Chapter 3 - Technologies underpinning AIS

Gerhard Kristandl

Introduction

It is now widely accepted that Information Technology (IT) has been, and still is, a major driver for accounting to become a knowledge service profession (Granlund, 2011). AIS have grown into complex decision-support systems whilst increasing the speed and accuracy of more traditional accounting tasks (Mauldin and Ruchala, 1999). IT impacts the quality of the AIS (measured in terms of scope, timeliness, aggregation, reliability, flexibility and usefulness), which in turn impacts the quality of accounting information (Wisna, 2013). To become the enabling and empowering tool AIS are considered to be, they require technological underpinnings that facilitates their smooth operation (Choe, 1998; Gelinas, Dull and Wheeler, 2015; Ghasemi et al., 2011; Wisna, 2013). Inadequate technology underpinning AIS can burden the company with extra maintenance and data recovery costs, and issues with data reliability, security and privacy. Thus, inadequate technology can potentially corrupt the very outcomes of an AIS, namely reports and decision-relevant information (Ghasemi et al., 2011), leading to incorrect, unreliable decisions.

In the remainder of this chapter, the general relationship between AIS and technology is detailed. Further, networks are revealed as a necessary technological feature of AIS, before the components that are aligned to build and run an AIS are discussed, namely hardware and software. As such, this chapter adopts a drill-down approach to illustrate the technologies that underpin modern AIS. It needs to be stated at this point that networks, hardware and software should not be seen as separate classifications – they have to work together to provide a reliable basis for any AIS to perform and provide reports and decision-relevant information, as the technology defines the scope and limitations of what an AIS can and cannot do (Curtis and Cobham, 2005). The users of an AIS (such as managers and accountants) need to be certain that the system works as intended, as information from AIS shape its users' perception of reality (O'Donnell and David, 2000).

Accounting information systems and technology

Accounting has experienced many improvements due to the computerisation of accounting processes (Ghasemi et al., 2011). Traditional paper-based ledgers and book-keeping processes have been automatised and mirrored in AIS, which eventually morphed into full decision-making systems (O'Donnell and David, 2000). An AIS in this context is a cohesive organisational structure (Boczko, 2007); a set of processes, functions, interrelated activities, documents and technologies (Hurt, 2016) that captures, processes and reports data, and as such provides information for decision-making and control purposes to internal (Quinn and Kristandl, 2014) and external parties (Hurt, 2016). Historically speaking, an AIS was a

specialised subsystem of Management Information Systems (MIS), and thus integrated with other information systems in firms. With the rise of material resource planning (MRP) and subsequently enterprise resource planning (ERP) systems, AIS have become even more integrated with other information systems (Gelinas, Dull and Wheeler, 2015). This has an important implication for the view taken in this chapter – the technologies that enable AIS to run are the same for other types of information systems. They share the same networks, hardware, software and other components that make up the technological basis for them to be efficiently and effectively operated. This view is in line with Gelinas, Dull and Wheeler (2015, p.14) who explain that the distinctions between separate information systems have become somewhat blurred, and a clear differentiation between IS and AIS has been given up today.

As discussed in more detail in other chapters of this companion, an AIS fulfils not only a decision-oriented, but also a controlling function. The cohesiveness of the AIS structure is achieved through prudent, integrated system design and the interactions of the human actors along a network of computerised resources that capture, process and deliver the required information (Boczko, 2007; Gelinas, Dull and Wheeler, 2015; Ghasemi et al., 2011). These resources can be determined as a collection of computer hardware and software, connected to one another within a network (Ghasemi et al., 2011; Quinn and Kristandl, 2014), and need to be implemented and maintained to support business processes (Gelinas, Dull and Wheeler, 2015). Although non-computerised AIS exist (Quinn and Kristandl, 2014), modern businesses that employ such a system can rarely do so without the use of computers. In very simple terms, to capture – ideally – *all* relevant, transaction-based information for accounting purposes in an organisation, the resources need to be linked to one another to input information, send the information to the right addressee (another computer or person) for processing, and finally onwards to the party that requires the processed data for decision-making, reporting, or control (including audit).

Networks

As of the time of writing, almost every business is connected to and uses networks, particularly the Internet. As of 2014, 97% of enterprises in the EU had access to the Internet, 92% used a fixed broadband connection, and 66% furnished their employees with mobile internet devices for business purposes (Eurostat, 2015). Using mobile technologies such as mobile payment systems and capturing document data (e.g. via a scan of invoices) have a major impact on AIS and the opportunities to collect and report data in real time (Brandas, Megan and Didraga, 2015; Trigo, Belfo and Estebanez, 2014). This illustrates the importance and – to some extent – implicitness of networks in today's corporate environment.

The technological view on networks is that of a so-called *hard network* (as opposed to social soft-type, or the logical, more abstract semi-soft type; see Boczko, 2007). A hard network is the physical representation of a group of devices (e.g. computers and servers) connected to one another via a network interface card (NIC) and wired/wireless links, managed by software allowing data exchange (Hall, 2016).

Wired connections (e.g. copper wire, twisted pair, coaxial, fibre optic) connect the various computers in a permanent manner, typically via point-to-point links (Tanenbaum and Wetherall, 2011). Wired connections, once set up, are difficult to reconfigure. Hardware links provide the physical means and infrastructure to enable networking; this includes computer-to-computer, computer-to-server, server-to-server, or computer-to-periphery (e.g. shared network printers; Quinn and Kristandl, 2014) connections. *Wireless* networks connect via broadcast links, such as high-frequency radio signals, infrared, electromagnetic signals, or laser for short-distance networks; or mobile telephony, microwaves, or satellite for long-distance networks (Boczko, 2007; Richardson, Chang and Smith, 2014; Tanenbaum and Wetherall, 2011). Wireless networks provide the advantages of mobility, rapid deployment, flexibility and scalability, low-cost setup and easy maintenance (Boczko, 2007; Richardson, Chang and Smith, 2014). However, they can be limited by the distance to the access point, as well as the number of wireless devices using the existing bandwidth at the same time. Wireless network access typically requires access to a wired intranet or internet, linked via a wireless network card (WNIC).

All types of computer networks can be described using the following three attributes (Boczko, 2007):

- Architecture
- Topology
- Protocols

Architecture

The architecture describes the technological layout and configuration of a network, and includes the definition of intra-network and inter-network relationships, the physical configuration, the functional organisation, the operational procedures employed, as well as the data used (Boczko, 2007; Magal and Word, 2012). It also determines the geographical distribution of the network resources and the scale of the network (Tanenbaum and Wetherall, 2011).

Network architecture can either be described in systems or hardware terms. A typical example of the former is a *client-server* model (C-S). The purpose of this type of *systems architecture* is to interconnect and distribute software and hardware efficiently and effectively across a network (Boczko, 2007). Here, a *client* is a computer or workstation that uses services (programs, applications, data processing), whereas a *server* is a computer that manages and allocates these services (Curtis and Cobham, 2005; Hall, 2016). Clients require servers to access network resources to process data in an AIS. The C-S model is an example of a multi-tier architecture (see Figure 3.1), where presentation, data/application processing and data storage/management are separated into different layers (see Chapter 4 - Systems Planning, Design and Implementation). Information systems such as SAP ERP are based on a C-S architecture, with only the graphical user interface (GUI) running at the user end. The C-S model is widely used in online commerce, and forms the underlying idea behind *cloud computing* with the main difference that the data is stored on a server that is owned by a cloud

provider instead of the company, accessed via the Internet (Lin and Chen, 2012; Zissis and Lekkas, 2012).

INSERT FIGURE 3.1 NEAR HERE

Figure 3.1: Client-Server Architecture

An evolution of the standard C-S architecture is *Service-Oriented Architecture* (SOA), based on the concept of designing and developing inter-operable functions and applications (services) that are reusable (Hall, 2016; Magal and Word, 2012). SOA enables a company to create composite applications such as AIS functions without needing to change the underlying application. This *modular* approach provides a cost-efficient way to lower cost and complexity during integration programmes, whilst simultaneously offering access to more complex software. Many providers of AIS, such as SAP, have service-enabled their applications to permit businesses to create composite information systems. Where C-S and SOA are examples for systems architecture, *hardware architecture* supports the distribution of software, data, and processes. Typical examples are LANs, WANs and VPNs, which are now discussed.

A *LAN* (Local Area Network; see Figure 3.2) is a network within geographically close confines, often within the same room or building, privately owned by a single organisation (Quinn and Kristandl, 2014; Tanenbaum and Wetherall, 2011). Within this type of network, computers (*nodes*), servers and peripheral devices, such as printers, are connected either wired or wirelessly (WLAN) (Richardson, Chang and Smith, 2014). Hubs and switches (see *Hardware* later) interconnect the devices and send packets (formatted, small units of data; Richardson, Chang and Smith, 2014) over the network. LANs allow use of a common network operating system, centralisation of shared data and programs from a central server, as well as their downloading for local processing, communication of personal computers with outside networks (e.g. the Internet), sharing of scarce resources, email, as well as access to and use of a centralised calendar and diary (Curtis and Cobham, 2005). Sharing resources over a network is cost-efficient since there is less need for large-storage hard disks or programs for every computer within the network. Computers may act as both clients and servers in smaller LANs– such a network is then called a *peer-to-peer network* (Boczko, 2007) due to the equivalent responsibilities that each workstation fulfils. In larger LANs, workstations act as clients only, and are then linked to a server (*server network*).

INSERT FIGURE 3.2 NEAR HERE

Figure 3.2: Local Area Network

A *WAN* (Wide Area Network; see Figure 3.3) is a network over a larger geographical area (e.g. a country), connected via public (e.g. phone lines) or private (e.g. leased lines or satellite) communication facilities (Hall, 2016; Shinder, 2001). WANs provide remote access to

employees or customers, link two or more separate LANs at sites within a company, and provide the business with access to the Internet (Richardson, Chang and Smith, 2014).

INSERT FIGURE 3.3 NEAR HERE

Figure 3.3: Wide Area Network

Depending on the location of a central hub, there are two types of WAN, namely centralised and distributed. In a *centralised* WAN, all major functions (e.g. accounting, procurement, sales order processing) are carried out at the central hub. The computers in the network do not process any transactions locally, but send data processing requests remotely to the central hub (Boczko, 2007; Curtis and Cobham, 2005). All data traffic can be closely monitored, but it puts a heavy burden on the network itself. The central hub needs to queue and prioritise all concurrent requests, rendering the network much more vulnerable to a complete standstill. A *distributed* WAN, on the other hand, is decentralised in terms of data processing (Boczko, 2007; Curtis and Cobham, 2005), and thus better able to transmit and process individual transactions simultaneously. In a distributed environment, LANs are connected to one another, and/or to larger WANs. Depending on the type of LANs linked, this is achieved via *bridges* (linking same-type LANs) or *gateways* (linking different-type LANs; Hall, 2016). The largest distributed WAN to date is the Internet, and the World Wide Web is - simply put - a WAN that uses a client/(web) server architecture to transmit data and process tasks (Shinder, 2001).

Variations of WANs are *MANs* (Metropolitan Area Networks) and *CANs* (Campus Area Networks) that can be quite large, but are confined within a city or campus (Shinder, 2001; Tanenbaum and Wetherall, 2011). A *VPN* (Virtual Private Network) is created using a secure *tunnel* between a corporate WAN and (home) offices via virtual links over the Internet rather than leased lines (Richardson, Chang and Smith, 2014; Tanenbaum and Wetherall, 2011). VPNs came to prominence due to the larger bandwidth availability, enabling remote access to corporate WANs from outside the business premises, such as salespersons, home offices and business partners that require access. Companies that use cloud technology for their networks are particularly in need of secure access points, which a VPN provides. This type of network provides a cheap way of connection, but due to their use of the internet, suffer a lower Quality-of-Service (QoS) than corporate WANs (Richardson, Chang and Smith, 2014).

Topology

The term *topology* denotes the shape of a network, determining how network devices are connected to one another, and how they communicate (Boczko, 2007). Figure 3.4 shows the most common topologies utilised in networks.

INSERT FIGURE 3.4 NEAR HERE

Figure 3.4: Network topologies

A *bus topology* is a linear topology where clients share a central line connection (Boczko, 2007; Hall, 2016), either linear or in a daisy chain (see Figure 3.4 a and b). When data is sent along the network, it contains a unique network address for the desired destination, and will thus be delivered to the correct network resource. Although a bus topology is easy to set up and extend, the connected devices are competing for connection resources, as only one line is available to them (Hall, 2016). In cases where two or more clients want to use the network at the same time, this might lead to queuing and slow operation of the network – a situation that is exacerbated by every additional node added to the bus. Thus, this topology is limited in size, as it may become difficult to operate and manage (Boczko, 2007).

In a *ring topology,* each node is connected to two other nodes, and represents a peer-to-peer arrangement (see Figure 3.4 c). As opposed to a bus topology in a daisy-chain configuration, a ring topology creates a closed loop of nodes, meaning that if a signal is sent along the network, and no destination node accepts it, it returns to the sending node (Boczko, 2007; Hall, 2016). Each node has equal status, but only one node can communicate at a time. Unlike in a bus topology, the nodes along a ring topology move the signal along rather than ignore it. This improves network speed; the scalability of the network is also superior to a bus topology, since additional nodes do not significantly impact network speeds. It requires, however, more connections than a bus network, is costlier to implement, and if one single node fails, it will impact the entire network. Both ring and bus topologies have become side-lined in favour of the more stable star typology (see below; Quinn and Kristandl, 2014). A *mesh topology* (see Figure 3.4 d) is a variation of the bus topology, where every node is connected with every other node (Boczko, 2007). Although providing a more stable and reliable network than the ring topology, especially for small networks, its complexity increases considerably when the network grows. This in turn can render network management and reconfiguration difficult and costly (Boczko, 2007). Mesh topologies are often used in WANs to connect various LANs to one another.

In a *star topology* (see Figure 3.4 e), all devices are linked to a central computer which acts as a transmission device (Boczko, 2007; Hall, 2016). In this case, signals are transmitted via the central host computer rather than along the entire network. It is relatively easy to implement, extend and monitor, and if a device fails, it typically does not have an impact on the entire network (Quinn and Kristandl, 2014), unless the central computer fails. Other disadvantages lie in higher costs (maintenance, security) and higher risk of infection (as all data runs via the central hub; Boczko, 2007). The star topology is often used in both centralised and distributed WANs where the central host computer is a mainframe (Hall, 2016).

The topologies above can be combined based on business needs. Examples of such *hybrid topologies* are star-bus (or tree) or star-ring (or token ring) topologies (Boczko, 2007) that aim to combine the advantages and eliminate the drawbacks of their individual contributors. Figure 3.4 f shows an example of a star-bus topology that is easier to extend and more resilient than a pure bus topology.

Protocols

Without instructions to manage the communication and flow of data between the devices, the network that an AIS is running on would merely be a physical arrangement of computers and cables. A network requires *protocols* - formalised and uniform set of rules and standards that govern the syntax, semantics and synchronisation of communications between nodes - to enable network devices to communicate (Hall, 2016; Quinn and Kristandl, 2014). AIS offerings need to comply with these standards - they define the formals rules of conduct and etiquette to avoid misinterpretation and discord. Hall (2016) states that

> "Establishing a standard of conduct through protocols, which all members of the community understand and practice, minimizes the risk of miscommunications between nations of different cultures." (p. 505)

Standardised reference models for network protocols help ensure that interconnection is achieved in a seamless manner. The *Open Systems Interconnection* (OSI) standard is a model of seven-layered protocols (Tanenbaum and Wetherall, 2011) that define standards that govern protocol development and intra-layer and inter-layer protocol communication. This model (see Table 3.1) also provides a clear distinction between data manipulation (layers 1-4) and data communication (layers 5-7).

INSERT TABLE 3.1 NEAR HERE

Table 3.1: The OSI and TCP/IP reference models (Adapted from Tanenbaum and Wetherall, 2011, p.46)

The OSI model is not used as often today, and has been criticised for being flawed, too complex and inefficient. Instead, the four-layered *TCP/IP* (Transfer Control Protocol/Internet Protocol; see Table 3.1) reference model is considered more practical and pragmatic (Tanenbaum and Wetherall, 2011), avoiding many issues that the OSI model brought with it[1]. Many of the better-known network protocols today, come from the TCP/IP model, such as:

- Ethernet (link layer);
- Internet Protocol (network layer);
- TCP (transport layer);
- Hypertext Transfer Protocol (HTTP, application layer); or
- FTP (File Transfer Protocol, application layer).

Tanenbaum and Wetherall (2011) emphasise that in spite of its criticism, the OSI model is still very valid for today, although many of its protocols are not in use anymore. For the TCP/IP model, the reverse is true: whilst the model is rarely used, the protocols are widely employed.

[1] The details of OSI v TCP/IP is beyond the scope of this chapter. For a detailed discussion, please see Tanenbaum and Wetherall (2011)

Hardware

Hardware in IT comprises all physical computing machinery and equipment used to capture, process and store data (Curtis and Cobham, 2005). This includes computers and their components (e.g. keyboards, disk drives, etc.) and servers, but also cloud-enabled devices such as tablets and smartphones (Quinn and Kristandl, 2014). A *computer* can be defined as a workstation that provides a network-human interface; an access point to the AIS for both input and output of the required accounting data. The role of a *server* (see also *Architecture* earlier) in a network is to process and manage the flow of information between the nodes, and allocate processing resources to the task at hand.

From a common systems model point of view, a computer comprises (Curtis and Cobham, 2005):

- Input devices that accept, convert and transmit data;
- A central processing unit (CPU) that executes program instructions, controls and coordinates data movement, and carries out arithmetic and logical operations whilst storing programs and data;
- A secondary (backing) storage that maintains a permanent record of data and programs beyond execution and for security;
- Output devices that receive information from the CPU and convert it into the required format.

This common systems model as illustrated by Curtis and Cobham (2005) can be detailed further. Table 3.2 lists examples of hardware that are typically present in an individual computer. However, from an organisational perspective where a higher degree of computing and communications power is required, it can be separated into individual devices which are connected via network links (Quinn and Kristandl, 2014). The connections between nodes and servers require communication devices (see below) that creates and manages these links, e.g. network cards, repeaters and hubs (Boczko, 2007; Richardson, Chang and Smith, 2014). As discussed earlier, these connections can either be wired or wireless.

INSERT TABLE 3.2 NEAR HERE

Table 3.2: Examples of computer hardware (Adapted from Quinn and Kristandl, 2014, p.15)

Input devices accept data, convert them into a machine-readable form, and transmit them within a computer system (Curtis and Cobham, 2005). Keyboards are a typical input device, where information is entered into the system and converted into binary code whilst being shown on a screen. Other input devices (see Table 3.2) also capture the initial data entered into the system. Scanners, for instance, are a widely used device, using technology like optical character recognition (OCR) or magnetic ink character recognition (MICR) to identify relevant data from a source document. OCR is often used by utility companies, credit card companies, or councils. Documents that are scanned using OCR typically come with specific instructions on how to fill in the data, such as writing in capital letters, black ink and within a confined box; this is to enable the OCR to correctly identify the characters written (Curtis and Cobham, 2005). A common use of MICR is in processing cheques in banks, where the cheque number, account and sort codes are written in magnetic ink on the cheque. Barcode readers are another widespread type of input device, particularly in logistical processes to record the movement of goods. A good example of an industry that relies on data input via barcode scanners is food retailing (e.g. supermarkets). Voice recognition via microphones are also widely used for data entry, for instance in call centres or customer service to screen and route calls. Lastly, pointing devices such as a mouse is a commonplace feature in computers nowadays. These various types of input devices have advantages and disadvantages related to accuracy and cost. Keyboards, for instance, are by and large inexpensive input devices, but are subject to error, since data is typically entered by a person (Curtis and Cobham, 2005). At the same time, data entry can be quite slow when keyboards are being used – this is different with scanners or barcode readers where data entry is quick and less prone to error, but this comes with the disadvantage that they are costlier when acquired and operated.

Processors enable a computing device to decode and execute program instructions, control and coordinate data movements, and perform arithmetic and logical operations (Curtis and Cobham, 2005; Quinn and Kristandl, 2014). Examples of processor manufacturers are Intel (Pentium), AMD, or Apple (A5 chip). Processors comprise the arithmetic and logic unit for calculations and data comparisons, and the control unit for data movements. Together with the main memory unit (random-access memory, RAM), used for storage of currently used data/programs and the operating system, processors build the CPU (Curtis and Cobham, 2005). The history of processors has shown an exponential increase in processing power, which in turn allows for quicker program execution and larger RAM for program-multitasking in computers today.

Storage devices serve to maintain the input and processed data as well as programs on a permanent basis (Curtis and Cobham, 2005), for immediate or later use. Storage devices can

also provide backup for data in case of security and integrity issues. As opposed to the main memory, where the CPU only stores data whilst the computer is switched on, the storage devices hold the data even if powered off. Different types of storage devices differ in speed of data retrieval, capacity, cost and robustness. Hard disks are a typical storage device in most computers systems. Types of hard disks are magnetic drives (hard drive disk, or HDD, where a laser records the data on and reads it off the disk), optical disks (Blu-ray, DVD/CD), flash drives (USB sticks, external drives), or more recently, cloud storage where the stored data is accessed via the Internet. The latter type in particular experienced a rapid increase in usage, as it allows not only large organisations, but also small and medium-sized companies to acquire and operate hitherto unaffordable AIS technology (Brandas, Megan and Didraga, 2015). Older types of storage devices such as magnetic tapes or floppy disks still exist in some organisations (such as the US Nuclear Weapons Force; BBC, 2016), but nowadays do not feature in modern AIS technology.

Output devices are used to display information in the required format. Typical examples are computer monitors, tablet and smartphone screens, printers, and speakers (Curtis and Cobham, 2005; Quinn and Kristandl, 2014). Screens in general appear to be the most common type. They are either connected to a desktop computer, embodied in laptops, tablets, smartphones, machinery, vehicles – the internet of things has enabled internet connectivity to the most unusual devices, and thus creates more opportunity for collecting data (Mazhelis, Luoma and Warma, 2012). Printers are another output device that issue information by means of laser, ink, dot-matrix, or thermal printing technology (Curtis and Cobham, 2005). Larger type of printers are plotters, chain and drum printers, but these are mostly attached to larger mainframe computers, and do not provide the quality and flexibility suitable for AIS information output. Of course, hard disks can also serve as output devices if the information stored is later used as an input in another computer system. In this case, the information is not issued to an end user.

Communication/network devices are hardware that allow network resources to interconnect with one another. As discussed earlier, the actual connection between a node and network is either done wired (cabling) or wireless (NIC). However, the cabling or connection alone is not enough – data that is transmitted along the network needs to find the right address. This is done by a *switch*, a specialised computer that determines an outgoing line for incoming data, sending it to the right address (Richardson, Chang and Smith, 2014; Tanenbaum and Wetherall, 2011). *Routers* act in a similar manner to switches inasmuch as they choose the most efficient communication path through a network to the required destination (Richardson, Chang and Smith, 2014). Routers use the Internet Protocol (IP) address of both the sender and the receiver of the data, and decide which path is the most direct.

Where switches and routers determine where an incoming data packet needs to go, *hubs* merely transmit them. Packets that arrive at one port are copied to all other ports, so that all other equipment connected to the LAN receive the packet (Richardson, Chang and Smith, 2014). Networks that use hubs instead of switches are called *non-switched networks*, where communication links are shared by all devices (Curtis and Cobham, 2005). Typically, the

performance on switched networks is higher than non-switched, as there are no data collisions, data can be transmitted simultaneously, and the capacity is used more efficiently (Tanenbaum and Wetherall, 2011). Not just from a performance, but also a data security point of view, switched networks are preferable, as data traffic is only sent to the address where it is required (Tanenbaum and Wetherall, 2011). They are also more efficient to monitor, as corporate *firewalls* are a security system comprising hardware like switches, routers, servers and software, to allow or deny a data packet that enters or exits a company LAN to continue on their transmission path (Richardson, Chang and Smith, 2014).

Software

Although hardware and networks are essential in enabling a smooth-running and purpose-driven AIS, without software it would not work. *Software* is the general term used to describe the instructions that control the operations of hardware (Curtis and Cobham, 2005; Quinn and Kristandl, 2014).[2] Software can either be categorised as operating systems (OS), database systems, or applications software, and requires the use of programming languages to design and create them.

Operating systems software

This type of software comprises programs that enable an efficient and smooth operation of the computer system (Curtis and Cobham, 2005), and is considered the "most important piece of software" (Richardson, Chang and Smith, 2014, p. 242). It is the basis for the hardware to function, controls the flow of multiprogramming, schedules tasks and provides a way for applications software (see below) to work with the hardware. It further allocates computer resources to users and applications, and manages the human-computer interface and access points to the network. An OS provides the following four functions (Curtis and Cobham, 2005):

- Handling of data interchange between input/output devices and the CPU;
- Loading of data and programs into and out of the main memory;
- Allocating main memory to data and programs as needed (managing processes and memory, so that all programs receive a share of the available resources);
- Handling job scheduling, multiprogramming and multiprocessing.

Examples of OS available on the market are Microsoft Windows, Apple OS, Linux, Unix, Chrome, and Android and iOS for mobile devices. Note that not all of these incur acquisition costs – Linux distributions like Ubuntu and openSUSE are free, whereas Windows incurs a cost based on different licence models. All of these OS provide a Graphical User Interface (GUI) as opposed to text-based interfaces that require the entry of command lines to work with the system – an example is MS-DOS which has been superseded by the more user-friendly Windows systems.

[2] The term *firmware* also exists, indicating an inseparable combination of hardware and software, it being a set of instructions that is permanently encoded on a microchip.

Due to its crucial role in the smooth running of an AIS, the OS needs to be secured against internal and external threats to its integrity. This includes intended or unintended security threats by users, computers, applications, the OS itself, as well as hacking or data leaks of sensitive information to the outside (Richardson, Chang and Smith, 2014). As such, the OS requires clear IT governance policies that control who can access the system, system and network resources, and actions that are allowed by users. These security features are even more relevant in a cloud-computing environment, where several "virtual" (rather than actual) OS share the same hardware (such as the same server), and could potentially become permeable, allowing access to resources between two instances of an OS running on the same platform.

Database systems software

In an integrated AIS, corporate accounting data is typically stored on a central database to ensure that all relevant applications access the same kind of information when processing data. As such, databases are another crucial component in an AIS, and require a database system that is able to record, manage and store a massive amount of day-to-day accounting data. A *database system* comprises two main software components, namely a data warehouse (a centralised collection of companywide data for a long period of time), and operational databases that draw data from the data warehouse (Richardson, Chang and Smith, 2014). Operational databases contain the data for the current fiscal year, updated whenever a transaction is processed. Periodically, data is uploaded from the operational databases to the data warehouse to provide decision-useful information to identify trends and patterns – the process of analysing them as such is called data mining, using Online Analytical Processing (OLAP).

Applications software

Applications software are programs that fulfil specific user functions (Quinn and Kristandl, 2014). In an accounting context, this may mean functions like sales ledger processing, budgeting, forecasting, or reporting (Curtis and Cobham, 2005). During their execution, programs and the data required are stored in the RAM. Most AIS are offered as *applications packages* that include functional modules such as sales ledgers, accounts receivable, accounts payable, payroll, credit and payment systems. These desktop accounting packages are offered and distributed by ERP software providers such as SAP or Oracle, or mid-level accounting package providers such as Sage. The same accounting packages are also provided via a subscription-based cloud offering by the same companies.

If applications packages are unable to fulfil a specific business need, a company could commission development (or develop it using their in-house resources). However, such specific software developments require lengthy analysis, design and testing phases (Curtis and Cobham, 2005). *Specially commissioned software* can become very costly as opposed to applications packages; if specially commissioned, the cost of software development, testing and subsequent updates needs to be absorbed in full by the commissioning business (Curtis and Cobham, 2005). Further considerations stem from the frequent requirement to run the

commissioned software on different types of hardware (portability), and the existence of professional documentation that enables adequate IT support. On the other hand, specially commissioned software provides a perfect fit to the corporate requirements; this includes the elimination of redundant functions that may not be needed (but paid for), and the compatibility with existing specially commissioned software (Curtis and Cobham, 2005). Whether a company is able to commission specially designed AIS software is more often than not a question of affordability, which by and large rules out the ability of smaller businesses to commission them. Interestingly, it appears that most companies prefer packaged software to self-developed or commissioned ones (Granlund, 2011).

To avoid compatibility issues between programs, businesses often acquire entire software suites, where several programs are integrated and sold together. Suites provide inter-program compatibility in data exchange and user interface – good examples are Microsoft Office (e.g. Word, Excel, Access, Outlook) or the SAP Business Suite (containing ERP, Customer Relationship Management, Supplier Relationship Management, Supply Chain Management, Product Lifecycle Management).

Programming languages

Software is written as a set of instructions to control computer operations. These instructions are written in a formal language to communicate them to the computer – a *programming language.* Table 3.3 details in brief three of the main categories of programming languages, as well as their characteristics (Curtis and Cobham, 2005):

INSERT TABLE 3.3 NEAR HERE

Table 3.3: Programming language categories

Machine-oriented programming languages have been by and large superseded by higher level ones due to their dependence on the source program and the computer architecture it is written on/for. Task-oriented languages that require compiling can be used over and over, independent of the source program, and provide portability between computer systems and architectures, as well as a certain level of security against tampering with the compiled code (Curtis and Cobham, 2005). However, task-oriented language programs operate slower due to inefficiencies in the compiling process, as well as the exclusion of the individual CPU structure. Object-oriented programming languages (OOPL) apply a logic that is different to the previous task-oriented ones, in that they do not define complex operations, but rather the objects and their (changeable) attributes that take part in the operations. OOPLs produce a more natural way of reflecting the real world, objects that are reusable (saving programming time and complexity) and a simpler syntax (Curtis and Cobham, 2005). A good example for an OOPL used in AIS is ABAP Objects, the programming language of the SAP ERP software.

Summary

This chapter detailed the technological underpinnings of AIS that are predominantly the same as for general corporate information systems. It detailed and discussed the technological perspective on networks, hardware, and software, that enable an AIS to record, process and display accounting information for reporting and decision-making. A main emphasis was placed on computer networks that not only enable AIS to record transactions in any part across a company, but also furnish the business with the computing power needed to process large amounts of data. Connected within a network are hardware that provides the physical resources, as well as software that enables the smooth operation of an information system. Common standards like protocols facilitate the inclusion of various accounting software packages to create an efficient technological environment for running an AIS.

The future of AIS will continue to be inextricably linked to their technological underpinnings. With cloud computing offering processing power hitherto unavailable to many businesses (Strauss, Kristandl and Quinn, 2015), the spread of AIS will increase with technological developments at an unprecedented growth rate. Smaller businesses will be able to employ AIS as efficiently as larger businesses have been able to for decades. At the same time, new developments like SAP HANA will enable business analytics based on the information in AIS for businesses of all shapes and sizes. Data can be gathered from more than just the typical user input, but from internet-enabled devices that were not linked to networks before – the *internet-of-things* will provide businesses with data from the most unusual of places, to be used for accounting purposes (Mazhelis, Luoma and Warma, 2012). Finally, the rise of mobile smart devices enables decision-makers to access and retrieve accounting information from their AIS on the go, at any time, in any place. As such, the technological future of AIS seems a promising and bright one.

References

BBC (2016). US nuclear force still uses floppy disks. [ONLINE] (Last updated on 26 May 2016). Available at: http://www.bbc.co.uk/news/world-us-canada-36385839 . [Accessed 16 June 2016]

Boczko, T. (2007). *Corporate Accounting Information Systems*. Harlow: FT Prentice Hall.

Brandas, C., Megan, O., and Didraga, O. (2015). Global perspectives on accounting information systems: mobile and cloud approach. *Procedia Economics and Finance*, 20, 88-93.

Curtis, G., and Cobham, D. (2005). *Business Information Systems: analysis, design and practice*. 5e. Harlow: FT Prentice Hall.

Eurostat (2015). Information society statistics – enterprises. [ONLINE] Available at: http://ec.europa.eu/eurostat/statistics-explained/index.php/Information_society_statistics_-_enterprises . [Accessed 16 June 2016].

Gelinas, U.J., Dull, R.B., and Wheeler, P.R. (2015). *Accounting Information Systems*. 10e. Stamford: Cengage Learning.

Ghasemi, M., Shafeiepour, V., Aslani, M., and Barvayeh, E. (2011). The impact of Information Technology (IT) on modern accounting systems. *Procedia – Social and Behavioral Sciences*, 28, 112-116.

Granlund, M. (2011). Extending AIS research to management accounting and control issues: A research note. *International Journal of Accounting Information Systems*, 12(1), 3-19.

Hall, J.A. (2016). *Accounting Information Systems*. 9e. Boston: Cengage Learning.

Hurt, R.L. (2016). *Accounting Information Systems – Basic Concepts and Current Issues*. 4e. New York: McGraw-Hill Education.

Lin, A., and Chen, N.-C. (2012). Cloud computing as an innovation: Perception, attitude, and adoption. *International Journal of Information Management*, 32(6), 533-540.

Magal, S.R., and Word, J. (2012). *Integrated Business Processes with ERP Systems*. Hoboken: John Wiley & Sons.

Mauldin, E.G., and Ruchala, L.V. (1999). Towards a meta-theory of accounting information systems. *Accounting, Organizations and Society*, 24(4), 317-331.

Mazhelis, O., Luoma, E., and Warma, H. (2012). Defining an Internet-of-Things Ecosystem, 1-14. In: S. Andreev, S. Balandin, and Y. Koucheryavy, eds. (2012). *Internet of Things, Smart Spaces, and Next Generation Networking*. Berlin: Springer.

Quinn, M., and Kristandl, G. (2014). *Business Information Systems for Accounting Students*. London: Pearson.

Richardson, V.J., Chang, C.J., and Smith, R. (2014). *Accounting Information Systems*. New York: McGraw-Hill Education.

Shinder, D.L. (2001). *Computer Networking Essentials*. Indianapolis: Cisco Press Core Series.

Strauss, E., Kristandl, G., and Quinn, M., (2015). The effects of cloud technology on management accounting and decision-making. *CIMA Research executive summary series*, 10(6).

Tanenbaum, A.S., and Wetherall, D.J. (2011). *Computer Networks*. 5e. Boston: Pearson Education.

Trigo, A., Belfo, F., and Estebanez, R.P. (2014). Accounting Information Systems: The Challenge of the Real-Time Reporting. *Procedia Technology*, 16, 118-127.

Wisna, N. (2013). The Effect of Information Technology on the Quality of Accounting Information system and Its impact on the Quality of Accounting Information. Research *Journal of Finance and Accounting*, 4(15), 69-75.

Zissis, D., and Lekkas, D. (2012). Addressing cloud computing security issues. *Future Generation Computer Systems*, 28(3), 583-592.