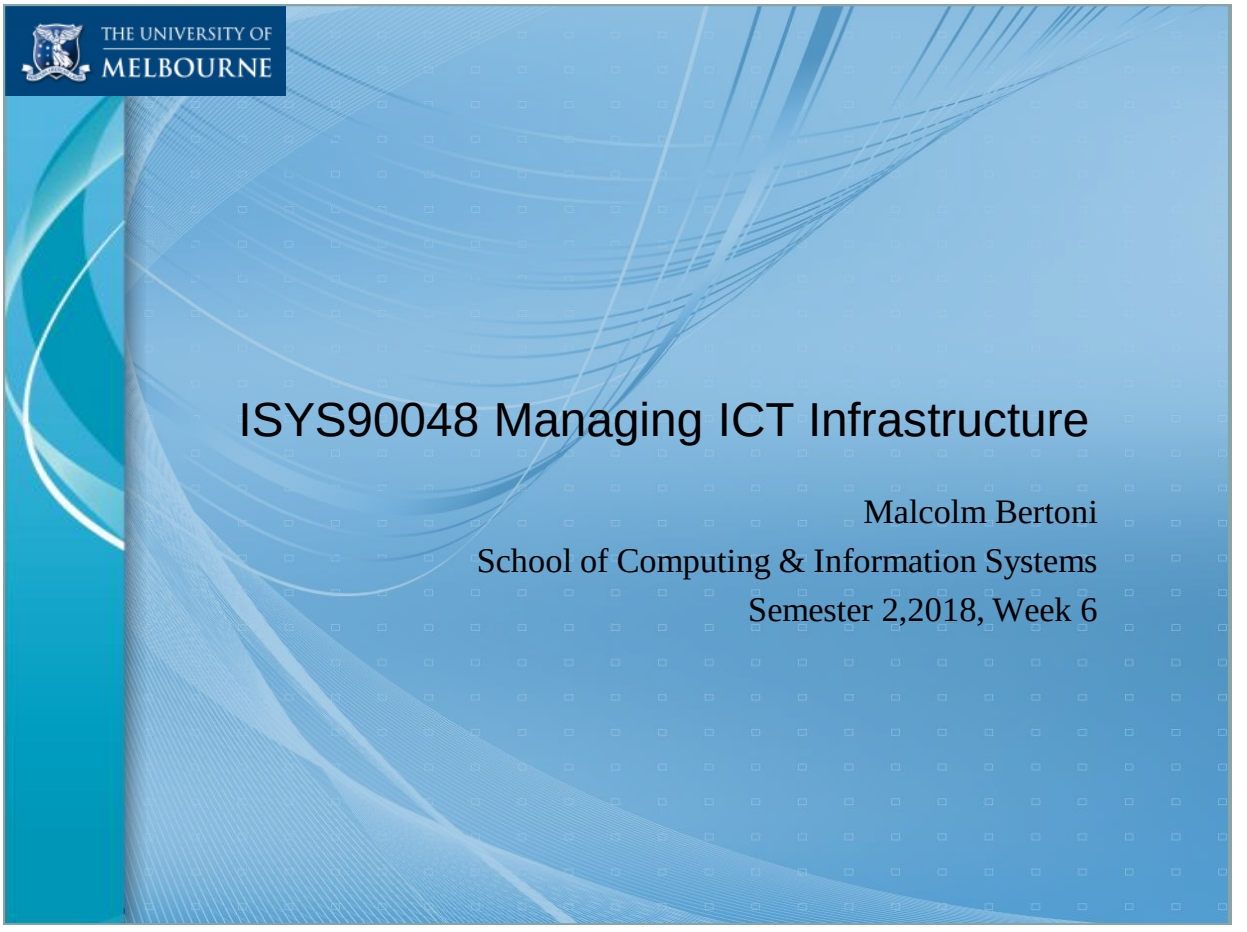

THE UNIVERSITY OF
MELBOURNE
ISYS90048 Managing ICT Infrastructure
Malcolm Bertoni
School of Computing & Information Systems
Semester 2,2018, Week 6

THE UNIVERSITY OF MELBOURNE

Teaching Session 06

- Demonstrate an understanding of the role and importance of ICT service provision and acquisition in the responsibilities of an ICT infrastructure manager
  - Current approaches to infrastructure architecture design – systems and objects
  - Services and servitization
  - Aspects of systems

THE UNIVERSITY OF MELBOURNE

Systems

System: A system is a collection of elements or components that are organised and work together for a common purpose

- Abstraction of systems is contained in all aspects of ICT infrastructure design
- Systems Architecture: As ICT infrastructure architects and managers we are expected to be systemic in our view of modelling the world and of designing processes, management practices and delivering outcomes
- Systems are a modelling artefact – they are an abstraction by which we seek to understand the real world, and by which we design components to match that real world view and process information
- Used by software engineers and systems architects
  - Use systems to model real world objects, with well defined purposed, scope, boundaries and interfaces

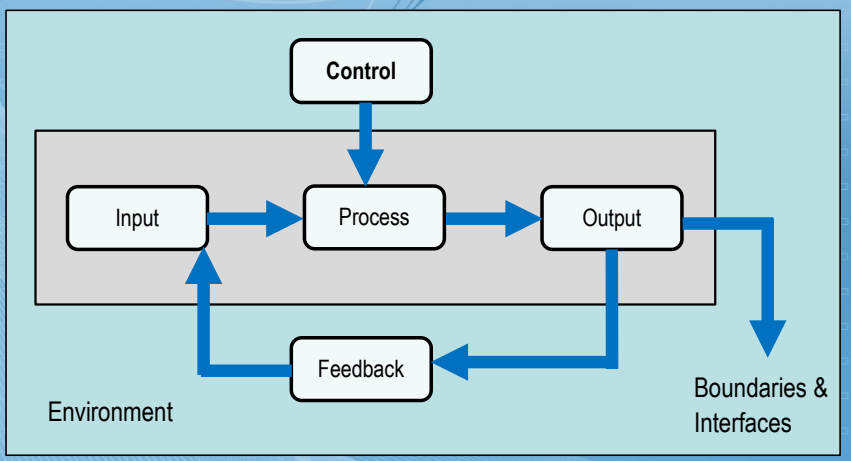
Basic Model of a System

- Can compose & decompose systems into systems or subsystems
- Open systems interact with their environment & other systems
- Closed systems do not interact with their environment
- Some systems may be transient – appear, operate, then disappear
- Others are persistent – last indefinitely
- Systems have a purpose, a function and a composition
- A system has structure, it contains parts that are directly or indirectly related to each other
- Systems can be well defined – they have a scope, purpose and boundary
- A system has behaviour, it exhibits processes that fulfil its function or purpose
- Systems interact via their systems interface by the exchange of messages – these messages may be requests from one system to another, or a response to a request, or an informative message that is communicating some status from one system to another

Ref: Golden, B *What is Systems Architecture,* available at http://www.lix.polytechnique.fr/~golden/systems_architecture.html#principles

Aspects of Systems

- The System Interface defines that nature of the messages that a system can receive and the nature of their responses to those messages
- A System State is the status of all of its internal settings (variables/data) at a given time – a system action will normally result in a change of its state



Systems from an ICT Perspective

- Systems need to be designed to meet internally-focussed requirements (smart systems), rather than being designed to meet externally-focussed requirements (users needs, composability of systems)
- ICT Systems must have interoperability - make correct requests to the right systems, which must process their requests correctly and their responses be handled correctly
  - This requires correct system interfaces and correct systems semantics
  - The complexity grows rapidly as the number of systems increases
- Scalability can only be achieved by the use of increasing complex systems, which are harder to understand and to achieve interoperability
- ICT infrastructure requirements are never static and will lead to the need to frequently redesign and re-implement the component systems and re-establish interoperability

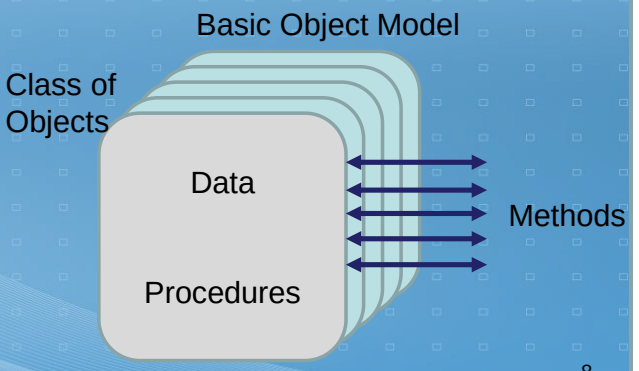
THE UNIVERSITY OF MELBOURNE

Systems - Summary

- Taking a systems view – seeing the world as systems – is so familiar that we rarely question it
- Systems provide a convenient abstraction of the real word because they can be well defined, are discrete, and can be translated into information system which can be implemented in ICT – this is requirements engineering + software/systems engineering
- Interoperability requires that systems are well defined in their requirement specification, and in their implementation
- Scalability of systems requires a “systems of systems” approach – building more complex systems by using compositions of sub-systems
- A systems engineering frequently focuses on the technological aspects of the systems, since these are the requirements that can be expressed formally and tested, and neglects the human aspects of information systems

Object-Oriented Design

- An object contains meaningful data and procedures grouped together to represent an entity
- Objects are abstractions of real-world or system entities
- Object oriented design combines data and methods together into a cohesive whole
- System functionality is expressed in terms of object services
- Objects communicate by message passing (providing a service)
- An object-oriented program is described by the interaction of these objects
- OOD aims to design software such that it is broken up into manageable chunks called classes



Objects from an ICT perspective

- Objects are similar to Systems:
  - Boundaries (Encapsulation)
  - An Interface, input, procedure (process), output/deliverable
  - Polymorphism (have different functions and different data types)
  - A purpose
  - Identity – each object can be identified and treated as a distinct entity
- But objects are also reusable
- Advantages of Objects:
  - Easier maintenance. Stand-alone entities
  - Reusable
  - Able to mimic real-world entities as system objects
- Limitations
  - Objects do not solve the issue of dealing with increasingly complex systems
  - OO design is essentially a software engineering methodology, rather than a systems architecture design methodology

THE UNIVERSITY OF MELBOURNE

Services

- A service provides a discrete/disconnected business function
- Its job is to ensure that the business functionality is applied consistently, returns predictable results, and operates within the quality of service required
- Services can be independently evolved, moved, scaled even in runtime
- Self-contained services communicate with each other
- Software developers reuse these services in combinations to assemble other applications as needed
- Services are kept simple: complexity is the enemy of reliability and flexibility
- Platform independent
- Example of a service: online form that, when completed, generates a file with data for use in another service

Services cont

- Four properties:
  1. It logically represents a business activity with a specified outcome
  2. It is self-contained
  3. It is a black box for its consumers (black box: users of the system have no knowledge of its internal workings)
  4. It may consist of other underlying services
- The services are supported by an infrastructure (SOA) that, together with the services, improves information flow within the enterprise and between the enterprise and external enterprises
- Provided over a network, middleware and enterprise service bus (ESB)
- Requires the design & management role of a Service Architect

Ref: Erl, T (2005) *Service-Oriented Architecture: Concepts, Technology, and Design* Prentice-Hall

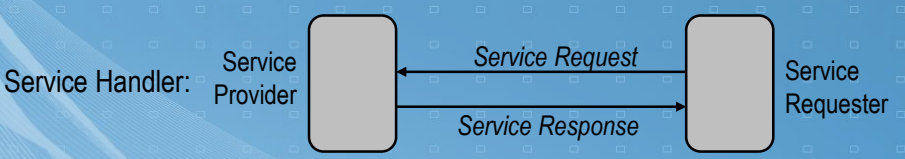
IT Service Management

- IT Service Management (ITSM) is a process-based practice intended to align the delivery of IT with the needs of the organisation, emphasising benefits to its customers
  - About 80% of all service breakdowns originate from “people and process issues”
- IT Service Management aims at providing high quality IT services meeting customers’ and users’ expectations by defining and installing management processes covering all aspects of managing the service lifecycle:
  - Planning
  - Roll-out and delivery
  - Operational support
  - Changing and improving

Service Delivery

- Service Delivery is primarily concerned with the proactive and forward-looking services that the business requires of its ICT provider in order to provide adequate support to the business users
- It is focused on the business as the *customer* of the ICT services
- Service Delivery encompasses the following processes:

1. Service Level Management
2. Capacity Management
3. IT Service Continuity Management
4. Availability Management
5. Financial Management (not covered)



Services are combined as Compositions of Services, in which a Service Handler relays requests to other Service Handlers

1 Service Level Management

- Service Level Management is the monitoring and management of the quality of service, especially ICT services specified in Service Level Agreements
  - Is one of five components in the ITIL Service Delivery area and arguably the most important set of processes within the ITIL framework
  - Service Level Management ensures that Service Level Agreements are in place with ICT Service Providers in the form of Operational Level Agreements and covering Contracts
  - Service Level Management includes the monitoring of performance against agreed benchmarks
- Service Level Management is the primary interface with the customer
  - Ensures that ICT services are delivered when & where agreed
  - Liaises with Availability Management, Capacity Management, Incident Management and Problem Management teams
  - Ensures that appropriate ICT Service Continuity Plans are in place

2 & 3 Capacity & Continuity Management

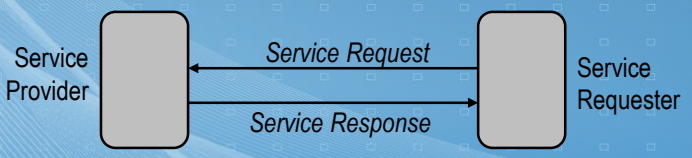
- ICT Capacity Management supports the optimum and cost effective provision of ICT services by:
  - Matching an organisation's ICT resources to its business demands
  - Integrating high-level activities of Systems and Communication Networks, Workload Management, Demand Management, Demand & Usage Modelling, Resource Capacity Planning, Resource Management, and Performance Management
- ICT Service Continuity Management seeks to:
  - Ensure the availability and rapid restoration of ICT services in the event of a disaster
  - Integrate the high level activities of ICT Risk Analysis, Contingency Plan Management, Contingency Plan Testing, and Risk Management

4 Availability Management

- Availability Management attempts to:
  - Maintain ICT service availability to support the business at a justifiable cost
  - Support activities of realising availability requirements, monitoring availability, & monitoring maintenance obligations
- Availability Management ensures that an ICT infrastructure can perform at an agreed level in the face of an anticipated demand:
  - Reliability – ability of ICT to perform at an agreed level
  - Maintainability – ability of ICT to be maintained at a operational state to deliver an agreed level of service
  - Serviceability – ability of a service provider to maintain the availability of component or function at an agreed level of service under a third party contract
  - Resilience – freedom from operational failure and maintenance of service reliability (Business Continuity Plans)
  - Security – ensuring confidentiality, integrity and availability of data, software & Intellectual Property (IP)

Service categories

- Service Requester:
  - *A service requester* or *service consumer* is the service handler that issues a service request as a message that is directed to a service provider
- Service Provider:
  - *A service provider or responder* is the service handler that responds to the service request/requester in the form of a service request
    - Usually the initiator
    - Often a Web Service
- Intermediary services:
  - Those service handlers which re-transmit messages along the routes between service requestors and service providers



Services

Services are those economic activities in which the primary output is neither a product or a construction

Services	Goods
An activity or process	A physical object or constructed artefact
Intangible	Normally tangible
Simultaneous production and consumption, consumers collaborate in the production and consumption	Separation of production and consumption, allows products to be warehoused and replaced
Heterogeneous by their nature	Normally occur in homogeneous batches
Have only an instantaneous life cycle	Can have an indefinite life cycle
Cannot be replicated	Normally capable of indefinite replication
Totally perishable – cease to exist immediately after delivery	Varying from non-perishable to perishable, depending on the nature of the object

Ref. Quinn, JB & Cagnon, CE (1986) 'Will services follow manufacturing into decline?', *Harvard Business Review*, Nov-Dec, 95-103.

Servitization: from Goods to Services

- Customers want more service and are no longer satisfied with goods alone
  - They expect a guarantee it works, ease of use & repair, 24-hour support
- Producers/suppliers capture market shares by layering services on top of products, to meet or exceed consumers' expectations
  - Consumers do not realistically distinguish between products and associated services
- Differentiation – goods are easy to replicate, a culture of good service is much harder to replicate
  - Consumers react more immediately and personally to good service than to good products alone
  - Service differentiation and support has more impact in the long term than product differentiation

Components of Service Quality

- Reliability – consistency of performance and dependability
- Responsiveness – willingness or readiness of providers to provide services
- Competence – the possession of skills and knowledge required to perform the service to accepted standards
- Access – approachability and ease of contact
- Courtesy – politeness, respect, consideration and friendliness
- Communication – listening and keeping customers informed in their language terms
- Credibility – trustworthiness, believability and honesty
- Security – freedom from danger, risk or doubt
- Understanding/acquiring knowledge of the consumers
- Tangible evidence of service – physical facilities, appearance of staff, tools and equipment, physical artifacts (business cards, documentation, tokens, associates, deliverables)

Ref: Buttle, F (1996) 'Servqual: review, critique, research agenda', *European Journal of Marketing*, Vol 30, 1, 8-32.

THE UNIVERSITY OF MELBOURNE

Service Blueprinting

- Services provide a convenient abstraction, that is understandable by both clients and also system engineers
- Requirements can be expressed as service blueprints – a collection of services, with associated service requests and service responses
- These services can be implemented as Service Units, which accept Service Request Messages, and reply with Service Response Messages
- A Service Blueprint provides a convenient interface between the human requirements of the clients, and the formal technical requirements required by system engineers
- A Service Blueprint can be implemented on top of a system, some of which will be implemented using a object-oriented design
- Service Orientation is a high level of abstraction that interfaces well with both a systems view, and an object-oriented design view

Service Blueprinting

- A service blueprint is an operational planning tool that provides guidance on how a service will be provided, specifying the physical evidence, actions, and support systems/infrastructure needed to deliver the service across its different channels
- Service blueprinting is a useful form of capturing requirements and clarifying both the client's and engineer's concepts of a service
- Service processes and interactions are visualised as a flowchart that includes interaction with the customer in the front stage, separated from the back stage
- Consists of:
  - Inputs
  - Processes
  - Outputs

Service Blueprinting Layers

Physical Evidence

- Tangible, physical evidence that a customer brings to or takes from the service

Customer Actions

- All of the steps that a customer can undertake as part of the service delivery process.
- Above the Line of Service Interaction

Onstage / Visible Contact Employee Actions

- The visible, face-to-face or over-the-counter actions between the server and the customer
- Above the Line of Service Visibility

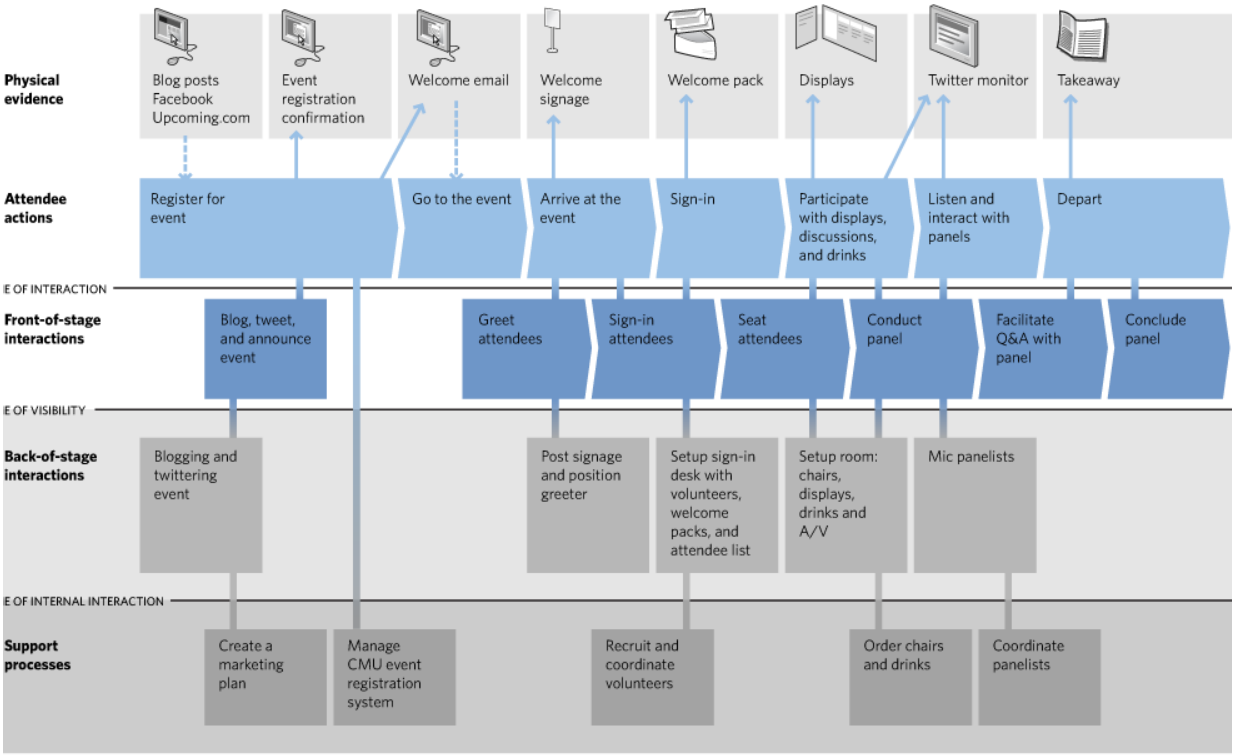
Backstage / Invisible Contact Employee Actions

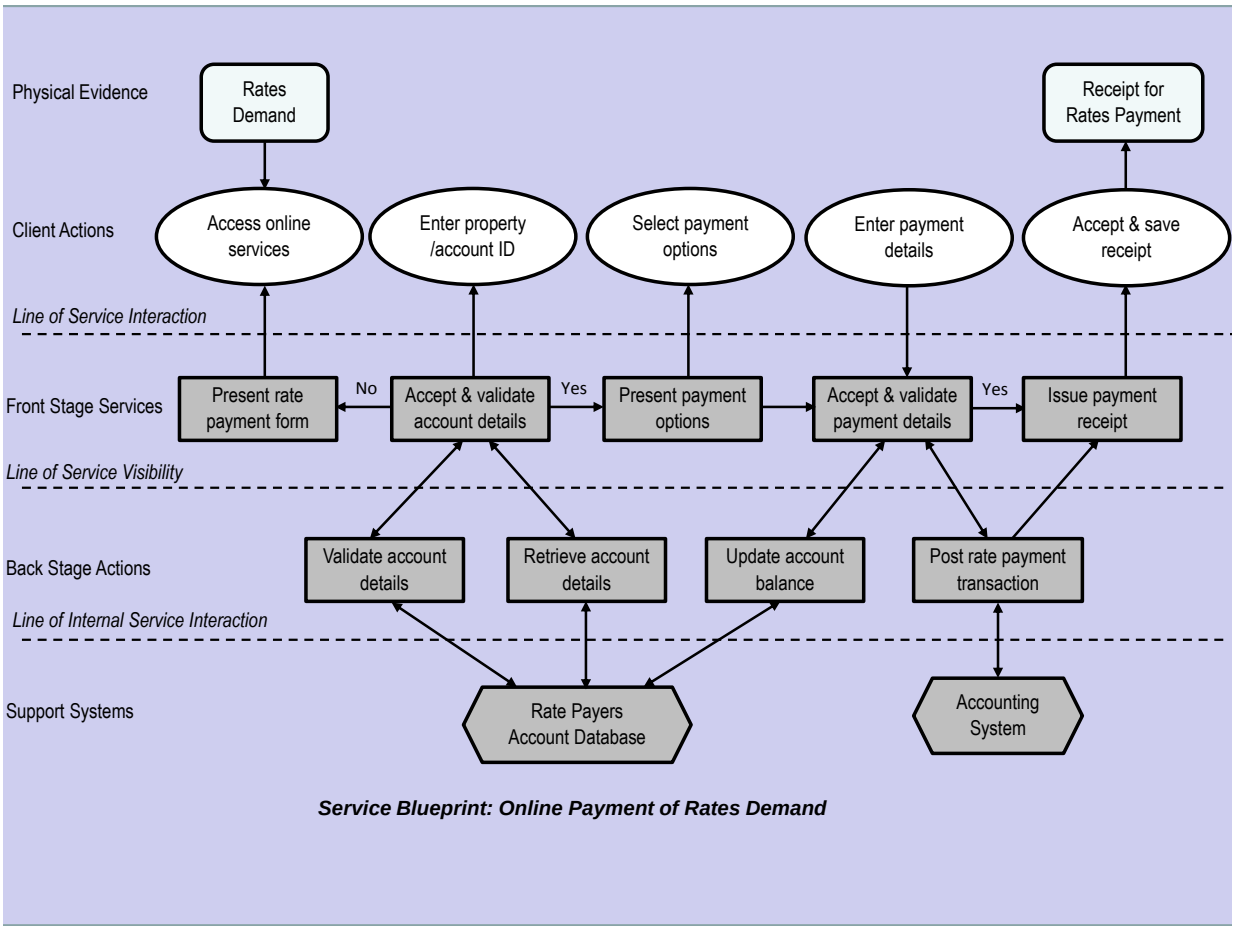
- The actions that need to be performed in order to support the service delivery
- Below the Line of Service Visibility

Support Processes / Systems

- the internal processes and systems that are required to support the backstage actions
- Below the Line of Internal Service Interaction

Service Blueprint for the organisation of a conference





Service Blueprint: Online Payment of Rates Demand

Service Blueprinting - Methodology

1. Clarify the service process(es) to be blueprinted
2. Identify the customers who will engage will these service processes
3. View the service from the customers' perspectives
4. Clarify the physical evidence of service for the customer at each step
5. Map the actions of the server, both visible (onstage) and invisible (backstage)
6. Connect the server actions to the required support processes / systems

This needs to be performed iteratively, until the service map is complete

Refs. Shostack, GL (1984) "Designing Services that Deliver", Harvard Business Review, vol. 62, no. 1 January - February 1984, pp. 133–139.

Bitner, MJ, Ostrom, AL & Morgan , FN (2008) "Service Blueprinting: A Practical Technique for Service Innovation", California Management Review.

Building a Blueprint

1. Identify activities, sequence of activities and linkages between activities
   - Activities include:
     1. Customer actions
     2. Front stage contact personnel actions
     3. Back stage contact personnel actions
     4. Support activities
2. Identify line of visibility and add to blueprint
3. Identify standards and tolerances, scripts, operating procedures, supporting services and inventory for each step and add to blueprint

Building a Blueprint cont

4. If required, draw additional lines such as line of physical interaction and line of ICT interaction (recommended for complex services)
5. Specify timeframes
   - Show average timing or minimum tolerable customer expectations for each step and indicate responsible personnel
6. Identify and note fail points
7. Manipulate divergence and complexity

Using a Service Blueprint

- Provides an understanding of service creation
- Services are becoming more challenging and complex
  - Service blueprints helps us to understand all the moving parts of a service by visualising their interconnections, dependencies, and breakdowns
  - Service blueprinting is also a good process for quickly prototyping the components of a service
  - Can be used to improve a service by understanding the original service in detail – & can identify & eliminate or minimise problems
  - Can be used to design a new service and for the creation of service prototypes and testing before a service is launched to customers
  - Can be used to clarifying how services work and can reveal gaps
  - Can be used to understanding all the actors in a service

THE UNIVERSITY OF MELBOURNE

Blueprinting & Service Oriented Architectures (SOA)

- Blueprinting an SOA provides a view that can proactively identify potential problems and adapt quickly to changes instead of churning along as a series of reactive unconnected parts, each operating independently
- Services are event-producing building blocks that processes the events as they occur, publishes the processes results via an event channel, which then triggers other services
- SOA is covered in more detail in Session 06

THE UNIVERSITY OF MELBOURNE

Summary

- Traditionally ICT infrastructures have been designed using a systems architecture paradigm
- Software engineering has tended toward object-oriented design as a dominant design paradigm
- Service Management is a key component of ICT Infrastructure management (ITIL)
- A general trend in ICT development has been away from technology and applications to processes and service

THE UNIVERSITY OF MELBOURNE

References

- Bitner, MJ, Ostrom, AL & Morgan, FN (2007) "Service Blueprinting: A Practical Technique for Service Innovation", Working Paper, Center for Services Leadership, Arizona State University, 24 pp.
- Nguyen, DK, Lelli, F, Pappzouglou, MP & van der Heuvel, W-J (2012) "Blueprinting Approach in Support of Cloud Computing", Future Internet 2012 4(1): 322-346, 25 pp.

Videos

- Service Blueprint (5.20 min) https://www.youtube.com/watch?v=gE_SNjP7Pik