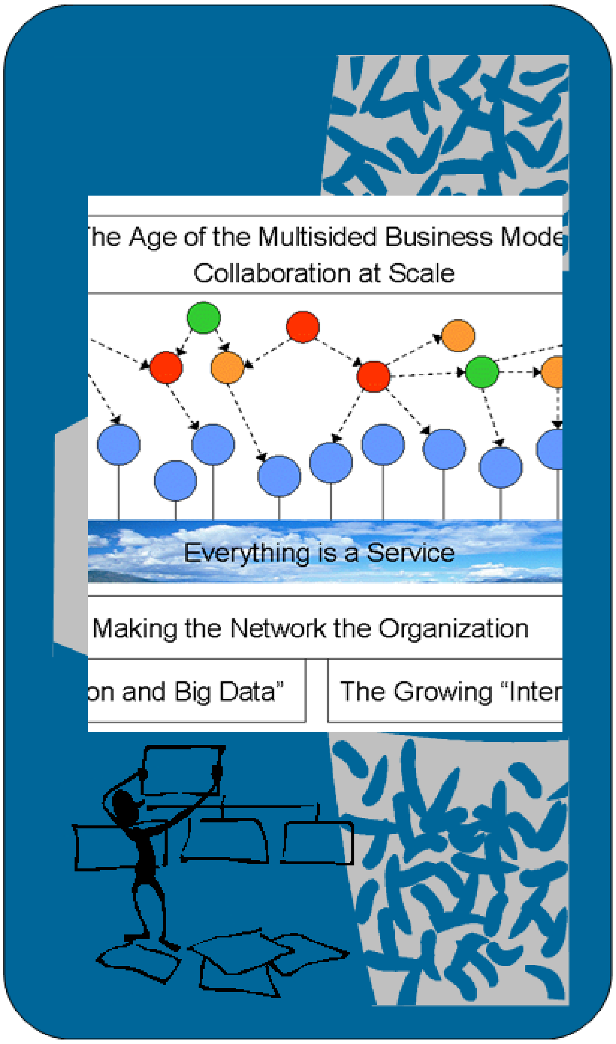
CBDI Journal

January 2011



Practice Guide . . . .

Making Sense of Cloud Computing

The terms virtualization, utility computing and Cloud computing are often used interchangeably which can be very confusing. This report aims to provide clarification - to identify the similarities and differences in those characteristics, and provide a framework in which organizations can decide which capabilities they require in specific situations – as it is unlikely that one model alone will suit all requirements.

By Lawrence Wilkes

Originally published in the January 2011 edition of the CBDI Journal

Independent Guidance *for* Service Architecture and Engineering

Making Sense of Cloud Computing

The terms virtualization, utility computing and Cloud computing are often used interchangeably which can be very confusing. This report aims to provide clarification - to identify the similarities and differences in those characteristics, and provide a framework in which organizations can decide which capabilities they require in specific situations – as it is unlikely that one model alone will suit all their requirements.

By Lawrence Wilkes

Introduction

A visitor from outer space would be forgiven for thinking that virtualization, utility computing and Cloud computing are different capabilities. But most of us understand that many product and service offerings use the terms rather casually and the trend is to assume they are all describing the same thing, where virtualization is synonymous with utility computing which is synonymous with Cloud. Some might say these are just steps in the evolutionary process – where utility has simply evolved into Cloud.

So, are they the same or different? Can the terms be used interchangeably, or are there clear distinctions between them?

The answer is not straightforward. There are clearly some common, overlapping characteristics that allow the terms to be used interchangeably. But at the same time there are other characteristics that enable them to be distinguished from each other.

This report aims to provide clarification. To identify the similarities and differences in those characteristics, and provide a framework in which organizations can decide which capabilities they require in specific situations – as it is unlikely that one model alone will suit all their requirements.

Capabilities

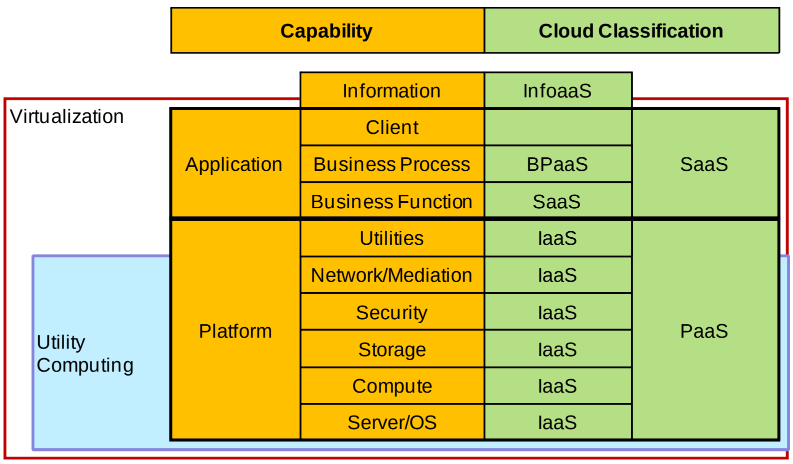


Figure 1 – Virtualization, Utility and Cloud Computing Classification

What capabilities are provided by utility and Cloud computing, and where does virtualization fit in? Figure 1 shows a layered 'stack' of IT capabilities and the relationships to Cloud Computing and Utility Computing capabilities.

Utility Computing[1] is the provision of infrastructure capabilities such as the server and operating system, compute capability, and storage on a pay as you go (PAYG) basis. In many cases utility computing utilizes server virtualization.

As Figure 1 illustrates, Cloud Computing equates to Utility Computing in terms of the capabilities provided. As mentioned earlier, some might see this as a trend with Cloud terminology simply replacing utility. But Cloud Computing can provide these capabilities on a finer-grained basis that was typically the case with Utility Computing. Cloud also raises the levels of abstraction, providing a higher level of interaction for the consumer via utilities and a management layer that can make them easier to use than raw infrastructure, and provides complete virtualization of resources, usually with self-service provisioning on a multi-tenancy basis. In Cloud terminology these infrastructure resources are provided as Infrastructure as a Service (IaaS).

Note we are happy to recommend the NIST definitions of Cloud Computing, which we covered in a previous report[2].

Security capabilities such as authentication and identification, and networking and mediation capabilities such as message delivery, routing and transformation can also be seen as part of Utility Computing and also classified as IaaS. These capabilities can be combined together into a 'platform' and classified as Platform as a Service (PaaS)[3]. PaaS provides the IaaS capabilities via a higher layer of utilities, and not directly. Hence the PaaS can be said to encapsulate the IaaS capabilities.

Layered on top of the platform are the applications or 'business' capabilities. These might be broken down into client, business process and business function capabilities, but more often than not are packaged together as an application. Cloud providers don't normally distinguish between them, and hence they are typically combined as Software as a Service (SaaS), regardless of the granularity or separation of layers.

The exception to this is Business Process as a Service (BPaaS). This is an emerging layer in which process assembly is offered as a service which allows the consumer to orchestrate services from disparate sources. BPaaS is therefore a specialization of SaaS.

PaaS is also used to characterize a suite (hence 'platform') of application functions that are exposed as service interfaces. For example Google Apps or Salesforce.com provide PaaS capabilities in addition to their core SaaS capability, to expose specific interfaces as a 'platform' on which consumers can assemble a custom application.

The use of PaaS and SaaS may be abused, and we advise that you interpret the capability offered on the basis of level of access and control - that is SaaS delivers applications with no opportunity to manage or control the underlying infrastructure or application capabilities. In contrast PaaS provides control over the deployed application and possibly the hosting environment.

---

[1] http://en.wikipedia.org/wiki/Utility_computing
[2] CBDI Journal. April 2010. Service Portfolio Planning and Architecture for Cloud Services (pdf)
[3] http://en.wikipedia.org/wiki/Platform_as_a_service

Their use may also be seen as contextual. That is, as an end user of Google Apps the context would be SaaS, but as a developer using Google Apps as a platform on which to implement a new application, the context would be PaaS.

Use of a higher layer in the stack usually means the lower layers are inherited. Typically there is no separation of supplier down through the layers from the consumer perspective. That is, whilst the SaaS provider may in turn use a different IaaS or PaaS provider, this is transparent to the SaaS consumer. The exception here is BPaaS which we think qualifies it as a specialization precisely because it allows orchestration of disparate capabilities.

The term virtualization spans all of these capability layers. Cloud computing environments virtualize the capability at its highest level of abstraction in order to make the underlying resources location independent, more scalable, resilient, and so forth. In the Cloud context, virtualization is a generic term and it may be expected that server virtualization and server clustering technologies provided by platform vendors will be used to provision individual capabilities and resources supporting the Cloud service.

However, the same virtualization technologies may be employed wholly within an organization's own private data centre, and so are not necessarily used to support Cloud behaviors (multi-tenancy etc.). Equally in house virtualization which is deployed with multi-tenant elasticity may reasonably be relabeled as a private Cloud.

Enabling virtualization is usually a necessary first step towards utilizing both utility and Cloud computing. If resources cannot be virtualized, how can they be deployed to the Cloud?

Characteristics

Most large enterprises will be commencing use of Cloud, but inevitably they will have a spectrum or continuum of capabilities deployed which span server virtualization, utility and Cloud computing. For most enterprises an early objective of Cloud initiatives will be to rationalize and modernize existing assets. Table 1 therefore provides a set of nomenclature which may assist in documenting and classifying existing and planned technology assets.

As well as considering virtualization, utility and Cloud, I have also included the notion of an outsourced data centre as a further option that many organizations will use. How does this compare to Cloud and Utility? I have also distinguished between public and private Clouds. A private Cloud will provide both utility and application capabilities. As Table 1 shows, in some respects it might be hard to distinguish a private Cloud from a traditional data centre, or the outsourced data center if administered by a 3rd party. However, if it exhibits distinguishing Cloud features such as metering usage on a PAYG basis, providing capability on a fine-grained basis through an easier to administer interface, then it can be identified as a different model. You could further distinguish between Private PaaS/IaaS and Private SaaS provisions, but I don't think that adds much to the model, as the only difference is the capability offered.

But beware provider organizations that may have merely relabeled these deployed assets with a more fashionable term!

Table 1 – Characteristics of virtualization, utility and Cloud computing

	Virtualization Technology	Outsourced Data Centre	Utility Computing	Cloud Computing Private	Cloud Computing (PaaS/IaaS) Public	Cloud Computing (SaaS) Public
Virtualization of Capabilities	Virtualizes a server, and/or a cluster of servers	Virtualizes a server, and/or a cluster of servers	Virtualizes a server, and/or a cluster of servers	Virtualizes a capability	Virtualizes a capability	Virtualizes an application or software service
Location	On-premise	Off-premise	On or off-premise	On or off-premise	Off-premise	Off-premise
Location Independence	Fixed	Fixed	Fixed	Fixed	Independent	Independent
Sharing	None	None Multi-Instance	Multi-Instance	None	Multi-tenancy	Multi-Instance Multi-tenancy
Exclusivity	Private	Private	Private or Public	Private	Public	Public
Operation	In-house 3rd party	3rd party	3rd party	In-house Specialist Provider	Specialist Provider	Specialist Provider
Capability Layer	Utility Application	Utility Application	Utility	Utility Application	Utility	Application

	Virtualization Technology	Outsourced Data Centre	Utility Computing	Cloud Computing Private	Cloud Computing (PaaS/IaaS) Public	Cloud Computing (SaaS) Public
Scalability/ Elasticity	Finite within resource limits	Finite within resource limits	Elastic	Finite within resource limits	Elastic	Elastic
Responsiveness	Autonomic within finite resource On-request	Autonomic within finite resource On-request	Autonomic addition of resource units	Autonomic within finite resource On-request	Autonomic increment of capacity	Autonomic increment of capacity
Unit of use	Server(s)	Server(s)	Server(s) Storage	Platform Compute Storage Application	Platform Compute Storage	Application Software Service
Business Model	Investment	Long-term Contract	Unit/Period Rental	Investment Long-term Contract	PAYG	PAYG
Unit of metering	N/A	N/A	Per unit Monthly/ Annual	Consumers are metered per usage	Per usage	Per usage

Taking each of those characteristics in turn.

Virtualization of capability: As already noted, each of these virtualize the capability provided, whether it is the server or application, or some component of either.

Location: Can be seen as either on- or off-premise. However, it is often the case that the data centers of many large companies are hardly 'on-premise' anyway, and located in a separate custom facility.

Location Independence: What is more important perhaps is how fixed that location is. Utility and Cloud computing offer the potential advantage of location independence that virtualization technology on its own does not, nor typically do outsourced data centers, at least not the very dynamic way that is expected of utility and Cloud usage models.

Sharing: Utility and public Cloud computing capabilities are normally provided on a shared basis. This is how the economies of scale are achieved. Sharing might be achieved in one of two ways

- Multi-tenancy. A single instance of some capability is shared by many consumers. Many Cloud computing capabilities operate on this basis.
- Multi-instance. Each consumer each has their own unique instance of the capability. It is the collection or 'cluster' of instances that makes it utility or Cloud-like in its provision.

However, it is not always clear to the consumer which of these two models is being applied. The concept of sharing may also be logical or physical – that is consumers may perceive they have a unique instance, but in reality they are just separated from other consumers by permissions and in fact share the same instance of a database for example. Further, the model may change at different layers in the stack. For example, a multi-instance application may run on a multi-tenancy server.

In many situations, it doesn't really matter to the consumer which is being applied. But clearly in others it will, where security and reliability are perhaps paramount.

An outsourced data centre may be run on a multi-instance basis in which it is used by several different organizations, each provided with their own unique set of resources. However these might be run as a server 'farm' to provide scalability for the different organizations. But equally there may be a distinct set of 'ring fenced' resources, in which case the only thing shared is the facility itself.

Exclusivity: This is not quite the same as sharing in that I use this here to distinguish between whether the capabilities are private and exclusive to the user (even when operated by a 3rd party), or public in the sense that anyone can come along and take out a contract to utilize them, as is often the nature of Cloud computing. The capacity of a private capability should be very clear, and finite.

Whereas with a public capability you have no way of knowing how many other consumers are sharing the instance and what their peak loads might be, or whether the peaks might coincide. Not that this should present a problem for the well run utility or Cloud provider.

Operation: Utility and Cloud computing capabilities are operated by 3rd parties. The consumer might be able to tweak operational parameters through a control panel. Though even fully owned, on-premise, private capabilities might be managed by a 3rd party on a contract basis.

Capability Layer: Only Utility computing and Cloud IaaS are focused on utility capability provision. Virtualization technology and data centers support both application and utility virtualization.

Scalability/Elasticity: This should probably be a sliding scale rather than a binary choice of finite or elastic. With in-house or a traditional data center type environments, though virtualized, there is a still a finite limit to the resources acquired or contracted for. Ultimately all resources are finite, but elastic Clouds hold the promise that they will be able to provision more than any one organization could consume. Moreover, they scale down as well as up (elasticity), and payment is only for the resources currently in use. Whereas in-house or with a data center there may be less flexibility – unused resources are still paid for.

Responsiveness: Regardless of whether the resources are in-house, in a data centre or in the Cloud, where virtualization technology is used, then responsiveness should be autonomic and instant – "on-demand" if you wish. However, in in-house or data center scenarios, going beyond the finite limit of resources as discussed under scalability, may then require additional resources to be provisioned "on-request" – requiring action by the consumer. That said, even with utility and Cloud models, consumers may have set upper limits to the resources they contract for that can be exceeded only "on-request".

Unit of Use: The unit of use is typically also the unit of contract. In virtualization, outsourced data centers and utility computing this is usually a server (either empty, or ready purposed with OS, database, etc as a 'platform'). In Cloud computing it is either a finer-grained capability – such as a component running on the platform – or is an application.

Business Model: Is a high level perspective of how the capabilities are funded, ranging from up-front investment in the capability to a PAYG model.

Unit of Metering: Or unit of pricing, is primarily concerned with utility and Cloud computing, which may be paid for on a term-basis or a fine-grained per-usage (transaction) basis

The internal accounting for the use of virtualization technology might also be on a metered basis. But the acquisition of the capability in the first place isn't normally paid for on this basis – rather it is leased or purchased. That said, there are some customer programs from hardware vendors who provide servers on a PAYG basis.

Of course the boundaries are not always clear. I am sure some readers will be able to find some examples that appear to blur the lines in Table 1. However, it should still nevertheless help organizations to assess the type of capability they need by selecting the characteristics they desire.

Decision Criteria

Table 2 looks at these and other characteristics from a slightly different angle to assist the consumer to make decisions based up them.

See also table 1 'Cloud Service Usage Decisions' in a previous report[1]

Factor	Considerations	Notes
Operating Cost	What are the ongoing costs of provision?	PAYG does not necessarily mean cheaper over the long term The direct cost may not be on a like-for-like basis, but can be calculated.
Investment	What is the level of investment required to provision the capability? How is the ROI calculated? How is the investment recovered, or factored into operating costs?	Lifetime costs including investment should be understood. However, where the lifetime is uncertain (e.g. new ventures), the decision to use a PAYG basis might make more sense even if more expensive on a daily basis.
Location	Where is the capability located? Is the location fixed?	This may not be as simple as off- or on-premise, but may also be a geographical/ political issue.
Privacy, Security	How secure is the capability? How private is the capability, or the data flowing through it?	This should be seen from both a physical, as well as a network/software perspective.
Reliability	What is the risk of disruption in supply?	Should be mitigated though provision of backup capabilities, regardless of which of the computing models in table 1 is used. The consumer must ensure adequate provision is made either by them or the supplier.
Mission Criticality	How critical to the enterprise is the capability (or its application)?	Privacy, security, and reliability might be converged in a decision on mission criticality.
Business Category	Is the business capability core or context?	Allied to mission criticality is the notion of whether the capability is the core purpose of the business, or just supporting
Separation of capability	How separated are the different layers of capability? Is the separation visible to the consumer?	Unforeseen dependencies across layers. Use of 'unacceptable' providers at a lower layer in the stack. (e.g. see location)

[1] Service Portfolio Planning and Architecture for Cloud Services, CBDI Journal September 2010

Factor	Considerations	Notes
Portability	How portable is the capability between suppliers? Can the supplier easily be switched?	Other than in its most basic form - an empty un-purposed server, or a common platform - there is little portability between the capabilities on offer, especially at the application layer.
Scalability/ Elasticity	How elastic are the resources? Is payment only for the resources actually used? Are the resources finite?	Finite resources will be a constraint on scalability and coping with changes in demand. Elasticity is strongly tied to responsiveness, in that if payment is only to be for the resources actually used, then elasticity must be instant and autonomic, not on-request.

Table 2 – Decision Criteria

Risk Assessments

Security and Reliability

Most often, risk assessment with utility and Cloud computing is primarily focused on disruption to supply and security, and rightly so given its off-premise, shared, and public characteristics. I don't intend to go deeper into this given that much has been written already.

I could have added security and reliability to the list of characteristics. However, in this regard I am not fully persuaded that utility and Cloud are necessarily any less risky than on-premise. Unless it is totally disconnected from the outside world, what makes an in-house data centre any more reliable or secure than that provided by a well run utility computing centre? More so given that the data centers of many large companies are hardly on-premise anyway, and may well be administered by 3rd parties or completely outsourced to an outsourced data center as previously discussed. And can an SME be expected to provide the level of reliability and security on-premise when compared to a highly optimized utility computing data centre?

Co-Location of Capability Layers

A difference between traditional utilities, such as power supply, and utility and Cloud computing is that whilst the traditional utility capability might be produced 'off-premise', its application is very much 'on-premise'. That is, the electricity might be produced at a power plant, but it powers the equipment such as my TV (the application) in my home.

Whereas, in utility and Cloud computing models then both the utility and the application capabilities are both 'off-premise'. That is, not only is the raw utility – the infrastructure capability – located 'off-premise', but so are the software applications and data hosted on it.

Consequently, disruption of utility computing supply also means a disruption in its application. Whereas with traditional utility, if the power fails for example, you might still be able to switch to an alternative without disruption to its application – via standby generators, batteries, etc.

How this risk compares to having one or both on-premise requires assessment. Clearly as mentioned in table 2, consumers must ensure adequate backup of both utility and application is made either by them or the supplier.

Portability and Separation of Capability

In terms of portability and separation of concerns, then the provision of empty un-purposed servers, or a common platform, by a utility computing supplier can be seen as equal to the raw products provided by traditional utilities. However, many of the Cloud IaaS suppliers such as Amazon Elastic Cloud, Google App Engine or Microsoft Azure, whilst providing a notional 'utility' do so at a higher level of abstraction than empty servers. Whilst they each provide similar 'compute' or 'storage' utilities, there is no real compatibility between them and hence the ability to change suppliers is greatly reduced in comparison to traditional utilities.

The resulting lock-in can be a constraint on agility (though there are open source initiatives seeking to provide a layer of interoperability across them, as well as nascent standards initiatives). It also increases risk as a disruption in supply can have a double impact if there is no directly compatible alternative that can quickly be established.

Figure 2 illustrates that the greatest risk is when all these criteria are taken in consideration together.

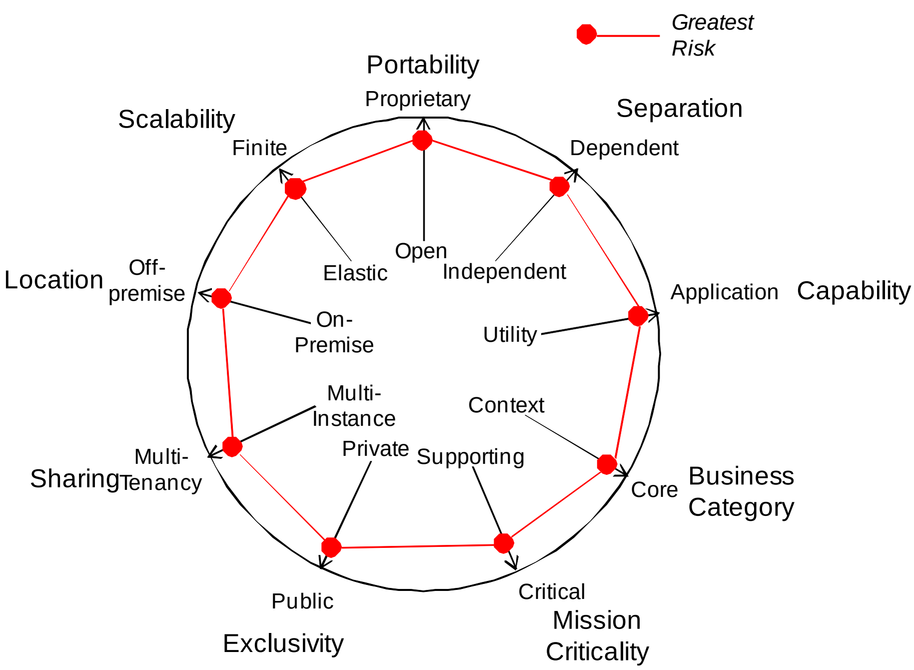


Figure 2 – Risk Assessment Spider Chart

Conclusions

It's too simplistic to imagine that a single, central, Cloud computing pattern will dominate computing in the future in the same way as power and water utilities did for the last century. Rather we should expect that the much greater complexity and in particular the wide range of levels of capability abstraction involved, will lead to a very wide range of delivery models in future.

Whilst IT is being encouraged to evolve towards the traditional utility model such as electricity or water which are exemplars of "pay per use" and shared resources, it is interesting to note that these same utilities are evolving away from hub and spoke model towards micro production methods such as solar, wind, or local drilled wells. Moreover, consumers are installing their own capability. This trend is occurring of course because of more advanced technologies that reduce the economic advantages of large scale production or reduce risk with self-sufficiency and lower dependence upon shared resources.

Related Everware-CBDI Report - Extract

Service Portfolio Planning and Architecture for Cloud Services

Abstract: Cloud Computing is concerned with deployment, but introduction of Cloud Services cannot be a purely technical deployment matter. There are numerous considerations that may impact on all the Stakeholder Views. In this report, we show how the CBDI-SAE approach can be used and extended to architect for Cloud Services. We extend our current guidance with new and refined classification systems, diagrams, policy types and techniques designed to promote visibility and good governance over Service Portfolio Planning activities and Cloud Services provisioning.

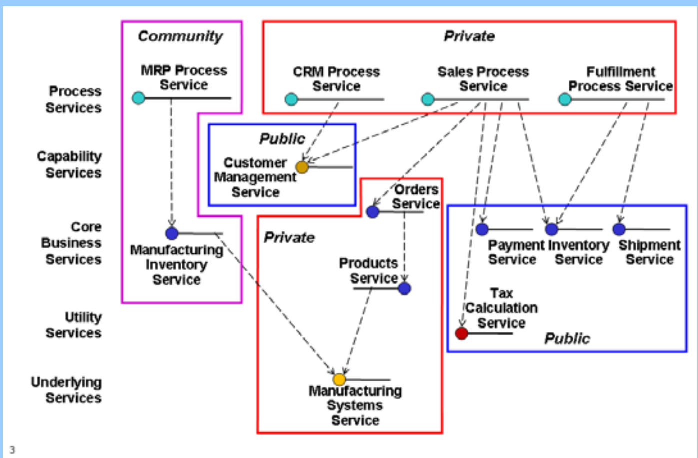


Figure 5 - Specification Architecture – Usage Pattern Classification

In Figure 5, we now show the classification of the Services according to the usage patterns that we outlined . . . In this example, as there is a many-to-many relationship between suppliers and manufacturers in the same industry they collaborate as an ecosystem to more efficiently match demand. Consequently the Services provided are open to any trusted participant in the community, and hence classified as Community. The 3rd party Cloud Services . . . are classified as Public, whilst the Orders, Products and Manufacturing Services are Private.

To read the full report: http://everware-cbdi.com/index.php?cID=32&cType=document
Available on registration or member sign-in

About CBDI

CBDI Forum is the Everware-CBDI research capability and portal providing independent guidance on best practice in service oriented architecture and application modernization.

Working with F5000 enterprises and governments the CBDI Research Team is progressively developing structured methodology and reference architecture for all aspects of SOA.

CBDI Products

The CBDI Journal is freely available to registered members. Published quarterly, it provides in-depth treatment of key practice issues for all roles and disciplines involved in planning, architecting, managing and delivering business solutions.

Visit www.cbdiforum.com to register.

Platinum subscription – A CBDI Forum subscription provides an enterprise or individual with access to the CBDI-SAE Knowledgebase for SOA delivering ongoing practice research, guidance materials, eLearning, tools, templates and other resources.

Everware-CBDI Services

At Everware-CBDI we enable large enterprises and governments to become more agile by modernizing their business systems. We have repeatable processes, resources, tools and knowledge-based products that enable enterprises to transform their current applications in an efficient, low risk manner, into an optimized service-based solutions portfolio that supports continuous, rapid and low cost evolution. Our consulting services range from providing practices and independent governance to architecture development, solution delivery and service engineering.

Contact

To find out more, and to discuss your requirements visit www.everware-cbdi.com or call

USA and Americas: 703-246-0000 or 888-383-7927 (USA)

Europe, Middle East, Africa, Asia, and Australasia: Telephone: +353 (0)28 38073 (Ireland)

www.everware-cbdi.com