectively in use under the two models of *metered usage of service* and *subscription basis*. In the infrastructure area, the OEM devices are provided with optimization capabilities. The infrastructure aspect offers virtualization layer over infrastructure resources by providing utilization interfaces. The customer area's focus is on providing interfaces that define the consumer segments with their communication, distribution, and sales channels as a touching point for service delivery. Overall, the three areas converge on the value proposition of a business model. It seeks to solve customer problems and satisfy their needs with value propositions. Readers can refer to Bucherer et al. [2] to find more details on the description framework.

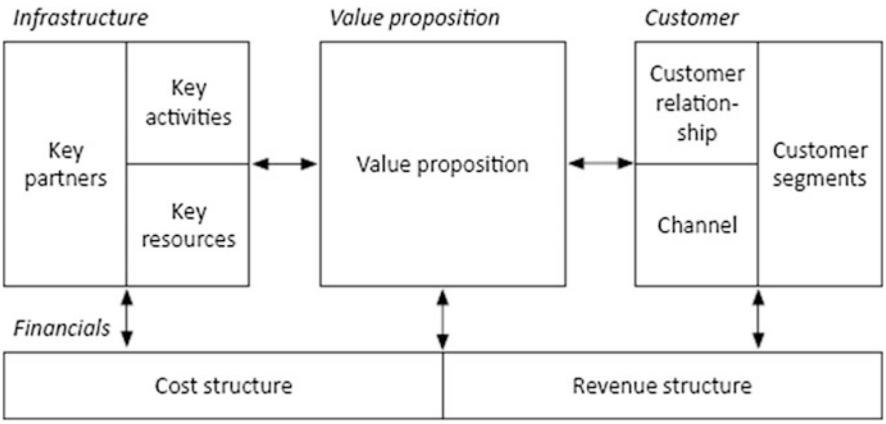

Fig. 4.2 The method of describing business models

4.4.2 IoT PaaS Architecture

The overall IoT PaaS architecture is depicted in Fig. 4.3. IoT PaaS is a domain-independent platform-as-a-service framework. In general, IoT solutions are highly domain specific, so the IoT PaaS framework is built to be generic and extendable enough to be used in different IoT domains. Furthermore, IoT PaaS provides essential platform services on cloud that can be used and extended by IoT solution providers.

To get a better understanding of the overall IoT PaaS architecture, we will start at the lowest tier, the *infrastructure*, which can be seen at the bottom of Fig. 4.3. The infrastructure inherits OEM devices, databases, file systems, and computation units. To communicate with these devices, the *virtualization* layer is used. This virtualization layer is both an integral part of infrastructure and platform. It provides device drivers and several low-level communication protocols to connect heterogeneous devices and furthermore offers a new level of abstraction by efficiently translating device/network interfaces to software interfaces. To deal with different domain-specific data models, which would lead to a new level of heterogeneity, the IoT PaaS uses *domain mediators* to mediate between different virtualized device interfaces. IoT PaaS provides two types of services related to data to handle real-time and persisted data, respectively. *Data processing* is focused on the processing and analyzing of real-time data generated by, for example, sensory devices, whereas data storing/retrieving facilitates storing, retrieving, and manipulating of persisted data by hiding the actual underlying data infrastructure. Since in IoT PaaS, each application runs in a complex and dynamic context, *application context management* is focused on providing and maintaining optimal runtime resources and software configurations for applications. Based on the resources acquired through tenant management, application context management helps applications to select the necessary resources at runtime to fulfil the

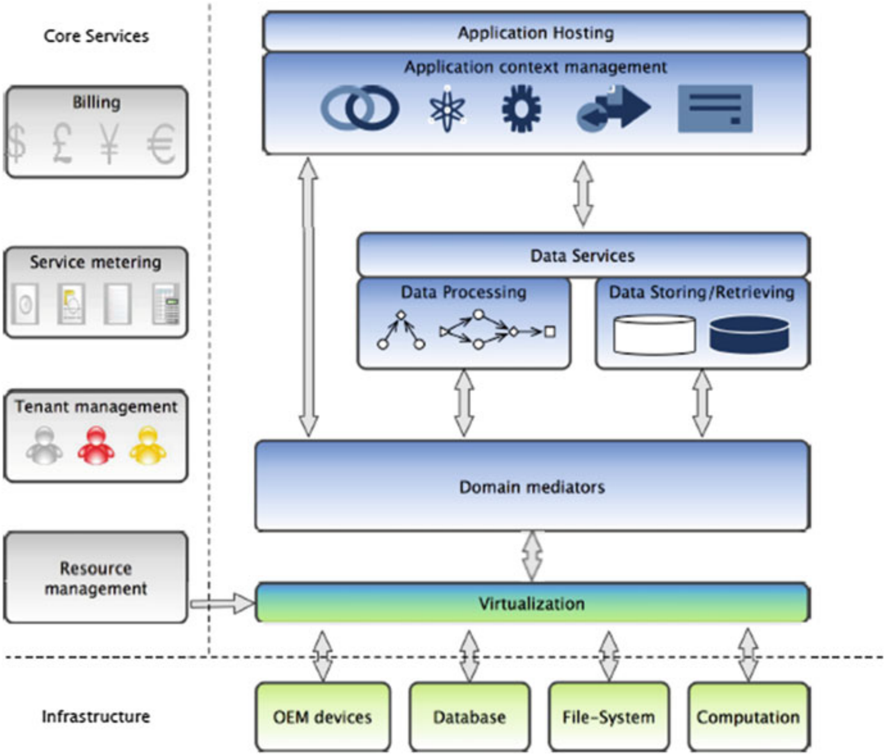

Fig. 4.3 The IoT PaaS architecture

functional requirements and meet service-level agreements and cost targets. The combination of tenant management and application context management provides a virtually isolated operational environment, enacting the concept of virtual verticals, for each application.

In addition to platform-specific components, IoT PaaS also offers a set of core services, which can be seen on the left-hand side of the figure. One of the most important services is *resource management*, which provides a registration point for any form of resource, for example, cloud resources, virtualized devices, control applications, etc. It monitors the resource status and enforces the access policies via the virtualization layer. In IoT PaaS, resources include not only cloud resources such as virtual machines and software instances in traditional cloud offerings but also IoT resources. Since device capabilities and control applications can be used by multiple tenants via virtualization, IoT PaaS uses *tenant management*, which assembles a consolidated view of the resources that are accessible by each tenant. To measure the usage of various services that can be used by an application, the *service metering* component mainly monitors service messages and invocations that are concerned by the platform and stakeholders. The metered information of both IoT and cloud resources gets composed to provide a comprehensive view of service usage. To conclude the set of core services, *billing* generates bills for stakeholders by analyzing the metered information according to charging schemes, which gets configured by stakeholders.

4.5 Business Models and Reference Architecture

The GSS architectures that fulfill the requirements and satisfy the stakeholders will become the cornerstone of flexible GSS business models.

4.5.1 Infrastructure Services

The capabilities of OEM devices and other computational resources play a significant role in the value generation of infrastructure services. GSS customers can benefit from infrastructure services for accessing these capabilities through virtualization, as illustrated in Fig. 4.4.

Its core value proposition is to provide optimization services on OEM devices, thus directly reducing the cost of using these devices. Resources are virtualized [8] so that GSS can easily access and control them. GSS providers are responsible for device virtualization that opens up the APIs for customers to access the infrastructure capabilities, as illustrated in Fig. 4.5.

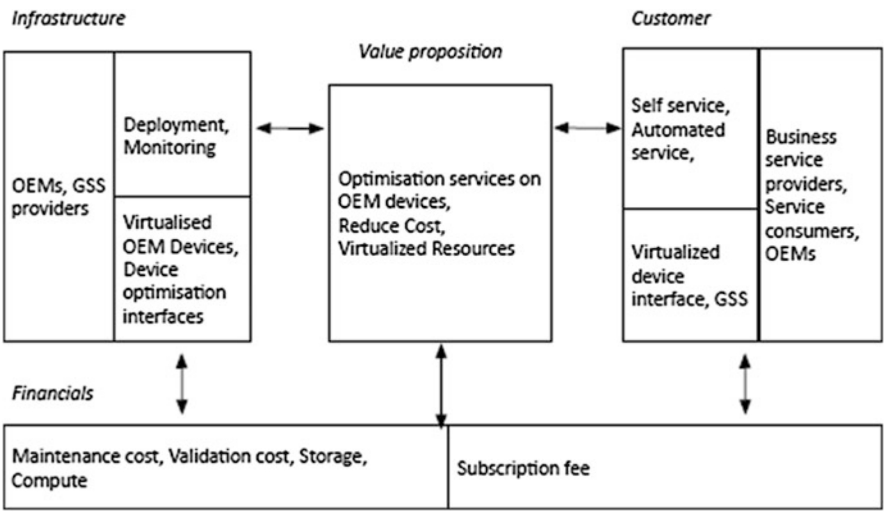

Fig. 4.4 Infrastructure services

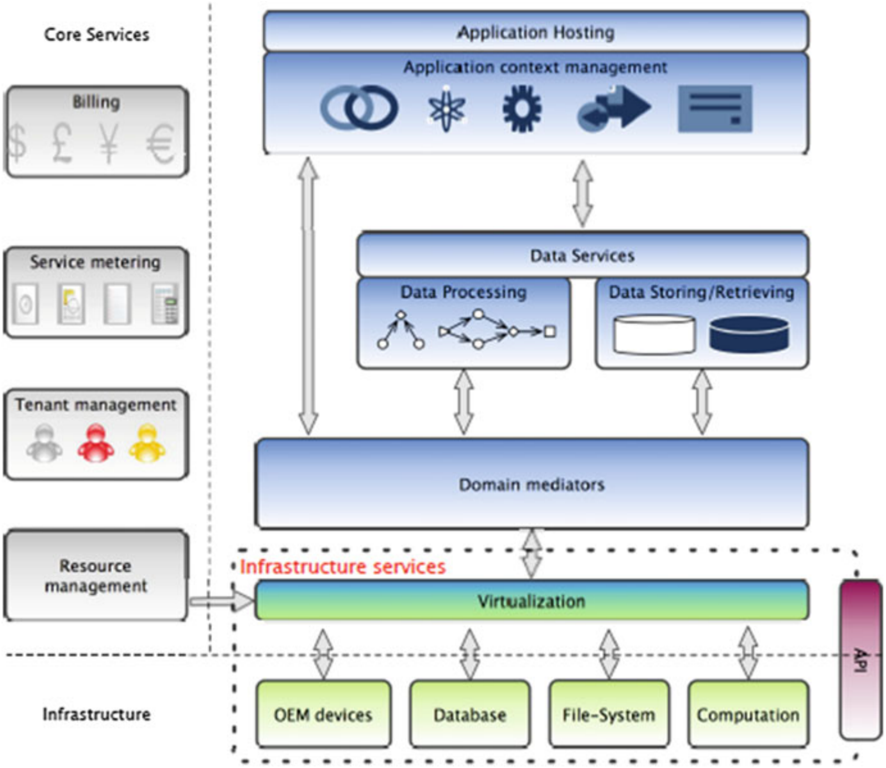

Fig. 4.5 Infrastructure services

The customer value that can be created by this model is mainly to provide business service providers and their consumers the ability to efficiently operate their facilities without constantly requesting the support of OEMs.

OEMs also benefit from this model through improved automation on their customer services since maintenance activities can be automated. The services under this model can be charged by "subscription" or "pay-as-you-go" models.

This business model provides an abstraction layer with programmable interfaces to perform administrative tasks over infrastructure resources. Therefore, GSS consumers do not manage or control the underlying resources but have control over how the infrastructure capabilities are used. Last but not least, another significant payback of these services is to cope with load fluctuations in an automated and consistent manner by cooperating with other virtualized resources, such as data storage and computational power.

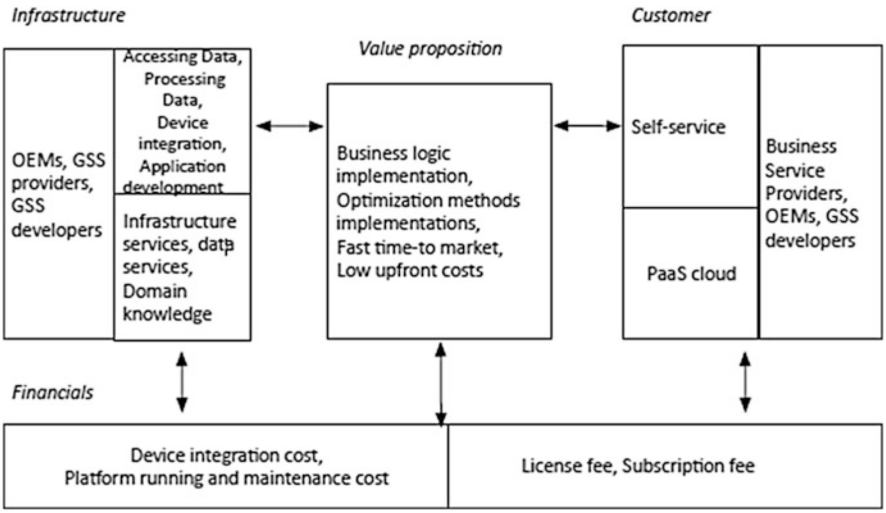

Fig. 4.6 Platform services

4.5.2 Platform Services

The platform services business model is illustrated in Fig. 4.6. It aims at providing GSS providers with a platform including application libraries, device integration APIs, data services, development environment comprising the end-to-end life cycle of GSS application coding, customization, testing, deploying, and hosting the applications as a service, as illustrated in Fig. 4.7.

The service provider of this model is not limited to certain business domains or optimization methods. OEMs can use the platform to realize the first business model—infrastructure services. GSS developers can develop GSS for specific customers on top of the platform, and GSS providers can operate the platform. The platform provides the core capabilities discussed in Sect. 4.3. Stakeholders can use the platform through a self-service portal. Essentially, this model is an adaptation of PaaS cloud for GSS. For stakeholders using the platform, the key value proposition is to realize green business logics and domain-specific optimization methods at lower upfront cost with faster time to market.

4.5.3 Virtual Verticals

Virtual verticals are provided to domain-specific business services such as smart buildings or data centers, as illustrated in Fig. 4.8. It deals with configuring and deploying appropriate GSS for specific domains and operational environment in order to make the vertical application more efficient in its business and

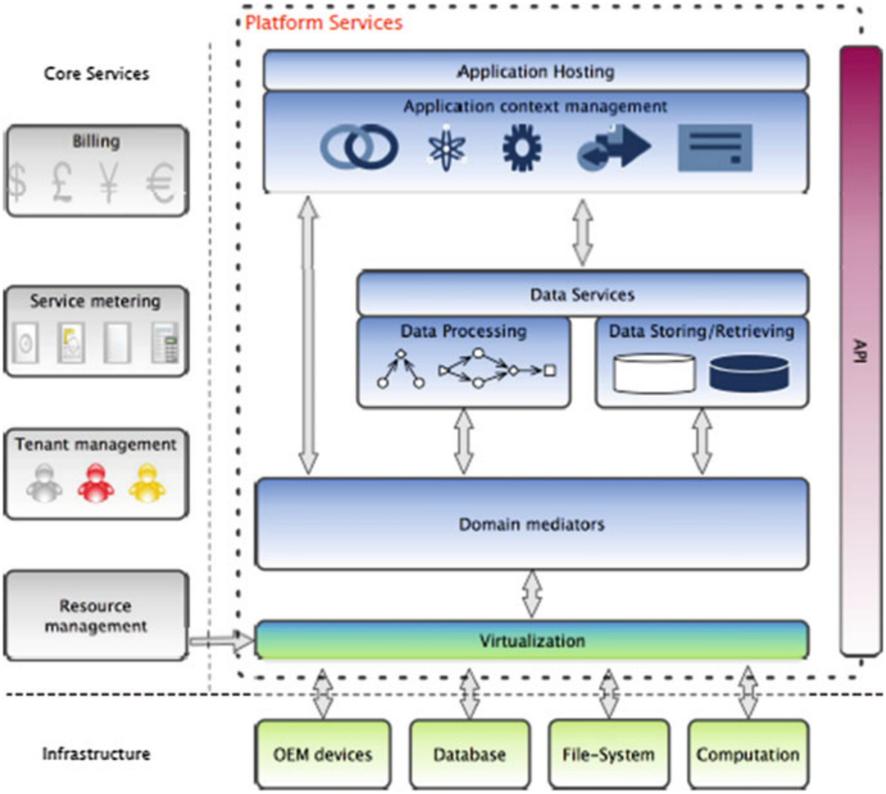

Fig. 4.7 Platform services

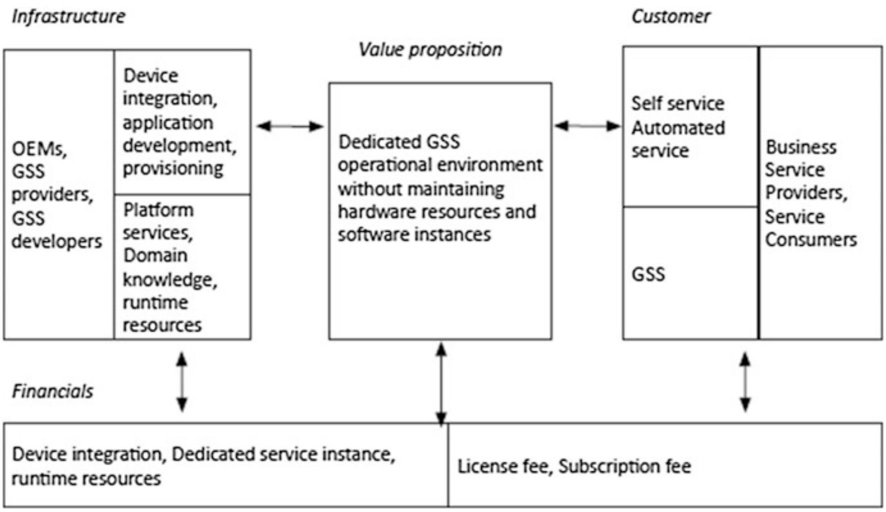

Fig. 4.8 Virtual verticals

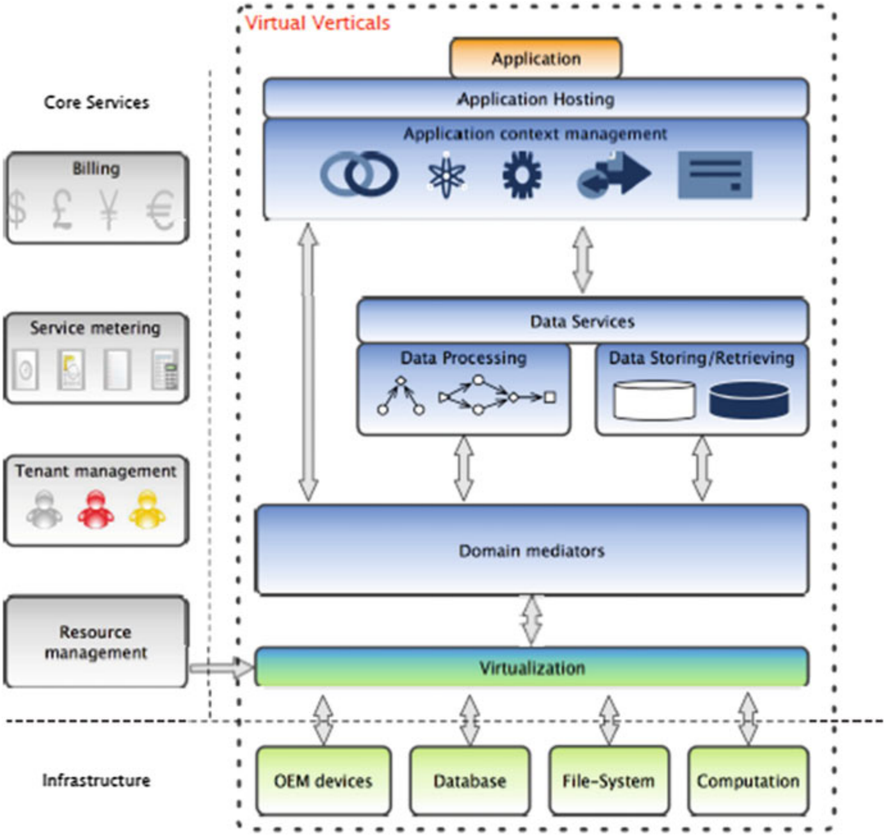

Fig. 4.9 Virtual verticals

technological context. 'Vertical' means that such applications are delivered as an end-to-end service coverage including physical devices, middleware, and applications for a certain physical environment, as illustrated in Fig. 4.9.

Virtual verticals can be realized by the collaborations of GSS providers, OEMs, and GSS developers. They integrate physical devices in the target environment and develop dedicated applications, such as light control or chiller management. Business service providers, who are the direct customers of virtual verticals, enjoy their dedicated GSS operational environments without committing computing resources or maintaining an IT infrastructure for GSS. Instead, they subscribe to virtual vertical services or license the software services for their operational environment. It is worth noting that the key difference between the proposed virtual vertical model and traditional physically isolated vertical model is the capability of sharing computing resources between verticals, which makes it easier for vertical applications to scale up.

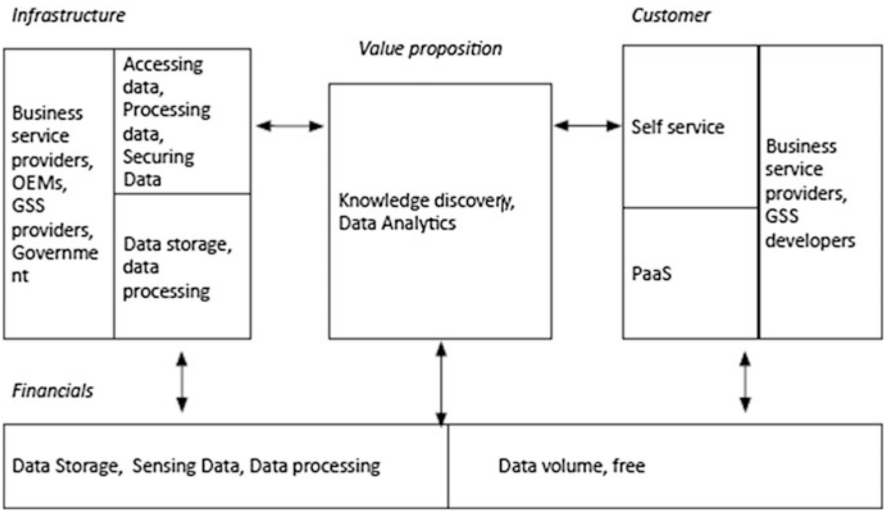

Fig. 4.10 Data services

4.5.4 Data Services

As the amount of data generated from business services is immense and still growing, the data service business model plays an important role in addressing data governance with a focus on data storage and processing aspects. Having the confidentiality of data properly managed and providing data to external experts or the public will further increase the utility of data. The data service business model is illustrated in Fig. 4.10. This model also deals with data concerns like privacy enforcement, up-to-dateness, data availability, and consistency in order to assure and improve data quality. Figure 4.11 illustrates the scope of data services. In principle, the service provider does not have to own the data sources, nor does it provide applications. It only concerns management and provisioning of data. Both real-time data and persistent data can be in the scope of service.

The data are owned by business service providers, OEMs, or governments. GSS providers offer the platform for them to open data access and establish business relationships with the customers who are interested in using the data for knowledge discovery or analytics. Business service providers can benefit from publicly available data to optimize their operations. GSS developers can create novel applications on the data or discover hidden knowledge in the data. The idea of providing data services have been realized through IoT platforms like Xively,[3] and more public data are being made available [7] for the developer community to discover their value.

[3] https://xively.com/

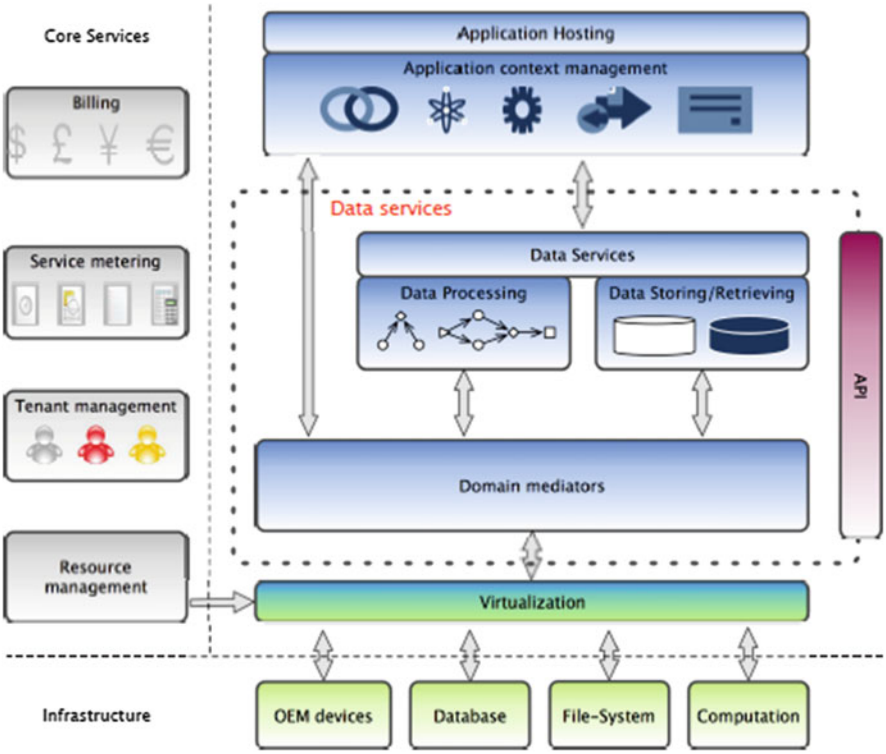

Fig. 4.11 Data services

4.5.5 *Third-Party Applications*

When the access to data, infrastructure, and core services is open to third-party developers, novel applications can be developed and provided as GSS, as illustrated in Fig. 4.12.

This business model can be built on top of either infrastructure services, platform services, or data services. In any case, this model is characterized by opening application development capabilities to third parties and, typically, providing application hosting service to them, as illustrated in Fig. 4.13. The involvement of third party in GSS can help establish a robust resource capacity planning with provider landscape. On GSS platform, the applications are offered online and used through subscription or licensing. Virtual verticals can be enhanced with third-party engagement to further extend the scope of applications. The range of applications can be broad, including optimized business processes, device optimization methods, or analytics. This model harnesses the creativity of the developer community and users to create various novel services.

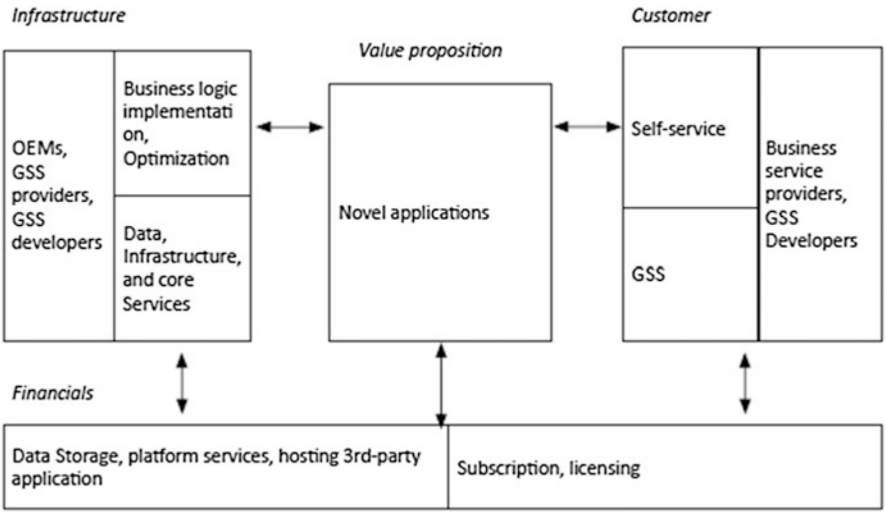

Fig. 4.12 Third-party applications

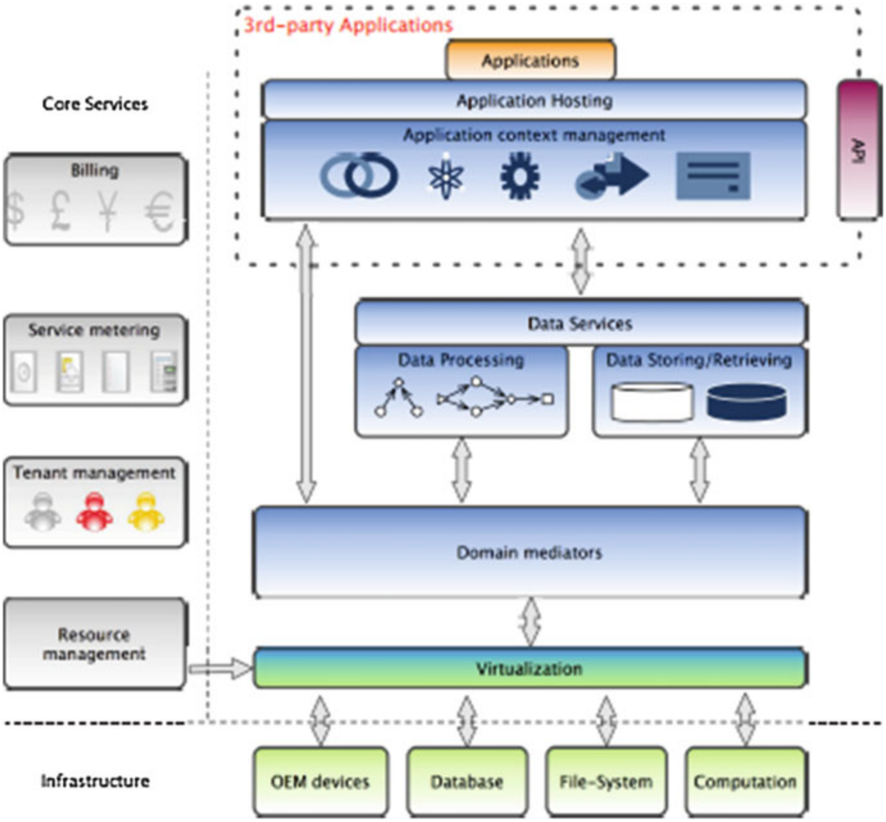

Fig. 4.13 Third-party applications

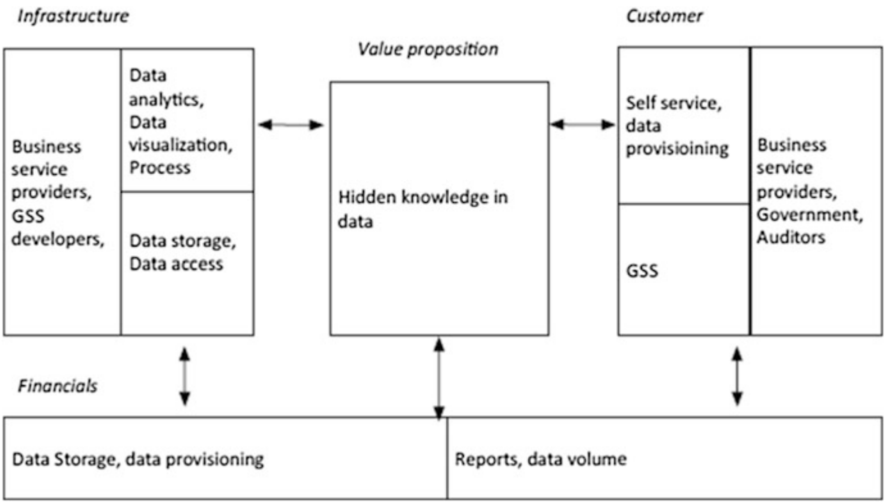

Fig. 4.14 Analytics as a service

4.5.6 Analytics as a Service

Analytics on the sustainability of business services are delivered by third-party analytics services [16], as illustrated in Fig. 4.14. Analytics as a service enables GSS consumers to leverage specialized analytics capabilities to identify previously unknown patterns and trends in their data.

This model is to a certain extent a type of third-party application. However, analytics as a service stresses that analytics are highly specialized tasks. External experts specialized in analytics are often required for conducting offline analysis. This model can also be employed by external auditing in order to assess the performance of GSS. Either data provided by the GSS platform or external data sources can be taken as input for analytics, as illustrated in Fig. 4.15. For the data to be effectively used in analytics, provisioning [10] (e.g., cleansing, normalization) is important but not necessarily a task of data providers since the provisioning can also be handled by data experts. Business service providers, governments, and auditors can all be interested in the results of analytics. They may purchase the reports or pay by the volume of data being analyzed.

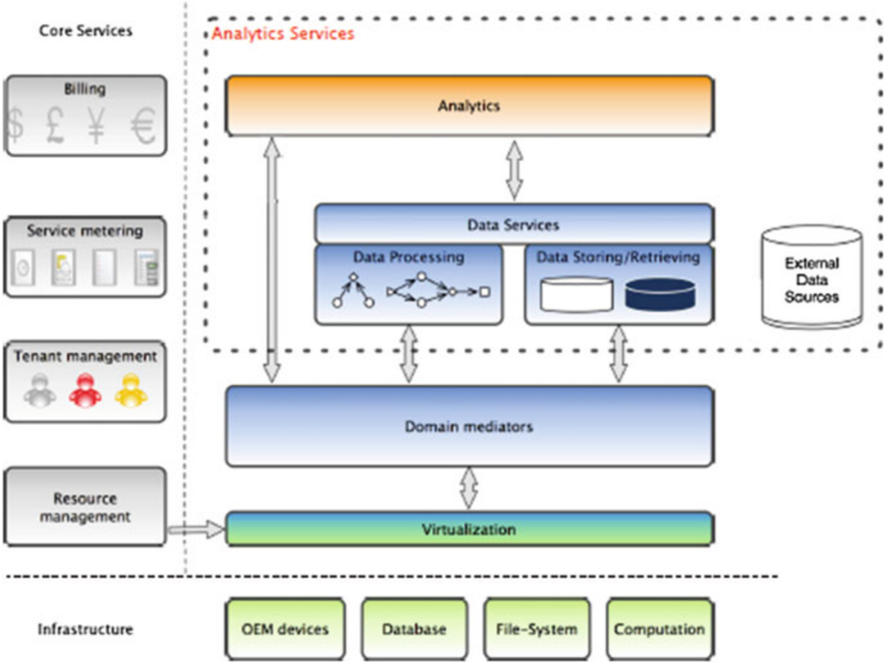

Fig. 4.15 Analytics as a service

4.6 Discussions

4.6.1 A Cloud Perspective to GSS

The rapidly growing popularity of cloud computing has made energy consumption of large data centers a trending topic in cloud research. Both of the two perspectives of green software research—energy efficiency of software and using software for energy efficiency—are applicable to cloud. On the one hand, the energy efficiency of cloud is affected by the operating systems, middleware, and applications. On the other hand, software tools can be built in order to manage the energy consumption of cloud. Mechanisms like resource scheduling and application workload prediction have been widely applied and are still improving.

This chapter relates cloud and green software in a new perspective: the service models of cloud. IaaS (infrastructure as a service), PaaS (platform as a service), and SaaS (software as a service) [12] are already familiar to researchers and IT professionals. While green software services are provided on the Internet, the models are referred to here in order to classify the services provided by each stakeholder. In fact, the business models proposed in this chapter can easily be mapped to the cloud service models. Infrastructure services that open up interfaces of OEM devices are similar to the IaaS cloud, which provides computing resources as services. The platform services and data services can be realized on a PaaS cloud by extending its capabilities of integrating with OEM devices. Third-party applications, virtual verticals, and analytics as a service are different forms of SaaS. Such mappings also serve as important reference for existing cloud service providers that plan to offer their services to the green software market.

4.6.2 Towards a Marketplace for GSS

The core vision of this chapter is to promote stakeholders in GSS to establish flexible business relationships through cloud-based service delivery models. As a natural result of the growing participation of stakeholders, a marketplace will emerge. In the marketplace, each type of service could be provided by multiple stakeholders, who offer different implementations for the same service with different QoS and price. Therefore, on top of the common platforms of GSS that stakeholders can collaborate at a technical level, there is a further need of supporting business activities including billing and SLA monitoring.

Although a cloud-based marketplace has been demonstrated by Amazon[4] and the concept of application store[5] is well accepted, a marketplace for GSS will face several new challenges. Evaluating and comparing GSS are hard since the effect of each service for each customer is tightly related to the specifics of target systems and their physical environments. Comprehensive metrics, especially domain-specific metrics, need to be incorporated or developed for objectively and accurately describing the services in the marketplace. These metrics will also help to establish effective monitoring mechanisms for GSS. A marketplace for GSS also means that the services should be delivered online. This challenges the deployment and provisioning mechanisms for GSS since business models such as infrastructure services and virtual verticals need to be coupled with devices in customers' physical environments. Automated or semiautomated methods, such as device integration or customer tools, need to be created to enable efficient and customized service delivery.

4.7 Conclusion

The research on the novel topic of green software is still at its infancy. Early research problems and technical solutions have been proposed, but wide adoption of green software is yet to happen. In this chapter, we tackled green software from the business perspective—trying to identify the requirements and the business

[4] https://aws.amazon.com/marketplace

[5] https://play.google.com/store

models that could benefit a wide range of stakeholders. We detailed the stakeholders and their interests in GSS. Based on this analysis, the high-level requirements of GSS were identified. Diverse business models were then proposed in order to motivate stakeholders to collaborate on the delivery of GSS. A cloud-based reference GSS architecture was presented with six variations to implement the corresponding business models. In the end, we discussed how the business models are related to cloud service models and the challenges of realizing a marketplace for GSS.

Acknowledgements The research leading to these results was supported by the Pacific Controls Cloud Computing Lab[6] (PC^{3}L), a joint lab between Pacific Controls LLC, Dubai, and the Distributed Systems Group at the Vienna University of Technology.

References

1. Atzori L, Iera A, Morabito G (2010) The internet of things: a survey. Comput Netw 54 (15):2787–2805. doi:10.1016/j.comnet.2010.05.010, URL http://dl.acm.org/citation.cfm?id=1862461.1862541
2. Bucherer E, Uckelmann D (2011) 10 Business models for the internet of things. Business 1–25
3. Cook J, Smith D, Meier A (2012) Coordinating fault detection, alarm management, and energy efficiency in a large corporate campus. In: 2012 ACEEE summer study on energy efficiency in buildings, pp 83–93
4. Dick M, Naumann S, Kuhn N (2010) A Model and Selected Instances of Green and Sustainable Software. In: Berleur J, Hercheui M, Hilty L (eds) What kind of information society? Governance, virtuality, surveillance, sustainability, resilience SE – 24, IFIP advances in information and communication technology, vol 328. Springer, Berlin, pp 248–259. doi:10.1007/978-3-642-15479-9 24
5. Dustdar S, Dorn C, Li F, Baresi L, Cabri G, Pautasso C, Zambonelli F (2010) A roadmap towards sustainable self-aware service systems. In: Proceedings of the 2010 ICSE workshop on software engineering for adaptive and self-managing systems – SEAMS '10. ACM, New York, pp 10–19. doi:10.1145/1808984.1808986. URL http://dl.acm.org/citation.cfm?id=1808984.1808986
6. Dustdar S, Li F, Truong HL, Sehic S, Nastic S, Qanbari S, Vogler M, Claesens M (2013) Green software services: from requirements to business models. In: 2nd international workshop on green and sustainable software (GREENS). IEEE, pp 1–7. doi:10.1109/GREENS.2013.6606415
7. (2012) Greenbiz: Hack City–Verge SF @Greenbuild Resources, GreenBiz Group Inc.
8. Guinard D, Trifa V, Karnouskos S, Spiess P, Savio D (2010) Interacting with the SOA-based internet of things: discovery, query, selection, and on-demand provisioning of web services. IEEE Trans Serv Comput 3(3):223–235. doi:10.1109/TSC.2010.3
9. James G, Cohen D, Dodier R, Platt G, Palmer D (2006) A deployed multi-agent framework for distributed energy applications. In: Proceedings of the fifth international joint conference on autonomous agents and multiagent systems – AAMAS '06. ACM, New York, p 676. doi:10.1145/1160633.1160752. URL http://dl.acm.org/citation.cfm?id=1160633.1160752

[6] http://pc3l.infosys.tuwien.ac.at/

10. Li F, Nastic S, Dustdar S (2012) Data quality observation in pervasive environments. In: The 10th IEEE/IFIP international conference on embedded and ubiquitous computing (EUC 2012), Paphos, Cyprus
11. Li F, Vögler M, Claeßens M, Dustdar S (2013) Efficient and scalable IoT service delivery on cloud. In: 6th IEEE international conference on cloud computing, (Cloud 2013), Industrial Track, Santa Clara, CA
12. Liu F, Tong J, Mao J, Bohn R, Messina J, Badger L, Leaf D (2011) NIST cloud computing reference architecture. NIST Special Publication 500, 292
13. Loutas N, Kamateri E, Tarabanis K (2011) A semantic interoperability framework for cloud platform as a service. In: 2011 IEEE third international conference on cloud computing technology and science. IEEE, pp 280–287. doi:0.1109/CloudCom.2011.45
14. OASIS: Open Building Information Exchange (oBIX) (2012). URL https://www.oasisopen.org/committees/tchome.php?wgabbrev=obix
15. Steigerwald B, Agrawal A (2011) Developing green software. Tech. rep., Intel
16. Sun X, Gao B, Fan L, An W (2012) A cost-effective approach to delivering analytics as a service. In: 2012 IEEE 19th international conference on web services. IEEE, pp 512–519. doi:10.1109/ICWS.2012.79
17. Zachhuber D, Doppler J, Ferscha A, Klein C, Mitic J (2008) Simulating the potential savings of implicit energy management on a city scale. In: 2008 12th IEEE/ACM international symposium on distributed simulation and real-time applications. IEEE, pp 207–216. doi:10.1109/DS-RT.2008.26