
Book Section:

McKay, A orcid.org/0000-0002-8187-4759, Davis, MC orcid.org/0000-0003-1577-7544, Hughes, HPN orcid.org/0000-0001-8454-8206 et al. (2 more authors) (2020) Designing Socio-Technical Systems: A Multi-team Case Study. In: Metcalf, GS, Kijima, K and Deguchi, H, (eds.) Handbook of Systems Sciences. Springer , Singapore . ISBN 978-981-13-0370-8

https://doi.org/10.1007/978-981-13-0370-8_13-1

Designing Socio-Technical Systems: A Multi-team Case Study

Alison McKay
a.mckay@leeds.ac.uk
Socio-Technical Centre & School of Mechanical Engineering
University of Leeds
Leeds
LS2 9JT
United Kingdom
phone: +44 113 343 2217

Matthew C. Davis
m.davis@leeds.ac.uk
Socio-Technical Centre & Leeds University Business School
University of Leeds
Leeds
LS2 9JT
United Kingdom
phone: +44 113 343 6831

Helen P. N. Hughes
h.hughes@leeds.ac.uk
Socio-Technical Centre & Leeds University Business School
University of Leeds
Leeds
LS2 9JT
United Kingdom
phone: +44 113 343 8285

Rebecca L. Pieniazek
r.pieniazek@leeds.ac.uk
Socio-Technical Centre & Leeds University Business School
University of Leeds
Leeds
LS2 9JT
United Kingdom
phone: +44 113 343 2481

Mark A. Robinson
m.robinson@lubs.leeds.ac.uk
Socio-Technical Centre & Leeds University Business School
University of Leeds
Leeds
LS2 9JT
United Kingdom
phone: +44 113 343 8284

Abstract

Technical system design processes are typically based on systems engineering vee models where designers move between functional and physical domains as they develop detailed designs of the overall system, and its subsystems and component parts. The movements between the functional and physical domains are informed by the core activities of any design process: synthesis, description, analysis and simulation, and decision making. However, delivering socio-technical systems design mindsets, such as those needed to design multi-team systems, requires a new branch of systems science that integrates human behaviour into system behaviour. Design processes built on such a science would allow system designers to compare alternative solutions in terms of their anticipated performance and consider different options with respect to functions carried out by humans and machines.

In this chapter we use a systems design process vee model and apply it to a case study that involves the design of a multi-team customer service system. Both the application of the vee model (i.e., the proposed design process) and the results of its application (i.e., the multi-team customer service system) can be regarded as socio-technical systems and are used to illustrate and elaborate on Clegg's (2000) socio-technical principles for system design. On this basis, we provide a practical framework for designing socio-technical systems and identify requirements for developing future methods and tools to support this process.

Keywords: socio-technical principles, function allocation, simulation, resilience, system evaluation

1 Introduction

The key premise of socio-technical systems design thinking is that people are an integral part of the system alongside technology and should be regarded as such through the entire design process. Many design methods include people, but as users of products and systems rather than as integral parts of the systems themselves. For example, human computer interaction specialists create software interfaces that are easier for people to use, user-centred design results in physical interfaces that are easier for people to use, and co-design includes people, users, and other stakeholders earlier in the design process so that their needs and views can be better understood. However, in each case, critical design decisions have been made before these methods are implemented. For example, who decided which functions would be carried out by technology and which by people? Similarly, who decided who the users and others involved in the design process would be? Furthermore, in both cases, what was the basis of these decisions and, if necessary, could they be audited and unravelled? In this chapter we advocate the development of socio-technical system design mindsets that support these kinds of decisions and allow system designers to consider, in a systematic manner, a range of alternative socio-technical system scenarios.

Systems engineering approaches are widely and successfully used in the development of large complex technical systems such as aeroplanes and automobiles. Principles of systems engineering (RAEng, 2007) emphasise the importance of understanding the purpose of the system, holistic and creative thinking, and the use of systematic processes to create high value and lasting solutions for users, on time and to budget. In this chapter, we explore the applicability of systems engineering principles to the design of socio-technical systems, to improve performance in technical product development systems among other outcomes. For example, to improve UK productivity, Clegg (2016) argues for the application of socio-technical principles to the design of industrial systems.

Socio-technical systems thinking argues that any system, whether it be an organization, a work group, or whole city, will function at its best when both social (e.g., people, skills, culture, attitudes) and technical (e.g., software, IT, tools) are jointly designed and the interactions between these parts are considered together (e.g., Clegg, 2000). Socio-technical systems thinking enjoys a rich history within management and organizational psychology, with the philosophy emerging from seminal studies of technology introduction in traditional UK industry in the 1940s and 1950s (Davis, 2019). Researchers at the Tavistock Institute recognised the disruption that new machinery could have on existing social structures and work processes (e.g., in coal mining) and the effect these changes had on employee outcomes such as motivation and absenteeism, in turn affecting the effectiveness and productivity of the technology itself (e.g., Trist and Bamforth, 1951). This led to the development of ideas surrounding the value of engaging end-users in the design of technology, designing in opportunities for employees to have control over the management of their work and to resolve problems as they arise (Daniels, Le Blanc & Davis, 2014). These early studies also illustrated the difficulty in designing a change to one part of a work system without considering the potential implications of this; with interdependencies throughout the system (Eason, 2014), a change in one aspect often leads to unintended consequences elsewhere (e.g., the coal mining machinery reduced the variety in the job for workers and reduced motivation/increased absenteeism, limiting the utility of the machinery itself). Socio-technical systems thinking has had a lasting impact on the areas of job design (most notably in the ideas of semi-autonomous workgroups and task variety), in addition to change management, technology implementation, and human factors in safety critical industries (e.g., aviation) (Davis, 2019; Eason, 2014; Mumford, 2006).

The key difference between technical and socio-technical systems lies in what drives their behaviour: the laws of physics determine the behaviour of technical systems whereas the behaviour of socio-technical systems is governed by human behaviour and human interactions with technology (e.g., Mumford, 2006). Factors that influence human behaviour relate to characteristics of individual people such as personal preferences and competencies, training, career development, and job satisfaction (Robinson et al., 2005). In addition, individuals' behaviours are influenced by characteristics of the groups within which they work. These may include team working, leadership, communication, coordination, capability, and shared mental models (Salas et al., 2005). More broadly, behaviour may also be influenced by characteristics of the social systems within which individuals live, such as organisational culture, social networks, and leadership (Brass et al., 2004; Davis & Coan, 2015). When applying principles of systems engineering to the design of socio-technical systems, a key challenge lies in how we accommodate human behaviour in system designs and predict its influence on overall performance in alternative system configurations (Crowder et al., 2012).

A further important difference between technical and socio-technical systems is that the latter are not realised in the same way as physical products. For this reason, the modified systems design vee model introduced in McKay et al. (2018) is more appropriate here than traditional systems engineering vee models that include product realisation. This modified model, as shown in Figure 3, highlights key aspects of systems engineering, such as the flow-down of design requirements and the flow-up of design solutions, and provides the basis for a design process that zig zags between functional aspects of the design (on the left) and solutions (on the right). In this chapter, we focus on four aspects of socio-technical systems that distinguish them from technical systems and so influence their design:

i) when formulating design requirements, how good quality requirements for a socio-technical system might be defined;
ii) when formulating (sub)system architectures, how alternative allocations of function between people and machines might be described and evaluated;
iii) when verifying proposed design solutions, how human behaviour might be incorporated within simulation tools; and
iv) when evaluating an overall solution, how the resilience of the whole system might be considered, especially with uncertain conditions and in uncertain environments.

The chapter builds on Clegg's (2000) principles for the design of socio-technical systems. His seven meta-principles govern the overall orientation of system design activities; the overall structure of this chapter is organised in a similar way, see Table 1.

Table 1: Meta Principles of Socio-Technical design (Clegg, 2000)

Principle	Description (based on Clegg (2000))
1) Design is systemic	The use of a well-defined process helps ensure that a systematic approach is used but this does not necessarily guarantee a systemic approach. However, the process ensures that systemic solutions are created by considering social and technical aspects of the design equally through the entire process.
2) Values and mindsets are central to design	Over time, the use of such a process has the potential to facilitate cultural change and so the values and mindsets of system designers, users, and other stakeholders are critical. As such, the design process can be seen as dynamic, and likely to both influence and be influenced both socially and by its environment.

3) Design involves making choices	The vee model includes the consideration of alternative solutions and so encourages the divergent thinking essential to provide the choices design involves.
4) Design should reflect the needs of the business, its users and their managers	The involvement of multiple stakeholders, in the formulation of the initial capability statement and throughout the design process, ensures that their needs are reflected in the design process.
5) Design is an extended social process	As the chapter unfolds, the importance of including a range of people in the design process becomes clear and as does the perspective of design as a socially shaped extended social process.
6) Design is socially shaped	
7) Design is contingent	The evaluation of design directions and solutions by multiple stakeholders, against design requirements and throughout the process, reflects design being contingent with no one best solution.

2 Background theory

We consider four aspects of background theory:

(1) Multi-team systems: the kinds of systems on which this chapter and its case study focus.
(2) Allocation of function: for example, in the formulation of solution architectures, there are choices related to how work is divided between technology and people.
(3) Simulation: in the development of solutions, alternatives can be visualised and predictions made about how whole systems (social and technical) might behave.
(4) Resilience: a characteristic of any socio-technical system that determines its overall capability to meet its design requirements under uncertainty.

2.1 Multi-team systems

In systems engineering, the design process is not completed by single engineers, from a single organization, who complete separate and clearly demarcated activities; rather, it is a messy and complex process (Xu et al., 2003). Even in the most simplified design system a designer can expect to work as part of a team, and, in the simplest circumstances, have only the task in hand to work on, along with a clear goal, and colleagues who are also focused solely on the same design project (Crowder et al., 2012). In reality, contemporary organizations usually operate in complex matrix structures in which team members are organized by both discipline and design component, and often report to different line managers with different priorities (Lechler & Dvir, 2010). Typically, a designer will be working on multiple projects at once, and working as part of concurrent engineering teams, in which they are dependent on others from different disciplinary backgrounds for information and advice (Hughes, 2017). Effective team work under these conditions is contingent not only on designers having strong *technical* competence, but also on designers having highly refined *interpersonal* skills (Robinson et al., 2005). Design requirements can change radically and/or incrementally throughout the design process and so, for individual designers, knowing what information to share, at what point, and with whom, is pivotal to their personal success (Busby, 2001). Moreover, it is imperative that the team more collectively shares a clear understanding of what is required, by when, and who in the team will complete each element of the design activity (Mathieu et al., 2000).

In previous organizational and simulation research to understand factors underpinning high performing and robust design engineering teams, we studied different design teams each responsible for different components of an overall design (see Crowder et al., 2012, for details).

While the overall team had an overarching design goal, each designer also had different types and levels of technical knowledge and competence, competing project pressures, different individual priorities, and varying time to dedicate to the project. Within the team, members also had different trust levels, which affected their willingness to share information. Trust was therefore an important interpersonal antecedent to effective communication. In addition, the performance of the teams in terms of the speed of work, its cost, and its quality were related to the extent to which the team shared an understanding of requirements and who they needed to communicate with and when. Evidently then, successful navigation of the social elements of this design process are as pivotal as the technical elements in many ways, not least because diary study research has shown that even the most technical of activities comprise substantial social elements (Robinson, 2012).

2.2 Allocation of function

The descriptions of the design processes above highlight the role of technologies and information systems in mediating interactions between individuals and teams, in addition to storing and managing requisite knowledge. The design of any socio-technical system involves decisions regarding the division of tasks and activities, both between individuals, groups, and subsystems, and between human actors, technologies, and software (Eason, 2014; Mumford, 2006). Such decisions involving "the allocation of functions or tasks between the humans and machines in a system" (Clegg, 2000) is described as *allocation of function* and may typically be considered during design to determine whether humans or machines are better suited to undertake particular discrete tasks, functions, or roles (Grote et al., 2014). Allocation of function has a rich history within ergonomics, human factors, and psychology disciplines, particularly within military domains, aviation, human fatigue in safety critical industries, computer systems design, and manufacturing (Waterson et al., 2002).

Research exploring allocation of function within socio-technical systems has identified three main approaches to such divisions (Challenger et al., 2013). In *function allocation by substitution*, tasks are assigned based on the capability of the human or machine, usually made as a static decision during the design process. On the other hand, in the *left-over approach* designers attempt to automate as much as possible to reduce the potential for human error or inefficiency, with any tasks or operations deemed to be too difficult or uneconomic to automate allocated to humans. Finally, in the *complementary approach,* design is dynamic and adaptive, with tasks allocated to capitalize on the capabilities of human and machines within the system, permitting functions to move dynamically between both parties dependent upon context, enabling interventions to be made and for overall system performance to be maximized. The research corpus demonstrates that each of these approaches towards allocation of function has implications for the design of the system as well as overall performance and resilience (Strain & Eason, 2000).

As with many other areas of systems design, there is a danger that decisions regarding allocation of function are made implicitly at early stages of the design process, based upon designers' assumptions and preconceptions (Challenger et al., 2013). Such a priori decisions run contrary to socio-technical systems principles regarding minimal critical specificity (Cherns, 1976, 1987). These principles suggests that the decisions regarding how work, tasks, and outcomes are achieved should be specified as little as possible, allowing those within the system to establish the most effective and productive ways of achieving set outcomes given necessary constraints, such as safety and cost parameters (Davis, 2019). Decisions made by experts early in the design process also run the danger of techno-centricity whereby technology is seen as the solution to any given problem (Clegg & Shepherd, 2007). As a result, design decisions reflect the biases and convenience of those same experts, reducing the choice and design options available to users and clients (Clegg, 2000; Eason, 2014). Formally including acknowledgment of allocation of function helps to mitigate such undesirable outcomes and also addresses calls by researchers to address ways in which accountabilities and interactions between upstream (e.g., designers, the organization) and downstream (e.g., users) actors are distributed (Grote et al., 2014). Recognizing that there are different ways to allocate functions between and within system components, and that this may result in multiple design options, requires a process to evaluate the potential costs and benefits of each scenario (e.g., in terms of short- and long-term system performance, safety, resilience, and job satisfaction). Agent-based and other forms of computer modelling present themselves as promising methods for simulating allocation of function within work systems, particularly with regards to emerging technologies such as autonomous vehicles, artificial intelligence, and robotics (e.g., Hettinger et al., 2015; Weyer et al., 2015).

2.3 Simulation

Agent-based modelling and simulation (ABMS) is a computer simulation technique in which individual agents act in accordance with rules which govern their behaviours and enable interactions. Interactions may occur between agents, and also between agents and their environment, leading to complex, collective, emergent outcomes (Hughes et al., 2012). This capability to incorporate a range of variables, and to examine complex problems with non-linear outputs and feedback loops, makes ABMS an ideal approach for modelling complex socio-technical systems (Crowder et al., 2012). For instance, ABMS has been used to simulate transport decisions in supply chains (Sha & Srinivasan, 2016), the management of major emergencies (Hawe et al., 2012), crowd behaviour in transport terminals (van der Wal et al., 2017), and, of relevance to this case study, team work in engineering (Crowder et al., 2012).

In the case study introduced later in this chapter, we have several teams working together in a multi-team system (Marks et al., 2005) to manage a restaurant. For instance, we have a waiting team taking customers' orders and a kitchen team preparing the orders. The teams therefore have inter-dependent goals and must communicate with each other and coordinate their activities to deliver the service (i.e., meals to the customers), which is an ideal scenario for ABMS to simulate (Hughes et al., 2012).

2.4 Resilience

The resilience of a socio-technical system concerns its ability to deal with uncertainties and disruptions, such that changes in its operational, regulatory, and/or economic environments do not cause operational down-time or threaten its ability to achieve its purpose (Pieniazek, 2017). First, by dealing proactively with threats before they result in actual disruptions, a resilient system can reduce the effects of the adversity when it actually presents as a disruption (i.e., the disruption results in smaller consequences than would have occurred in a less resilient system), or can prevent the disruption from materialising altogether (Hollnagel, 2014; Pieniazek, 2017). A resilient system is one which has resources to anticipate (identifying potential disruptions), plan (consider system changes to address threats and disruptions), implement (make the changes), and learn (reflect on lessons) to address disruptions if and when they occur (Hollnagel, 2014; Pieniazek, 2017). Second, given resilient capability enables a system to deal with uncertainty, it also includes a responsive capability to deal with disruptions on the spot which are affecting its operations in real time (Pieniazek, 2017). Hence, there are resources within the system to monitor (detect disruptions in the environment and their impact on current performance) and respond (mobilise resources to return to normal operations).

In systems that are designed, opportunities become available to specify required levels of resilience. One way of ensuring system resilience is to increase reserves, for example, of human, material, and financial resources necessary to protect and guarantee the normal functioning of an organisation within the system (Bourgeois, 1981). Resilience can be designed into the system within its components or overall architecture, or through processes and behaviours that the system invokes when its performance is or could be under threat. This design requirement must be traded off against other requirements. For example, continuous improvement approaches used to improve the performance of manufacturing systems involve removing waste and non-value adding activities to create so-called 'lean' solutions. While such systems have increased outputs with reduced costs (Shah & Ward, 2003), this may be at the expense of system resilience. In manufacturing and supply chain systems, the integration of agile methods into lean systems helps improve resilience by improving the system's ability to respond to changes in demand (e.g., Kundu et. al., 2008). These approaches, however, tend to conceptualise the system architecture as being fixed. This may be appropriate in circumstances where the system architecture is tightly coupled with expensive equipment and infrastructures. However, even in such situations, given that disruptions can originate in a system's internal and external operating environments and change a system-state (e.g., loss of data, human error, systems failure, natural disasters), the introduction of socio-technical systems mindsets has the potential to deliver significant benefits.

While lean processes are likely to result in higher peak performance levels, because during routine operations all resources are focused on present performance, in periods where disruptions occur, resilient systems perform best in the long term and with more stability. Hence, allocation of function within a system has implications for its resilience capability. For example, fully automated systems would yield higher performance in normal operations than a similar system including some human operator involvement; however, the fully automated system may be less resilient to unexpected change or errors, leading to decreased performance in the medium term or an unacceptable risk when managing exceptions in safety critical industries (Casner et al., 2014). An example of this is aviation, where the technology has long existed to automate the take-off and landing of commercial airplanes (Billings, 1997). However, the relative efficiencies and reduction in human errors that higher automation (and reduced roles for pilots and flight crew) may yield in 'normal' circumstances, need to be balanced against the need to retain flight crews' expertise. This expertise, which includes skills gained through experience and focus, allows flight crews to intervene in abnormal situations, such as when technical malfunctions occur, and forms the rationale behind complementary approaches where function allocation is shared between machine and human (Casner et al., 2014). For this reason, while designers typically design for performance, designing for resilience requires acknowledging the whole system and the environment within which it operates.

3 Multi-team system design case study

A fictitious pizza restaurant, *Pizza Pizzazz*, is used here as a case study. Restaurants have various common expectations that they need to fulfil in order to provide a dining experience for their customers. These typically include a way for customers to decide what to eat and drink, a place for customers to sit, preparation of food and drink, delivery of food and drink, clearing of plates and glasses, and a way to pay (e.g., Amigo et al., 2008). There are many ways that restaurants can decide to organize themselves in order to meet these expectations and to decide who or what (i.e., people or technology) delivers each part. We present this restaurant as a demonstration of the complexity present in seemingly simple organizational systems, the implications that different design decisions can have, and the interactions within such systems. Pizza Pizzazz is a "casual dining" pizza restaurant chain that aims to provide customers with an affordable and efficient dining experience. The business model is based upon turning tables quickly and providing a consistent experience across the chain.

3.1 System organisation and the allocation of functions

Pizza Pizzazz has traditionally organized itself as a multi-team system (De Vries et al., 2016), with different teams responsible for delivering different parts of the customers' dining experience. A *front-of-house team* is responsible for greeting customers and showing them to a table. They are divided into sections to cover specific tables within the restaurant. A *waiting team* is responsible for taking customers' food orders using a paper notepad, delivering their food, checking customers are satisfied, and taking payment. A *bar team*, covering all of the tables in the restaurant, takes customers' drinks orders, makes the drinks, and delivers them to the tables. The *kitchen team*, covering all of the tables with members allocated specific custom pizzas, prepares and cooks the food. The *cleaning team* clears and cleans tables and the restaurant, and prepares new tables. Each team is managed by a supervisor and has its own goals, pressures, and priorities (e.g., Crowder et al., 2012). This restaurant is therefore an excellent example of a complex socio-technical system, with people interacting with technology and infrastructure, and following processes to achieve goals within the organization's cultural context (Davis et al., 2014).

In delivering a service of this kind, the organization has made choices about how things are done. While some such choices may have been made inadvertently, they are important because they have consequences (sometimes unintended) and ripple effects for other parts of the system. For instance, choices about the organization of job roles will impact on the training and competency requirements of team members, as well as on the communication requirements that are needed across sub-teams (Crowder et al., 2012). They may also impact on the work-flow. For instance, a table cannot be found until a front-of-house team member is available, which could lead to queues at the desk, while other team members (e.g., those in the kitchen) might have nothing to do. Each team member will be balancing competing tasks and is dependent upon information and action from members of other teams. For example, the kitchen team is dependent on the waiting team passing customer orders through accurately and the front-of-house team balancing customer bookings to stagger demand. The kitchen team also has to cook dishes for multiple tables at once and communicate among themselves to ensure that dishes for a single table, with different preparation times, are all ready for the waiting staff to deliver simultaneously. The waiting team shares a fixed proportion of their individual tips with the other teams, who split the tips evenly between their team members. This reward structure incentivizes the waiting team to provide a high quality customer experience, without rewarding individual performance elsewhere.

Pizza Pizzazz are keen to keep improving their efficiency and effectiveness, so using the *System Scenarios Tool* (Hughes et al., 2017) we are able to consider the socio-technical implications of two alternative ways of organizing the service, as outlined in Table 2.

Table 2: Two alternative design alternatives for the Pizza Pizzazz case study using the System Scenarios Tool (Hughes et al., 2017)

Factor	'As is'	'To be'
How does it work? (This refers to the characteristics of the work carried out within the organizational unit.)	- Overall service is traditional, delivered by small sub-teams organised by job role (e.g., front-of-house team, waiting team, bar team, kitchen team). Each sub-team is managed by a supervisor. - Pizzas are prepared freshly to order.	- Overall service is delivered by a small number of staff in transferable roles, and is dependent on customer input via some fundamental technologies. - Pizzas are mass-prepared in a central kitchen and delivered to store for heating.

Goals (This refers to the goals of the system under examination and to the metrics which are used to assess the performance of the system and the people working in it.)	“To provide customers with an affordable and efficient dining experience”. - Sub-teams have individual goals (e.g., the front-of-house team need to get people seated as quickly as possible; the waiting team need to ensure customers receive food, bills, and can make payments as soon as they are ready). - Waiting staff keep a fixed proportion of their individual tips, incentivising high quality customer service. They share the rest with the other sub-teams, split evenly between team members.	“To provide customers with an affordable and efficient dining experience”. - The overall goal is increased customer foot-fall. Workers aim to provide start-to-finish service as quickly as possible to keep tables turning. - Tips are divided equally by the payment system between all staff members regardless of number of tables served or level of tip provided by individual tables.
Processes (This refers to the work processes and working practices that are in use in the system. It includes the organizational structure and the ways in which the work is organized.)	- The customer experience is sequential and each sub-team has a process determined by their role in this sequence, which maps who speaks to whom, about what, and when.	- The customer experience is sequential but the customer has more involvement in the speed of this process (e.g., how quickly they find seating, order, and pay).
Technology (This refers to the technologies, tools, and equipment used within the system, and can include both hardware and software.)	- Bookings and seating are manually recorded using a paper diary. - Orders are taken manually using pen and paper, and are physically transferred to other sub-teams (e.g., waiting team to kitchen team). - Payment is taken using an electronic till system (cash or card).	- High-tech service. Technology is used for ordering, paying, and communicating with staff.
Infrastructure (This refers to the physical infrastructure of the system, for example including its buildings and the physical assets. But it can also include the financial infrastructure, such as the business model in use.)	- The physical space is set as a traditional restaurant with numbered tables varying in size to accommodate different sizes of customer group. - The business model is based around turning tables quickly and consistent service.	- Traditional restaurant with high-tech facilities is required (flexible number of ordering units to enable different group sizes). - A new central kitchen for pizza preparation is required (or a supply chain partner). - The business model is based around turning tables quickly and consistent service.
People (This refers to the people working in the system and also the key stakeholders – including customers – and includes their attitudes, behaviours, skills, and competencies.)	- Staff skills are specialised (e.g., chefs, waiters, bar staff, cleaners). - Staff require differential, role-relevant training and competencies.	- Staff need training in the full range of roles (though there are only a few of these), and they need transferable competencies. This increased skill level may necessitate a higher wage.
Culture (This refers to the shared norms, beliefs, and values that permeate the system. This can be local to the system and/or shared more widely outside the system.)	- The environment is pressurised as work is fast-paced. Staff are empowered to resolve issues within their role, but are dependent on others from different groups for their tips. Waiting staff work hard for tips, but others feel less pressure to do so as they receive tips regardless.	- The environment is pressurised and fast-paced. Staff need to communicate to avoid duplication in orders and aspects of the process, but this is helped by technology, and they can step in to help each other at busy times. All staff need to work hard to maintain the reputation and to receive tips, but this can also create loafing among some colleagues who can be carried by more motivated or faster workers.

System benefits (examples)	- Clearly allocated roles and responsibilities. - Clear command and control structure. - Service process is unambiguous. - Staff training requirements are minimised because roles are specialised (only need to train in one aspect of the service).	- Lower staff costs and a reduction in roles required (and associated training). - Quick and reliable communication between the customer and staff (relies on customer input rather than accurate recording of customer choice). - Can deal with more customers, more efficiently, and at the customer's pace.
System costs (examples)	- Bottlenecks can arise at busy times - Customers voice frustration at seeing some staff appearing to lack work while they are not being served effectively.	- Additional costs for the installation and maintenance of service technology. - New facilities (e.g., a new central kitchen) are required.
System risks (examples)	- Low resilience in case of staff absence (staff are not trained to cover). - Communication errors across sub-teams can result in service delays.	- Service is dependent on technology, so low resilience to technological outages or breakdown. - Fewer staff available in non-routine situations.

For instance, Pizza Pizzazz is interested in comparing their service approach with that of their rival Techy Mexican Food Company (TechMex) where customers arrive and locate a free table to sit at rather than wait to be seated. Customers are expected to place their food and drinks orders through an at-table tablet or their smart phone. A waiting team then delivers the customers' food and drinks directly to their table. Different members of waiting staff may deliver different parts of a customer's order. Customers are asked to use the tablet or their smart phone to place additional orders, to summon help, and to pay their bill. The kitchen team are responsible for fulfilling specific table orders as they are received (i.e., preparing all dishes for that table). The majority of dishes are prepared in a central kitchen and delivered to individual restaurants. This reduces the preparation and cooking time in the restaurant and means that one staff member can prepare multiple dishes simultaneously for one table. When a customer pays their bill, it triggers an alert for a member of the waiting team to clear the table ready for the next customer. Tips are divided equally by the payment system between all staff members regardless of number of tables served or value of tip. By organizing the service in this alternative way, TechMex has reduced in its wage bill, and has increased the speed at which it turns its tables thereby increasing capacity. However, as with any service redesign or introduction of advanced technology, there are trade-offs to be made (Clegg & Davis, 2016). We outline the challenges that a move to this way of working may pose in Table 2 using the Systems Scenario Tool (see, Hughes et al., 2017, for a discussion regarding the application of this tool to different contexts).

3.2 System evaluation

Evaluation is an essential aspect of any design process (see Clegg, 2000; Suh, 1990). However, the measurement systems and metrics used to evaluate a system's effectiveness can be controversial, because appropriate metrics depend on the needs, priorities, and requirements of different stakeholders and the measurement system used is likely to influence the behaviour of the system under consideration (Mannion & Braithwaite, 2012). Metrics may be similar, subtly different, or altogether incongruent with each other (Hughes et al, 2017; Robinson, 2018). For example, a manager tasked with the goal of resource optimisation may regard the best process as one that delivers solutions as quickly as possible using minimal resources. On the other hand, people whose lives will be affected by the *operation* of the designed system (e.g., the waiters and bar tenders) may be more interested in ease of use and its impact on their abilities to meet their goals (such as being given a tip by the customer for high quality service). Often, however, the different stakeholders will be fixated on their own goals, and may not realise that their priorities differ to those of others. This can be problematic as it can lead to a system that only meets the needs of certain stakeholder groups. Additionally, it can mean that subsequent design choices, or decisions to adapt the system to alleviate problems for one group, inadvertently lead to difficulties for other groups.

Consequently, the System Scenarios Tool (Hughes et al., 2017) advocates the need for different stakeholders to be given the opportunity to feed into the criteria that the system is evaluated against, because this can be useful in helping to make different system priorities explicit. In addition, it is suggested that the identification of multiple evaluation criteria can help provide a more holistic framework to evaluate the system against, as well as alternative system configuration scenarios (for instance, how do alternative configurations fare overall in an economic downturn, or where particular metrics are given greater weighting?). In this chapter, we now introduce three ways of evaluating alternative system designs. In Section 3.2.1 we evaluate the as-is and to-be system designs with respect to Clegg's (2000) content principles; in Section 3.2.2 we consider the use of computer simulations to predict the behaviours of alternative designs; and in Section 3.2.3 we consider the assessment of system resilience.

3.2.1 Evaluation of the Pizza Pizzazz design alternatives using Clegg's (2000) content principles

A comparison of the alternative system designs provided in Table 2 is given in Table 3. In this section we use Clegg's (2000) content principles to evaluate these designs in more detail.

Table 3: Evaluation of the alternative system designs for the Pizza Pizzazz case study using Clegg's (2000) content principles

Principles	**'As is'**	**'To be'**
Principle 8: Core processes should be integrated	The service process is unambiguous but bottlenecks can arise at busy times because the process is split across the artificial organisational boundaries of the sub-teams.	The service process remains unambiguous but individual workers aim to provide a complete, start-to-finish, service process. In developing this alternative, it will be important to ensure that the information systems to be used match the requirements of the service process.
	In both cases, an early step in developing the design needs to be the design of the logical processes that will be used in each scenario.	
Principle 9: Design entails multiple task allocations between and among humans and machines	The use of technology is limited to the payment system.	Technology is used in all stages of the process; this gives customers some control over the speed of their service experience.
	Each scenario includes allocation of tasks between people and technology. The 'as-is' scenario is more heavily reliant on people and the 'to-be' scenario is more reliant on technology.	

Principle 10: System components should be congruent	Congruence with surrounding systems and practices is achieved through the payment system.	
	The emphasis on manual processes in the service delivery and the preparation of pizzas to order minimises the need for integration with external systems.	The use of technology for ordering, paying and communicating with staff creates new requirements for integration with external systems such as mobile technologies. The reliance on mass-produced pizzas creates a need to integrate with the central pizza kitchen which may, in turn, create a need for new IT systems. Integration failures will put the service delivery at risk.
Principle 11: Systems should be simple in design and make problems visible	If we regard the as-is system as the baseline then the to-be design is likely to be regarded as simpler for service users and staff but the design of the underlying to-be system is likely to be more complex. This evaluation highlights the importance of visibility and the need to consider it further in both designs.	
Principle 12: Problems should be controlled at source	Systems are most effective when problems are easy to resolve as they arise.	
	The high specialization of roles promotes a depth of knowledge regarding the likely bottlenecks in service, enabling these to be pre-empted, as well as familiarity with strategies and ways of resolving these when problems do arise. For example, kitchen staff will be aware of the dishes that may prove most popular given the current weather conditions and either sell out or create bottlenecks in cooking. For example, pizzas sell better on a warm evening than a wet evening and there is limited capacity to prepare the dough fresh to order. Staff can either anticipate the spike in demand and prepare a greater number of bases early in service or ask waiting staff to influence subtly diners' menu choices to spread demand across dishes.	The engagement of waiting staff across the whole range of service activities may be expected to increase the range of problems that staff members may recognise. However, the depth of knowledge and experience in a given specialism will be limited, potentially reducing the likelihood that staff are able to successfully resolve the problem identified. The pre-prepared nature of the meals limits the scope for kitchen staff to intervene to resolve or respond to problems during service. For instance, if the pizza oven failed, they could not work with the range of raw ingredients to produce a greater number of pasta-based dishes or salads.
Principle 13: The means of undertaking tasks should be flexibly specified	The various roles are clearly defined. To ensure consistent quality and experience, the service and food preparation activities are also highly specified. The emphasis on consistency of experience limits opportunity for staff to develop their own style or ways of delivering the service.	The individual tasks that staff undertake are highly specified and adaptation of these is limited by the constraints imposed by the technology (or pre-prepared nature of the food). The organization of staff within the service team is much less rigid, enabling a greater degree of autonomy over how each team operates, adapting to the interests, skills, and speed of the staff on any given shift. The team can self-organize and decide who undertakes particular tasks and change this approach over the course of a shift.

In addition to Clegg's (2000) content principles, we introduce two forms of design evaluation that are especially appropriate for socio-technical systems: (1) simulations, and (2) assessment of resilience. To conduct these evaluations, we need more detailed descriptions of the two designs. Service blueprinting is a widely used tool for describing services and has been applied to engineering service systems (McKay & Kundu, 2014). The diagrams in Figures 1 and 2 are service blueprints for the two designs. Service blueprinting highlights delivery processes and their visibility to customers; people and technology are not explicitly shown. Although the ‘to-be’ design appears simpler than the ‘as-is’ design (because it as it has fewer tasks and flows) the detailed process may be more complex due to the supporting technological infrastructure required. This introduces its own risks and impacts system resilience.

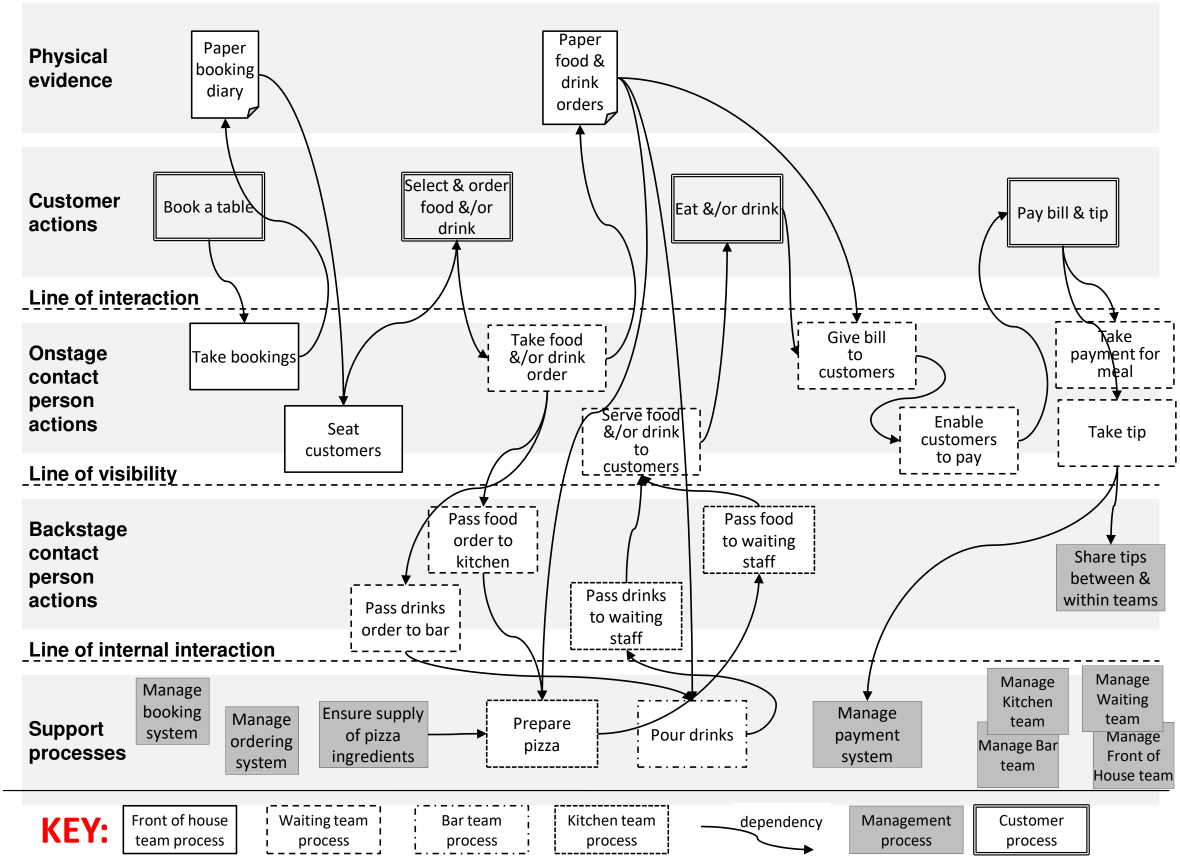

Figure 1: AS IS: “To provide customers with an affordable and efficient dining experience”

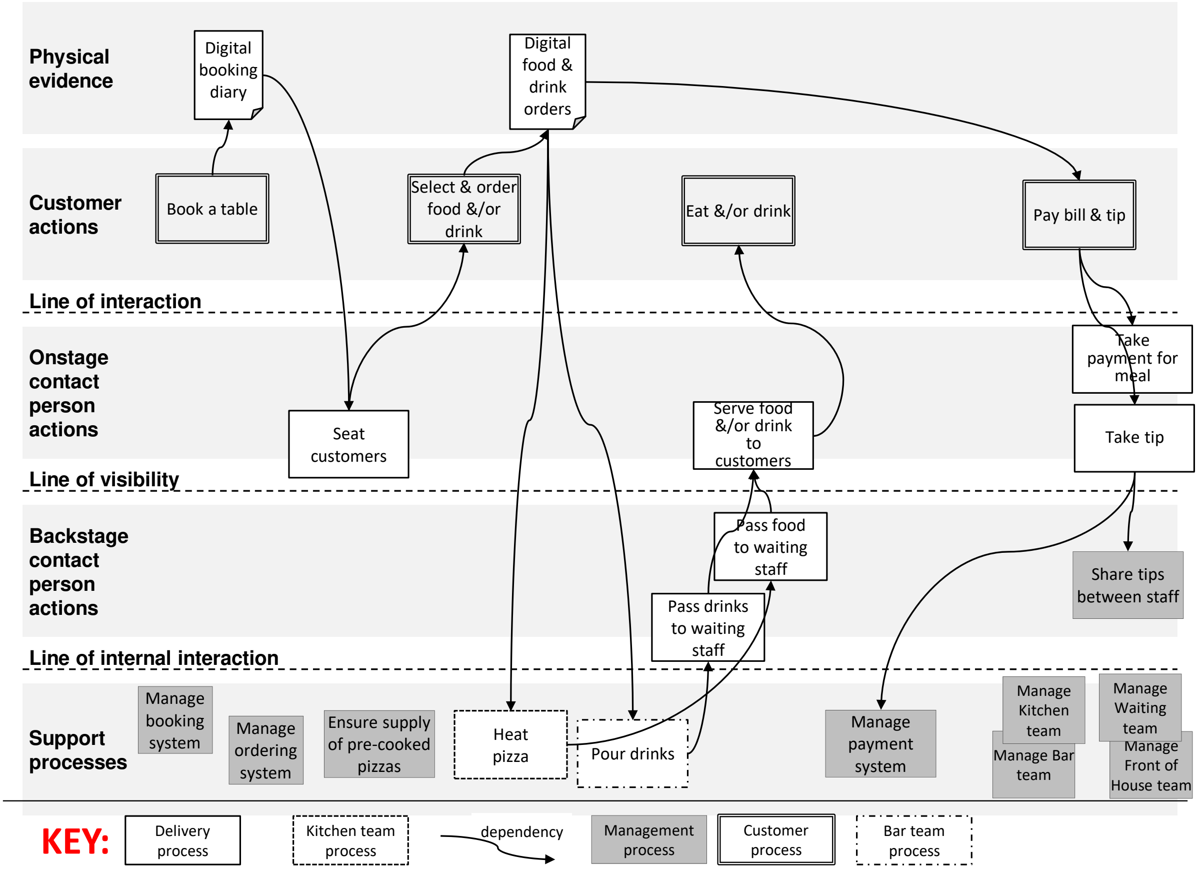

Figure 2: TO BE: "To provide customers with an affordable and efficient dining experience"

3.2.2 Simulating system behaviours

Given the complexity of the Pizza Pizzazz multi-team system, ABMS would be ideally suited to studying its function and systematically comparing the as-is and to-be scenarios. The capability to incorporate a range of variables, and to examine complex problems with non-linear outputs and feedback loops, makes ABMS an ideal approach for modelling complex socio-technical systems (Clegg et al., 2017).

In the current scenario, we have three teams working together in a multi-team system (De Vries et al., 2016; Marks et al, 2005) to deliver a quality dining experience to restaurant customers. All three teams have inter-dependent goals and must communicate with each other and coordinate their activities to deliver the customer service. We could simulate this working environment using a hypothetical ABMS model. In the first 'as-is' scenario model, customers would enter the restaurant and explain their requirements to the front-of-house team who would seat them at an appropriate table and notify the waiting staff. The waiting staff would then take the customers' orders and communicate these to the bar staff and the kitchen staff who would prepare the customers' meals and inform the waiting staff who would deliver them to the customers. On finishing their meals, the customers would communicate with the waiting staff who would bring the bill to the customers and process their payment. In the second 'to-be' scenario model, customers would seat themselves on arrival and place their order via an electronic app on the restaurant's tablet or their own smart phone. Staff members would be alerted to the order and one staff member would be allocated to the customers' table. This staff member would then be responsible for all aspects of the service, in a ‘start-to-finish’ manner, from preparing the drinks and reheating the centrally prepared pizzas, to delivering the meal to the customers.

Simulations would then be run for each of these two scenarios under a number of different operating conditions, varied systematically using an experimental approach (Robinson, 2016), to examine the effect on performance in terms of service time, operating cost, and customer satisfaction. First, in the control conditions, to establish the baseline performance levels, simulations of the two scenarios would be run under routine conditions. Here, the hypothesis would be that performance would be higher in the current ‘as-is’ scenario, as specialisation generally leads to higher peak performance during optimal conditions (Cressy et al., 2007). Once these baseline performance measures had been established, a number of experimental conditions would be run each introducing a number of non-routine conditions to the system. For example, one member of the customer group could have a food allergy, requiring non-routine food preparation (i.e., product variance) [Condition 1]. Alternatively, a member of the restaurant’s staff could be removed from the model, to simulate short-staffing due to sickness (i.e., service variance) [Condition 2]. Similarly, customer numbers could be varied from the baseline control condition, either reduced to simulate a quiet day (i.e., reduced system load [Condition 3]), or increased to simulate a busy day (i.e., increased system load [Condition 4]).

Each simulation would be run multiple times, generating considerable data for analysis and enabling performance distributions to be identified. The accompanying hypotheses would be that the ‘as-is’ team configuration would out-perform the ‘to-be’ configuration in Condition 1 because the kitchen staff would be able to customise orders, whereas this would be more difficult with pre-prepared pizzas (in Condition 4) because specialisation permits more efficiency for busy periods. Meanwhile, the ‘to-be’ configuration would out-perform the ‘as-is’ configuration in Condition 2, because other staff members are able to step in to fulfil other roles, unlike the specialists, and also in Condition 3, as the quiet day enables the kitchen to run at a lower capacity and the chef does not have to work continually. However, the ‘as-is’ configuration would perform best in Condition 4, due to the extra efficiency enabled by specialisation.

3.2.3 Assessing system resilience

Different allocations of functions will also have different implications for system resilience and the different system architectures (‘as-is’ and ‘to-be’) can each be assessed accordingly. Resilient outcomes can be measured in three ways. First, the smaller the variability around mean performance, the more resilient Pizza Pizzazz would be (Pieniazek, 2017). The focus would be on performance variability, rather than the baseline performance level, because striving to deal with adversity can actually improve performance (Lengnick-Hall el at., 2011; Sutcliffe & Vogus, 2003). Second, during a disruption, the quicker the system can resume normal functioning, the greater its resilience (Bodin & Wiman, 2004). Third, given that resilience not only involves detecting and responding during a disruption, but also detecting and making changes proactively based on potential future disruptions, the lower the down-time of functions over time, the greater the resilience (Hollnagel, 2014; Pieniazek, 2017).

Measuring the resilience capability of Pizza Pizzazz will involve assessing its ability to anticipate, plan, implement, learn, monitor, and respond (Pieniazek, 2017). Given that resilience is about using and sharing resources across boundaries to create slack (Kahn et al., 2018; Pieniazek, 2017), the different sub-teams within Pizza Pizzazz could vary in terms of the degree to which they share with or withhold resources from the other teams, thus enabling or inhibiting resilience. Furthermore, as part of anticipating and monitoring, both potential and actual environmental demands are likely to be detected within the system and these can be recorded, assessed, and measured in their own right alongside resilience assessment. As variable environmental demands affect the different system parts, assessing the resilience of Pizza Pizzazz as a whole would likely not be sufficiently nuanced (Kahn et al., 2018).

4 Discussion: Realising multi-team system design processes

In this section we use the Pizza Pizzazz case study to develop thinking on Clegg's (2000) process principles (see Table 5). Design processes are a key means by which one-off socio-technical system design processes can be systematised within organisations and practice. An important difference between technical and socio-technical systems is that the latter are not realised in the same way as physical products. For this reason, the systems design vee model introduced by McKay et al. (2018) is more appropriate here than traditional systems engineering vee models that include product realisation. The systems design vee model in Figure 3 highlights key aspects of systems engineering (the flow-down of design requirements and the flow-up of design solutions) and provides the basis for a design process that zig zags between functional aspects of the design (on the left) and solutions (on the right).

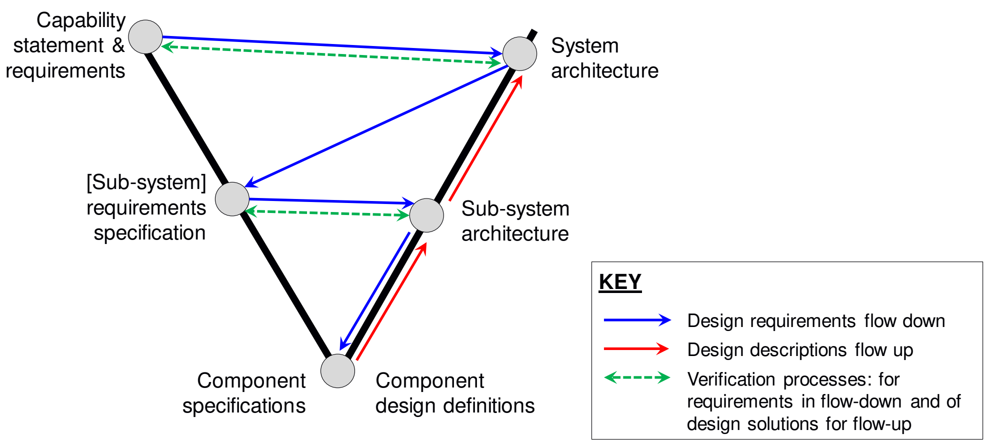

Figure 3: Systems design vee model (adapted from McKay et al., 2018)

The vee model in Figure 3 can be expanded into a stage-gated design and development process. For simplicity, in this chapter we consider a two level vee where component and sub-system designs are merged. Figure 3 includes the three processes shown in Figure 4: (1) the flow-down of design requirements; (2) the flow-up of design solutions that are integrated into sub-systems and, ultimately, a whole socio-technical system that delivers the capability statement; and (3) the verification processes where design solutions are evaluated against design requirements. In Figure 4, the flow-down of requirements and flow-up of designs are shown as process steps and the verification processes are shown by the diamonds; details of the verification processes are given in Table 4.

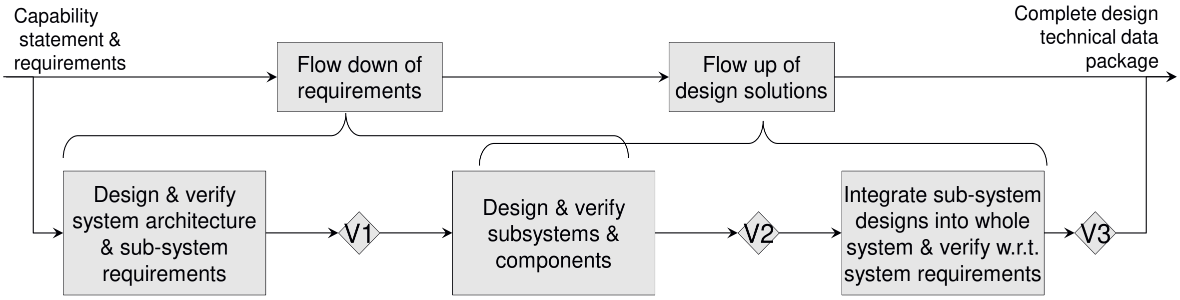

Figure 4: Derived process for a system with one level of sub-systems and components (reproduced from McKay et al., 2018)

Table 4: Details of the verification process in Figures 2 and 3

	Design element being evaluated	**Evaluation with respect to**
V1	System architecture (and sub-system requirements)	Initial capability statement
V2	Each subsystem and component design	Sub-system requirements
V3	Overall system design	Initial capability statement

In the remainder of this section we consider how the case study relates to this generic design process and Clegg's (2000) process principles using the process steps in Figure 4. We begin by elaborating each of the process steps shown in Figure 4 and then reflect on how these relate to the process principles.

Design & verify system architecture & sub-system requirements: Each scenario has the same initial capability statement but this is realised using different system architectures and different allocations of function between people and technology. The design descriptions that would result from this step would include system architectures, which could be used to inform the structure of computer simulations, but precisely what these architectures would be remains an open question. For example, Figure 5 shows three possible system architectures, two each for the as-is and to-be designs. From the customer perspective, it can be argued that the architecture is the same in each case, whereas if the system architecture is defined from the perspective of the service deliverer then there are clear differences. In all three cases, an indication of which sub-system will be done by people and which by technology (i.e., functional allocation) is missing from these architecture diagrams.

For a physical system, the allocation of function would be done at the next level of system decomposition, in the subsystem architectures, as part of the *Design & verify subsystems & components* process step. Completing this step for a socio-technical system would require a way to develop and define functional descriptions, socio-technical systems, and their components (people and technology). For this reason, such functional descriptions are likely to include requirements for and specifications of job designs, workspaces, management strategies, and technology-based solutions such as IT systems.

The final process step, *Integrate sub-system designs into whole system & verify w.r.t. system requirements,* results in the description of the final design that is intended to deliver the required capability. For physical products, such a description is needed to enable the system to be realised. Its applicability to the design and development of socio-technical systems depends in part on the processes used to realise socio-technical systems. For many situations, these will be change processes and so very different to those used in the realisation of technical systems (Clegg & Walsh, 2004). An evaluation of these ideas for socio-technical systems design processes with respect to Clegg's (2000) process principles is given in Table 5.

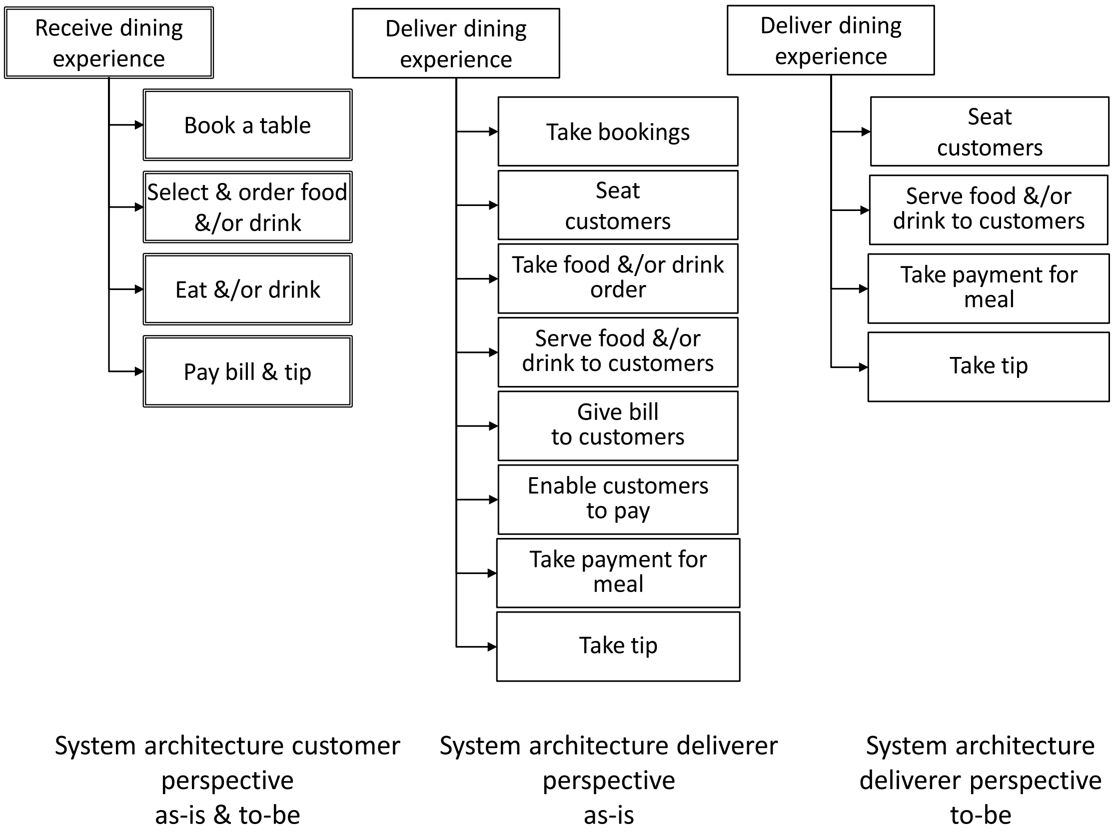

Figure 5: Possible system architectures for the as-is and to-be scenarios

Table 5: Process evaluation using Clegg's (2000) process principles

Principle 14: Design practice is itself a sociotechnical system	The implementation of the systems design process outlined in this section would include stakeholders and designers, individuals, and teams. Clegg refers to the need for structured methods, CASE tools, and new forms of working. The process introduced here would provide a framework within which such methods could be used, but there is a currently unmet need for new ways to record design decisions and outcomes that reflect social and technical aspects of the solution and the processes used to develop solutions.
Principle 15: Systems and their design should be owned by their managers and users	This could mean that, in a change process, restaurant staff should be included in the process. Design thinking and integral prototyping processes could support this and system simulations could form these prototypes. For resilience, simulations could highlight areas of risk, such as the number of suppliers of pizza ingredients in each case
Principle 16: Evaluation is an essential aspect of design	Evaluation is an integral part of the systems engineering process, and stage gates allow formal design reviews. In the context of this case study, customer feedback and financial profit could be used to evaluate design changes.
Principle 17: Design involves multidisciplinary education	Designing either the as-is or the to-be scenario requires specialist design knowledge and so multi-disciplinary working. However, since this principle relates to the design process rather than the resulting

	[designed] process, we are unable to comment further on this principle.
Principle 18: Resources and support are required for design	Resources include staff time and tools to support design processes.
Principle 19: System design involves political process	Implementing organizational change is a political process and requires key stakeholders to be engaged with and supportive of the changes. This would need to be considered in the deployment of the to-be solution.

5 Conclusions

In this chapter, we have illustrated how socio-technical principles can be considered and embedded into the design of organizational systems. We have illustrated our approach with reference to a hypothetical case study of the restaurant industry, and introduced practical methods to facilitate and support such socio-technical system design. While our case study focused on a particular context, we believe that the underlying principles and approach are applicable to a wide range of organizational and societal domains and problems.

While socio-technical principles are currently considered when developing organizational and technical systems, this is often done at a relatively late stage once the technical aspects of the system have already been designed and implemented. For instance, in architecture, post-occupancy user evaluations are common. While these existing approaches are useful, we are advocating a more proactive approach where the socio and human elements of the system are considered earlier in the design process, alongside the technical elements, as integral and inter-related components rather than as afterthoughts. With this aim in mind, we believe that the socio-technical tools and methods we have introduced in this chapter, such as the systems design vee, the scenario planning tool, and computer simulation, offer great potential for realising this vision.

The ability to anticipate possible behaviours and functions of socio-technical systems in different environments will be extremely valuable, particularly if this extends to identifying emergent properties and unintended consequences. We believe that this is particularly important given rapid advances in technology and the implications that this will have for the automation of tasks and jobs, and the knock-on effects for functional allocation and potentially unintended consequences arising from this. We believe that the socio-technical approach and methods presented in this chapter can help organizations consider these issues in a more holistic and proactive way, earlier in the design process, so that systems can be optimised for performance and resilience.

Acknowledgements

We dedicate this chapter to the memory of our dear friend and inspirational colleague Professor Chris Clegg, whose insights underpin many of the ideas in this chapter and the ongoing work of the Socio-Technical Centre that he founded at the University of Leeds in 2009.

References

Amigo, S., Smith, A., & Ludwig, T. (2008). Using task clarification, goal setting, and feedback to decrease table busing times in a franchise pizza restaurant. *Journal of Organizational Behavior Management*, *28*(3), 176–187.

Billings, C. E. (1997). *Aviation automation: The search for a human-centered approach*. Oxford, UK: CRC Press.

Bodin, P., & Wiman, B. (2004). Resilience and other stability concepts in ecology: Notes on their origin, validity, and usefulness. *ESS Bulletin, 2*(2), 33–43.

Bourgeois, L. J. (1981). On the measurement of organizational slack. *Academy of Management Review, 6*(1), 29–39.

Brass, D. J., Galaskiewicz, J., Greve, H. R., & Tsai, W. (2004). Taking stock of networks and organizations: A multilevel perspective. *Academy of Management Journal*, *47*(6), 795–817.

Busby, J. S. (2001). Error and distributed cognition in design. *Design Studies*, *22*(3), 233–254.

Casner, S. M., Geven, R. W., Recker, M. P., & Schooler, J. W. (2014). The retention of manual flying skills in the automated cockpit. *Human Factors*, *56*(8), 1506–1516.

Challenger, R., Clegg, C. W., & Shepherd, C. (2013). Function allocation in complex systems: reframing an old problem. *Ergonomics*, *56*(7), 1051–1069.

Cherns, A. (1976). The principles of sociotechnical design. *Human Relations*, *29*(8), 783–792.

Cherns, A. (1987). Principles of sociotechnical design revisited. *Human Relations*, *40*(3), 153–161.

Clegg, C. W. (2000). Sociotechnical principles for system design. *Applied Ergonomics*, *31*(5), 463–477.

Clegg, C. W. (2016). A systems approach to increasing UK productivity. McKay, A. (Ed.). Report published on White Rose Research Online, February 2016. http://eprints.whiterose.ac.uk/96376/

Clegg, C.W., & Davis, M.C. (2016). Automation/Advanced Manufacturing Technology/Computer-Based Integrated Technology. In S.G. Rogelberg (Ed), The Sage Encyclopedia of Industrial and Organizational Psychology (2nd ed). Thousand Oaks, CA: Sage.

Clegg, C. W., Robinson, M. A., Davis, M. C., Bolton, L., Pieniazek, R., & McKay, A. (2017). Applying organizational psychology as a design science: A method for predicting malfunctions in socio-technical systems (PreMiSTS). *Design Science*, *3*, 1–31.

Clegg, C., & Shepherd, C. (2007). 'The biggest computer programme in the world… ever!': Time for a change in mindset? *Journal of Information Technology*, *22*(3), 212–221.

Clegg, C. W., & Walsh, S. (2004). Change management: Time for a change! *European Journal of Work and Organizational Psychology*, *13*(2), 217–239.

Cressy, R., Munari, F., & Malipiero, A. (2007). Playing to their strengths? Evidence that specialization in the private equity industry confers competitive advantage. *Journal of Corporate Finance*, *13*(4), 647–669.

Crowder, R. M., Robinson, M. A., Hughes, H. P., & Sim, Y. W. (2012). The development of an agent-based modeling framework for simulating engineering team work. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *42*(6), 1425–1439.

Daniels, K., Le Blanc, P., & Davis, M. (2013). The models that made job design. In M. Peeters, J. de Jonge & T. Taris (Eds.), *An Introduction into Contemporary Work Psychology* (pp. 63-88). Chichester, UK: Wiley

Davis, M.C. (2019). Socio-technical systems thinking and the physical environment of OB. In, R. Ayoko & N. Ashkanasy (Eds), Organizational Behavior and the Physical Environment. Routledge

Davis, M. C., Challenger, R., Jayewardene, D. N. W., & Clegg, C. W. (2014). Advancing socio-technical systems thinking: A call for bravery. *Applied Ergonomics*, *45*(2), 171–180.

Davis, M.C., & Coan, P. (2015). Organizational Change. In J. Barling & J. Robertson (Eds.), *The Psychology of Green Organizations* (pp. 244-274). New York, NY: Oxford University Press

De Vries, T. A., Hollenbeck, J. R., Davison, R. B., Walter, F., & Van Der Vegt, G. S. (2016). Managing coordination in multiteam systems: Integrating micro and macro perspectives. *Academy of Management Journal*, *59*(5), 1823–1844.

Eason, K. (2014). Afterword: The past, present and future of sociotechnical systems theory. Applied Ergonomics, 45(2, Part A), 213-220.

Grote, G., Weyer, J., & Stanton, N. A. (2014). Beyond human-centred automation – concepts for human-machine interaction in multi-layered networks. *Ergonomics*, *57*(3), 289–294.

Hawe, G. I., Coates, G., Wilson, D. T., & Crouch, R. S. (2012). Agent-based simulation for large-scale emergency response: A survey of usage and implementation. *ACM Computing Surveys*, *45*(1), 8.

Hettinger, L. J., Kirlik, A., Goh, Y. M., & Buckle, P. (2015). Modelling and simulation of complex sociotechnical systems: Envisioning and analysing work environments. *Ergonomics*, *58*(4), 600–614.

Hollnagel, E. (2014). Resilience engineering and the built environment. *Building Research & Information, 42*(2), 221–228.

Hughes, H. (2017). *The role of advice networks in the design and development of jobs*. Doctoral thesis, University of Leeds, UK.

Hughes, H. P., Clegg, C. W., Bolton, L. E., & Machon, L. C. (2017). Systems scenarios: A tool for facilitating the socio-technical design of work systems. *Ergonomics*, *60*(10), 1319–1335.

Hughes, H. P. N., Clegg, C. W., Robinson, M. A., & Crowder, R. M. (2012). Agent-based modelling and simulation: The potential contribution to organizational psychology. *Journal of Occupational and Organizational Psychology, 85*(3), 487–502.

Kahn, W. A., Barton, M. A., Fisher, C., Heaphy, E. D., Reid, E., & Rouse, E. D. (2018). The geography of strain: Organizational resilience as a function of intergroup relations. *Academy of Management Review*, 43(3), 509–529.

Kundu, S., McKay, A., & de Pennington, A. (2008). Selection of decoupling points in supply chains using a knowledge-based approach. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *222*(11), 1529–1549.

Lechler, T. G., & Dvir, D. (2010). An alternative taxonomy of project management structures: linking project management structures and project success. *IEEE Transactions on Engineering Management*, *57*(2), 198–210.

Lengnick-Hall, C. A., Beck, T. E., & Lengnick-Hall, M. L. (2011). Developing a capacity for organizational resilience through strategic human resource management. *Human Resource Management Review, 21*(3), 243–255.

Mannion, R., & Braithwaite, J. (2012). Unintended consequences of performance measurement in healthcare: 20 salutary lessons from the English National Health Service. *Internal medicine Journal*, *42*(5), 569–574.

Marks, M. A., DeChurch, L. A., Mathieu, J. E., Panzer, F. J., & Alonso, A. (2005). Teamwork in multiteam systems. *Journal of Applied Psychology*, *90*(5), 964–971.

Mathieu, J. E., Heffner, T. S., Goodwin, G. F., Salas, E., & Cannon-Bowers, J. A. (2000). The influence of shared mental models on team process and performance. *Journal of Applied Psychology, 85*(2), 273–283.

McKay, A., Baker, R. B., Chittenden, R. J., & De Pennington, A. (2018). A framework for students to visualise the implications of design decisions in global supply networks. Accepted in *International Conference on Engineering and Product Design Education* Dyson School of Design Engineering, Imperial College, London, United Kingdom.

McKay, A., & Kundu, S. (2014). A representation scheme for digital product service system definitions. *Advanced Engineering Informatics*, *28*(4), 479–498.

Mumford, E. (2006). The story of socio-technical design: reflections on its successes, failures and potential. Information Systems Journal, 16(4), 317–342.

Pieniazek, R. (2017). *Organizational resilience: An investigation into its conceptualisation, outcomes and, boundary conditions*. Doctoral thesis, University of Leeds, UK.

RAEng. (2007). *Creating systems that work: Principles of engineering systems for the 21st century,* London, UK: The Royal Academy of Engineering. Available from: www.raeng.org.uk/publications/reports/rae-systems-report.

Robinson, M. A. (2012). How design engineers spend their time: Job content and task satisfaction. *Design studies*, *33*(4), 391–425.

Robinson, M. A. (2016). Quantitative research principles and methods for human-focused research in engineering design. In P. Cash, T. Stanković, & Štorga, M. (Eds.), *Experimental design research: Exploring complex design activity* (pp. 41–64). Cham, Switzerland: Springer.

Robinson, M. A. (2018). Using multi-item psychometric scales for research and practice in human resource management. *Human Resource Management*, *57*(3), 739–750.

Robinson, M. A., Sparrow, P. R., Clegg, C., & Birdi, K. (2005). Design engineering competencies: future requirements and predicted changes in the forthcoming decade. *Design Studies*, *26*(2), 123–153.

Salas, E., Sims, D. E., & Burke, C. S. (2005). Is there a "big five" in teamwork? *Small Group Research*, *36*(5), 555–599.

Sha, M., & Srinivasan, R. (2016). Fleet sizing in chemical supply chains using agent-based simulation. *Computers & Chemical Engineering*, *84*, 180–198.

Shah, R., & Ward, P. T. (2003). Lean manufacturing: Context, practice bundles, and performance. *Journal of Operations Management, 21*(2), 129–149.

Strain, J., & Eason, K. (2000). Exploring the implications of allocation of function for human resource management in the Royal Navy. *International Journal of Human Computer Studies*, *52*(2), 319–334.

Suh, N. P. (1990). The Principles of Design. Oxford University Press.

Sutcliffe, K. M., & Vogus, T. J. (2003). Organizing for Resilience. In K. Cameron, J. E. Dutton, & R. E. Quinn (Eds.), *Positive Organizational Scholarship* (pp. 94-110). San Francisco: Berrett-Koehler.

Trist, E. L., & Bamforth, K. W. (1951). Some social and psychological consequences of the longwall method of coal-getting: An examination of the psychological situation and defences of a work group in relation to the social structure and technological content of the work system. *Human Relations, 4*(1), 3-38.

van der Wal, C. N., Formolo, D., Robinson, M. A., Minkov, M., & Bosse, T. (2017). Simulating crowd evacuation with socio-cultural, cognitive, and emotional elements. *Transactions on Computational Collective Intelligence*, *27*, 139–177.

Waterson, P. E., Older Gray, M. T., & Clegg, C. W. (2002). A sociotechnical method for designing work systems. *Human Factors*, *44*(3), 376–391.

Weyer, J., Fink, R. D., & Adelt, F. (2015). Human-machine cooperation in smart cars. An empirical investigation of the loss-of-control thesis. *Safety Science*, *72*, 199–208.

Xu, M., Qi, X., Yu, G., Zhang, H., & Gao, C. (2003). The demand disruption management problem for a supply chain system with nonlinear demand functions. *Journal of Systems Science and Systems Engineering*, *12*(1), 82–97.