Lancaster University Management School:

Author Accepted Manuscript

Please cite this paper as:

Williams, R.A., (Accepted), Towards an agent-based model using a hybrid conceptual modelling approach: a case study of relationship conflict within large enterprise system implementations

ACCEPTED FOR PUBLICATION | August, 31st 2022

ORCID NUMBER: 0000-0001-6333-9448

DOI: 10.1080/17477778.2022.2122741

Dr Richard Alun Williams
Senior Lecturer in Management Science
Lancaster University Management School
Lancaster, LA1 4YX

ORIGINAL ARTICLE

Towards an agent-based model using a hybrid conceptual modelling approach: a case study of relationship conflict within large enterprise system implementations

R. A. Williams

Department of Management Science, Lancaster University, Lancaster, UK

ARTICLE HISTORY
Compiled September 7, 2022

CONTACT R. A. Williams. Email: r.williams4@lancaster.ac.uk

ABSTRACT
There is a long history of using modelling and simulation (M&S) to investigate complex systems. With advances in computing power and simulation frameworks, we can now model and simulate increasingly complex systems, in particular socio-technical systems, which has resulted in M&S becoming an essential technique to aid decision-making. An often-overlooked activity within the design and development of computational models for socio-technical systems, is the development of a comprehensive conceptual model defining the scope of the model with respect to actors, technical resources, environment and abstraction level. With specific reference to modelling large IT and IS implementations, this incurs the added challenges of dealing with qualitative information relating to project scope and implementation processes, along with quantitative and qualitative information regarding the social network of the project resources and emergent behaviours that result from interactions between them. Our approach involves a Multi-Paradigm Hybrid Study to develop a Cross-Disciplinary Hybrid Model of relationship conflict within an enterprise system implementation. We identify that Soft Systems Methodology, Social Network Analysis, Unified Modelling Language are complementary approaches from Operational Research, Social Sciences and Software Engineering, that provide a powerful combination of techniques to develop hybrid conceptual models for subsequent encoding into a simplified Agent-Based Model of complex socio-technical systems.

KEYWORDS
Agent-Based Modelling; Hybrid Conceptual Modelling; Relationship Conflict; Social Network Analysis; Soft Systems Methodology; Unified Modelling Language

1. Introduction

We recently argued that conflict is inherent to the project-based approach for implementing large information systems through multiple third-party providers (Williams, 2019). An example of such a system, is an *enterprise system*, which provides a suite of applications that are fully integrated and enable an organisation to efficiently run their administrative functions (e.g. Human Resources, Payroll, Procurement, etc) through preconfigured and standardised business processes. Within these project-based working environments, conflict can develop through frequent (or acute) differences in opinions/outlook and can also be characterised as incompatibilities between resources at the interpersonal level (Boulding, 1963). Three forms of conflict have been found within Information Technology (IT) and Information Systems (IS) project settings (Zornova, Ripoll, & Peiro, 2002), relating to *Task Conflict*, *Process Conflict*, and *Relationship Conflict*. The most damaging is relationship conflict, which is linked to significant differences in demographics/characteristics between team members, or simply through personality clashes.

The use of computational modelling techniques to investigate social systems has advanced significantly over the past few decades, progressing from mere quantitative data analysis to one that harnesses advanced computational techniques, allowing us to model and simulate at the level of individual social actors (Williams, 2021). The progression of techniques has moved from mathematical approaches that rely on deterministic or stochastic equations, to computational ones, such as Agent-Based Modelling and Simulation (ABMS), Discrete Event Simulation and System Dynamics. Alam & Geller (2012) declare that unlike physical systems that can be definitively measured and quantified, social systems are modelled descriptively and validated qualitatively, meaning fieldwork is required to collect empirical data and evidence.

Social systems that form within project-based environments are complex, encompassing a range of properties and characteristics that emerge from interactions of the individual project members with each other, with technologies that are being used, and with the organisational/management environment. The social system can therefore be considered as a social network of interactions/relationships that occur between team members, with two main dimensions relating to time and physical space, which is important because project teams are often located within a set of organisational locations. From previous work (Williams, 2021), we believe that ABMS is the most suitable computational modelling paradigm for modelling organisational social systems, due to its inherent ability to model the micro-to-macro mapping of the social system, with the complex system-level behaviours emerging as a result of low-level interactions between the human agents. As with all models, Agent-Based Models (ABMs) provide simplified views of real-world systems, and therefore require key design decisions to be made on how real-world data is interpreted and translated, to ensure it can be used for design and development of the ABM. In accordance with previous work (Williams, 2018), we contend that development of a comprehensive *Conceptual Model* is essential to inspire faith in ABMs of complex systems.

It is timely, that conceptual modelling has once again become a topic of interest within the Operational Research (OR) community in recent years, with special consideration being applied to the processes and tools associated with developing conceptual models for computational simulations. A number of these tools have come from fields outside of *hard* OR, including: Soft Systems Methodology (SSM) from *soft* OR (e.g. Kotiadis (2007)); Social Network Analysis (SNA) from Mathematics and Social Sciences (e.g. Williams (2019)); and Unified Modelling Language (UML) from Computer Science and Software Engineering (e.g. Vasilakis, Lecznarowicz & Lee (2009)). Whereas the use of multiple techniques for development of computational models is termed *Hybrid Simulation* (Mustafee & Powell, 2018), the use of multiple tools and approaches for other parts of the simulation lifecycle (e.g. requirements analysis, model design, validation, etc) is termed *Hybrid Modelling* (Powell & Mustafee, 2017; Tolk, Harper & Mustafee, 2021). This manuscript provides an account of our experiences in using SSM, SNA and UML to develop a hybrid conceptual model, alongside a simplified ABMS (Tako, Tsioptsias & Robinson, 2020) of relationship conflict within a large enterprise system implementation. Our approach, as defined by Mustafee & Powell (2018), involves a *Multi-Paradigm Hybrid Study*, using the single computational modelling *Technique* of ABMS, which provides a discrete time-stepped *Method* and is developed using the FLAME package as the *Tool* (Coakley, Smallwood & Holcombe, 2006). Similarly, as we utilise both *Soft* (e.g. SSM) and *Hard* (e.g. ABMS) OR approaches, our study employs a *Hybrid OR/MS Modelling* approach, which Mustafee, Harper & Onggo (2020) define as a Type D1 model. Furthermore, because our use of SNA takes inspiration from the Mathematical and Social Sciences, this is defined as a *Cross-Disciplinary Hybrid Model*, which corresponds to a Type E model (Mustafee, Harper & Onggo, 2020).

The manuscript is structured accordingly: section 2 provides an overview of the major concepts that contribute to the theory behind our study; section 3 discusses the methods used; section 4 presents our modelling; and section 5 provides our discussion and conclusions.

2. Related work

This section provides an overview of major concepts that contribute to theory behind our study.

2.1. Soft systems methodology

SSM was developed by Peter Checkland and colleagues in the 1970s (Checkland, 1981) to provide a complementary *Soft* approach for OR in order to facilitate analysis of the complex social and environmental factors in organisational settings. SSM has become a leading method for investigating socio-technical problems within IS/IT due to its integration of *Human* and *Social* factors of organisations (Cox & Kirkham, 2019). SSM is used as a problem structuring method to develop the problem definition and semi-formally define an abstracted model of the problem (Kotiadis & Robinson, 2008). Checkland himself described SSM as an organised learning system that was specifically developed to facilitate definition and deconstruction of *complex* and *messy* organisational problems (Checkland, 1999).

SSM allows analysts to investigate the problem situation through two main tools. The first, is the *Rich Picture*, providing an informal diagrammatic representation of the significant components of the problem and relationships to the wider organisational context. Rich pictures represent the major processes, technologies and human actors, along with their relationships/interactions, and importantly the opinions of these human actors (Checkland & Poulter, 2010). The second, is *Analysis One, Two and Three*, representing *Role Analysis*, *Social System Analysis*, and *Political System Analysis*. Additional tools include: *Root Definition*, which provides a formal definition of the system; identification of the *Transformation Process* (T) of system input(s) to output(s), which through observing the system from multiple perspectives, can lead to development of interventions to eradicate organisational problem(s); and the *CATWOE* mnemonic, which is used when developing the root definition and consists of: C (Customers), A (Actors), T (Transformation process), W (Weltanschauung or World-view), O (Ownership), and E (Environmental constraints).

2.2. Social network analysis

Social Network is a term used to describe a group of people and the links between them (Scott, 2013). The academic investigation of social networks takes inspiration from mathematical graph theory and is termed *Social Network Analysis* (SNA; (Scott, 2013)). Within a work-based social network, links between resources can represent any type of social behaviour/interaction that arises in the workplace (Williams, 2019). Traditionally, SNA has used quantitative approaches to perform structural analysis, providing empirical measures, such as graphs and descriptive statistics on network topology, for analysing the person-level data obtained through qualitative approaches.

When applied to enterprise system implementations, the wider programme-level social network is in fact a product of the individual project-level social networks. These project teams are routinely focused on implementing one or more of the functional modules within the enterprise system and consist of team members from across the various organisations involved in the implementation (Williams, 2019). These workplace social networks facilitate communication between resources, with the structure of the network enabling the effective and efficient transmission of knowledge and information. It is therefore widely assumed that resources are not independent, but are influenced by others within the network. With this in mind, Borgatti, Mehra, Brass & Labianca (2009) postulate that the social network enables the flow of attitudes, behaviours or emotions through either physical transfer, which in our case would be face-to-face communication, or an imitative process through individual resources acquiring the attitudes or emotional state(s) from others within their environment.

2.3. Unified modelling language

Conceptual modelling provides an abstract representation of the real-world system of interest, through documentation of our understanding, which may include statements around the background context for the computational model, along with the definition of assumptions, constraints, and indeed relationships and interactions between components of the system (e.g. people, processes, technology). Along with SSM to provide the background context and qualitative systems analysis, the Unified Modelling Language (UML) provides a complementary approach to semi-formally document the required functionality of the computational model, and has become the *de facto* standard for modelling software systems by software engineers (Chaudron, Heijstek & Nugroho, 2012). The UML specification (version 2.5.1) as developed by The Object Management Group (2017), defines 14 separate notations, categorised into three groups: *Structure Diagrams*, defining the static structure of components; *Behaviour Diagrams*, defining the dynamic behaviour(s) of components; and *Implementation Diagrams*, defining the IS/IT infrastructure.

2.4. Brief primer on agent-based modelling and simulation

ABMS has been used within the OR community for over three decades. The ability to conveniently model complex systems, and specifically to model emergent behaviour(s) following interactions of individual components (i.e. agents), was a major driver for its popularity (Taylor, 2014). ABMS is used to either increase our understanding of mechanisms involved in generating emergent behaviours of real-world complex systems, or to predict how the system's dynamics are likely to be affected by perturbations (Williams, 2018). We previously discussed (Williams, 2021) that data collected through participant interviews or observations within management/organisational settings, is applicable to the object-oriented paradigm for design of computational models. The design of an ABM is closely aligned to the object-oriented approach, but builds upon this paradigm because from a design perspective an ABM is not bound by a central control mechanism because the individual agents are autonomous and have the ability to adapt at the individual level (Macal, 2016; Macal & North, 2010).

Boudon (1998) advises that ABMs are considered to be non-black-box models, which means that the design of agents within ABMs of social systems, oftentimes relies on a mixture of qualitative data, experimental data, and empirically validated theory. This means that an ABM of social network behaviour needs to have the relevant rules for agent interactions and behaviours defined, which comes from the qualitative data (usually collected through interviews/observations), alongside the complete list of agent relationships and network topology, which is part of the experimental data (usually collected through observations and document analysis). It is therefore essential that we can explain how the social network being modelled actually forms, along with how the social processes (derived from theory) facilitate the dynamics and individual agent behaviours. In addition, a major advantage of ABMS is that it enables us to model complex, non-deterministic, and heterogeneous behaviours, which facilitates investigation of differences in attributes (e.g. behavioural or demographic) of individual agents within social systems (Williams, 2021). Importantly, the behaviours of individual agents are determined by the rules and constraints imposed on them within the model and the rules governing interactions between agents. Within the ABMS, these individual agent behaviours become aggregated, and lead to the *emergence* of system-level patterns, structures, behaviour(s) and dynamics (Epstein, 1999).

3. Materials and Methods

Our methodology for hybrid conceptual modelling is based on case study research (Yin, 2018) for data collection and a multidisciplinary approach for data analysis. Specifically, we used SSM (Checkland, 1981) for problem structuring and defining the background, SNA (Scott, 2013) for investigating the topological structure of the formal work-based relationships between project team members, UML (The Object Management Group, 2017) for defining agent interactions, and FLAME for developing the simplified ABMS.

3.1. Data collection

The hybrid conceptual modelling uses our previously collected empirical data of a case study at a large UK-based organisation that was undergoing a strategic business transformation project, which required a new Enterprise System to provide significant cost savings and efficiencies (Williams, 2019, 2020). This implementation was structured around a number of projects that aligned to modules within the enterprise system and was collectively referred to as the Resource Management Programme (RM Programme). Data collection took a hybrid form, beginning with documentary analysis to develop a thorough understanding of the case study, then four focus groups with key resources to understand the causes of conflict development and subsequent impact on individual projects and wider programme-level implementation. Research ethics approval for the data collection was provided by the Faculty of Arts and Social Sciences and Lancaster University Management School Research Ethics Committee at Lancaster University.

3.2. Design and development of the ABM

Our approach for design and development of the ABM followed the Complex Systems Modelling and Simulation (CoSMoS) framework (Stepney & Polack, 2018), which provides a principled approach to modelling and simulation of complex systems, with a view to using simulation-based experimentation as a scientific instrument. This framework has three phases: Discovery Phase, identifies and structures the problem in order to establish the scientific basis of the project, and models the real-world domain within a conceptual model; Development Phase, where the computational model is developed, which in our case is a simplified ABM of relationship conflict within the RM Programme; and Exploration Phase, where simulation-based experimentation is performed.

With specific reference to the conceptual model, we followed the approach of Robinson (2008b), and utilised three complementary strands that enabled the conceptual model to act as the functional specification for our ABM. First, SSM was used to structure the problem of how relationship conflict had developed and propagated throughout the RM Programme social network. Second, SNA was used to establish the project and programme-level network topologies of the formal work-based relationships between resources. Finally, UML notations were used to semi-formally define the interactions between resources, sequence of events, and activities, that led to relationship conflict. Validation of the conceptual model was performed using: audits by the author to ensure modelling adheres to established practices; desk checking by the author to ensure the SSM, SNA and UML models are correct, complete, consistent and unambiguous; and face validation of prototype diagrams (developed on white-board) by workshop participants to compare against their detailed understanding and judgment of conflict within the RM Programme (Balci, 1994).

For development of the simplified ABM and running of simulations, we used the Flexible Large-scale Agent-based Modelling Environment (FLAME) (Coakley, Smallwood & Holcombe, 2006). Here the agents are modelled using XML templates to define the attributes and internal states of the agents, with the rule-based functions of agent behaviours being defined using C code. Simulation was performed using 50 replicate runs. Validation of the ABM was performed using a number of techniques from Balci (1994) at validation *Levels 0 and 1* from Axtell & Epstein (1994). *Level 0* validation treats the ABM as a *caricature* of reality as established through simple graphical devices, which allowed us to validate the individual agent code against the conceptual model and also their 2D location on the social network map using code walkthroughs, consistency checking, structural analysis and syntax analysis. *Level 1* validation requires the ABM to be in *qualitative* agreement with empirical macro-structures, i.e. the initial development of conflict and its propagation along the formal relationships as defined in the social network map, which used consistency checking, control-flow analysis, and visualisation.

4. Results

The case study relates to a UK-based organisation (the Customer) that was implementing a large strategic modernisation programme, underpinned by the replacement of a significant portion of their IT and IS infrastructure. From an organisation-wide business process perspective, the largest organisational costs related to Resource Management. As such, the Customer embarked upon a major IT/IS programme of work, requiring integration of new and legacy systems to yield meaningful cost savings and more efficient business processes, which would ultimately lead to a reduction in headcount across the organisation, but particularly the HR Department (Williams, 2019, 2020).

Existing infrastructure consisted of multiple standalone software packages from different vendors that were not completely integrated. The most important of these, with respect to Resource Management, related to Finance, Human Resources (HR), and Payroll. These standalone Enterprise Resource Planning (ERP) systems, had historically been developed as discrete solutions. The customer's aim for the RM Programme, was to install and configure a new enterprise system that would provide a set of standardised business processes that were fully integrated across the Financials, HR, Procurement and Payroll functions. In addition, this new enterprise system would be fully integrated with the remaining legacy systems that were not being replaced.

4.1. The SSM model

Initial analysis utilised the Analysis One, Two, Three technique and focused on the interview transcripts and workshop outputs, along with design and project management documents. Subsequent SSM analysis developed the CATWOE definition, root definition and rich picture.

Analysis One concentrated on the roles within the RM Programme. Workshop outputs and documentary analysis indicated the RM Programme had 159 team members who were organised into a Project Management Office (PMO) and Programme Directorate to focus on governance and administration, along with six project teams, three of which mapped onto ERP (i.e. HR, Financials and Payroll) and the other three to Technical development, Training, and developing IT infrastructure (see Figure 1). The Customer outsourced delivery to a number of third-party organisations. The most important, was a global software vendor that specialised in enterprise systems, to act as the sole Vendor to install, configure and test ERP modules and interfacing to legacy systems. The Customer also employed three Professional Service Providers (PSP), two of which acted as Subject Matter Experts (SME) for configuration and extension of business processes, whilst the third focused on IT architecture for hosting the enterprise system. The composition of RM Programme membership was: 65 customer, who were SMEs to facilitate design, configuration and testing of the enterprise system, along with PMO; 70 Vendor, who designed and configured the enterprise system, designed and developed bespoke extensions, along with data conversion and integration routines; 5 PSP1, who acted as independent SMEs in HR and Payroll; 7 PSP2, who acted as independent SMEs in Financials; and 11 PSP3, who developed the technical infrastructure to host the enterprise system. Finally, the overall problem owner was the Operations Director from the Customer who acted as Project Sponsor.

Analysis Two concentrated on the social system and provided insight into the background context and business drivers. As discussed above, the Customer initiated the RM Programme in order to harness the efficiencies that the enterprise system would give regarding business processes and the number of system users required to fulfil administrative functions. In fact, the underlying business case revolved around medium-term cost savings (i.e within 5 years) through automation of a large number of administrative tasks, yielding a reduction in organisational headcount.

Analysis Three concentrated on the political system, particularly power dynamics between the various organisations. The majority of resources were members of their project team for the duration of its implementation. There were however some changes in personnel on the Vendor side, which were due to a number of reasons, including: individual consultants needing to be closer to home in order to minimise overnight stays; resource constraints, such as a small number of consultants with *Expert* status in particular skills, who needed to be redeployed to other customers; or professional relationships, for instance the breakdown of cordial working relationships with Customer resources, which generated relationship conflict and required their removal and replacement. Our CATWOE definition is provided in Table 1.

Our root definition for the RM Programme: 'A large UK-based organisation is engaged in a major strategic business transformation programme, specifically focused on the installation and configuration of a new enterprise system that is fully integrated with existing custom-developed legacy systems, in order to streamline administrative functions through the use of standardised business processes within the HR, Financials and Payroll departments. It is envisaged that these large-scale business process efficiencies will significantly reduce operating costs through automation of current manually intensive processes, which would free up resources to focus on more value-oriented tasks, and ultimately allow the organisation to restructure around the new operating procedures, leading to significant savings in payroll through a reduction in headcount'. See Figure 2 for the corresponding rich picture of the RM Programme. In addition, Checkland (1981) advises the recasting of root definitions into the form: do P by Q to facilitate the achievement of R. Our PQR definition is provided in Table 2.

4.2. *Designing the ABM*

4.2.1. The RM Programme social network

As shown in Figure 1, the RM Programme was structured into six projects along with the Programme Directorate and PMO. Individual resources were assigned to one of these groups, which allowed the use of SNA to analyse the person-level and work-based relationship data. The 159 resources, along with their relationships (i.e. with other resources who they interact with to complete their assigned tasks), were defined within an adjacency matrix, to facilitate analysis of the network topology. The social network was found to contain 972 undirected relationships, providing an average degree of connectedness (i.e. average number of connections that individual members had to other members) of 12.23.

The social network map is defined in Figure 3. As expected, the network topology is structured around the Project Teams, alongside a combined grouping of the Programme Directorate and PMO and a grouping for the Vendor's offshore functional and technical resources. The pivotal resource within each grouping is the PM for each organisation, meaning that each project team has a Customer PM, a Vendor PM and potentially a third PM from a PSP. Building upon this, and of particular importance is that a number of *knots* (terminology used as per (Scott, 2013)) are identified, which represent sub-groups within the projects that correspond to resources from a particular organisation. We conjecture that these resources have strong ties to each other at the beginning of their project, due to them having similar educational and professional backgrounds, along with working towards the same organisationally imposed objectives. In addition, the PMs within the individual projects and Programme Managers from the different organisations, act as *bridgers* to *officially* connect the different Project Teams together.

The overall high-level perspective of the RM Programme has been diagrammatically defined using the UML Class Association notation (see Figure 4). This represents associations between the highest-level definitions of components within the RM Programme, such as Project Teams, Project Implementation Processes and the IS/IT. Although the enterprise system contains the individual ERP modules, it is situated in a much larger technical environment, requiring servers, a database, security and communication infrastructures, middleware and PCs for users to access the application. In addition, the ERP modules to be installed and configured in the RM Programme are implemented through project teams with resources from Customer, Vendor and PSP, and report to the PMO and Programme Directorate, which ensures compliance to Customer PMO Processes.

4.2.2. Relationship conflict within the RM Programme

The RM Programme demonstrated substantial relationship conflict throughout its 3.5 year duration, with analysis indicating these primarily revolved around: misalignment of organisation objectives between Customer, Vendor and PSPs; different organisational cultures between Customer, Vendor and PSPs; personality clashes between resources; interpersonal incompatibilities through personal, educational or professional characteristics; power imbalances between resources that work together, which develops into relationship conflict after disputes become personal and no longer about implementation of specific tasks.

In addition, task or process conflict, which are other types of conflict seen in large IS/IT implementations are able to morph into relationship conflict if not adequately managed. Within the RM Programme, we saw this happen when bad behaviours emerged from the Vendor because they were being blamed for technical issues and delays in delivery of custom extensions and system-wide configuration. However, the cause of these issues, was the Customers unwillingness to take accountability for programme-wide decisions. This caused mistrust between the majority of Vendor and Customer resources, which propagated to also negatively impact relationships between Vendor and PSP resources (who had a contractual relationship with the Customer).

Of critical importance, is the fact that the RM Programme was underpinned by the need to make significant cost savings and efficiencies by streamlining back-office functions, ultimately leading to a reduction in headcount. This led to misalignment between the Customer organisations' corporate drivers and the personal drivers of its resources, who part-way through the implementation became cognisant of the consequences of success. The result was that a number of Customer resources developed negative attitudes towards the RM Programme, became demotivated, decoupled themselves from their project's objectives, and in a few cases became actively hostile to Vendor resources, employing subversive behaviours to try and initiate project failure (see Figure 5).

4.2.3. High-level design of the ABM

The ABM was designed using the concept of *Communicating Stream X-Machines*. An *X-Machine* is a formalised specification developed by Eilenberg (1974) that has the capability to model both the *system's* data and the specification method (*function*) for controlling the system. From a conceptual modelling perspective, X-Machines employ both a diagrammatic approach and formal notation to model the system, where the X-Machine contains infinite internal memory, a set of functions, and a current state. The current state of control (the function defined in the specification method) and the current state of the memory, is processed alongside an input symbol from the input stream to determine the next state of the X-Machine, update it's memory state, and calculate the output symbol, which becomes part of the output stream used for communicating to other X-Machines. As such, the system's new *state* is a product of it's current state (using memory and the list of input symbols) and the relevant function (Kehris, Eleftherakis & Kefalas, 2000). A communicating stream X-Machine model is a formalised specification that builds upon that of X-Machines to allow computation of functional behaviours at the individual agent level, with these dynamics aggregated to generate the emergent behaviour at the system level. The formal notation of the communicating stream X-Machine specification utilises a 10-tuple notation, with C_i^X representing the ith communicating stream X-Machine component, which is defined by Stamatopoulou, Kefalas & Gheorghe (2007) as:

$$\mathrm{C}_i^X = (\Sigma_i,\Gamma_i,\mathrm{Q}_i,\mathrm{M}_i,\Phi_i,\mathrm{F}_i,\mathrm{q}_{0i},\mathrm{m}_{0i},\mathrm{I}\Phi_i,\mathrm{O}\Phi_i)$$

where:

- Σ and Γ are the input and output alphabets.
- Q is the finite set of real-world system states.
- M is the set that relates to memory.
- Φ, the *type* of the X-Machine X, is a set of partial functions φ that translates a specific input and a specific memory state to a specific output and a possibly different memory state, $\varphi : \Sigma \text{ x } M \rightarrow \Gamma \text{ x } M$.
- F is the next state partial function, $F : Q \text{ x } \Phi \rightarrow Q$, which given a state and a function from the type Φ determines the next state.
- q_0 and m_0 are the initial state and initial memory.
- $I\Phi_i$ is the communication interface for the input messages.
- $O\Phi_i$ is the communication interface for the output messages.

The simplified ABM was designed as a two-dimensional spatial model, with the 159 resources being represented by individual agents that are assigned to specific locations. Figure 6 provides a screenshot of the FLAME model, which uses the grey-scale colour scheme and workplace relationships that were defined in Figure 3. All agents have the following attributes: Person ID; Workstream (i.e. Programme, HR, Financials, etc); Role Title; Organisation (i.e. Customer, Vendor, PSP); Office Location; Role Type (i.e. Management, Functional, Technical, Training, Administrative); Relationship Conflict Quotient; Stress Quotient; Conflict Recovery Quotient; Formal Authority Quotient. The PM agents also have an attribute relating to the probability of promoting conflict in subordinates, termed *PM Chance*. Similarly, non-PM agents have an attribute relating to the probability of promoting conflict in fellow team members, termed *Employee Chance*.

4.3. The ABM of relationship conflict

The main source of relationship conflict was focused within the HR Project Team due to poor professional working relationships between Customer HR PM (CustHRPM) and Vendor HR PM (VenHRPM). CustHRPM became aware part-way through the project that although successful implementation would provide significant organisation-wide efficiencies and cost savings, the HR Department would lose a significant number of employees, which had a very personal impact on CustHRPM because their employees were longstanding and sometimes friends.

We therefore developed our simplified ABM to use CustHRPM as the origin of relationship conflict. Individual agents within the HR Project Team become *conflict infected* through stochastic interactions between them and CustHRPM, which are modulated through *Relationship Conflict Quotient* parameter value. These initial *conflict infected* agents are usually the PMs from either the Vendor or PSP2, but were occasionally others, notably Consultants from the Vendor or employees that CustHRPM line-managed. As expected with CustHRPM being the origin of conflict, our simulations indicated that the HR Project Team is where initial conflict propagation occurs because once CustHRPM generates conflict, it can stochastically propagate throughout the team. Furthermore, once the majority of agents within the HR Project Team have become *conflict infected*, agents within other Project Teams become *conflict infected*, in particular the Payroll, Financials and PMO teams, who have close links to the HR Project Team due to the integrated nature of an enterprise system and the administrative oversight of the PMO. This is because once a threshold of conflict infected agents has been reached, those that act as *bridgers* can propagate the *infection* to other Project Teams.

Our initial experimentation focused on the likelihood of conflict propagating from one Project Team to another, through changing the threshold values, which relates to the percentage of team members being *conflict infected* before the conflict can propagate into another Project Team through the *bridgers*. Simulation results provide no statistically significant difference, suggesting that once relationship conflict has developed within a Project Team, it will inevitably propagate throughout the programme-wide social network unless an intervention is implemented to dampen spread of conflict and enable resources to *recover* from their relationship conflict.

Further experimentation focused on the differing probability of conflict propagation between PMs and normal team members, through varying the *PM Chance* and *Employee Chance* parameter values. Figure 7 shows the effect of varying the probability of PMs propagating conflict on the time required for conflict to *infect* the programme-wide social network. Similarly, Figure 8 shows the effect of varying the probability of team members propagating conflict on the time required for conflict to *infect* the programme-wide social network, which has a quicker rate of *infection* of the whole social network, and also markedly reduced variability across replicate simulation runs. Importantly, the time required for complete conflict propagation can be seen to stabilise once the team member probability of propagation reaches a certain threshold (*Employee Chance* parameter value of 0.3 or greater). We conjecture this to mean that once conflict has occurred, it must be contained to a small subset of team members, or measures need to be applied to minimise propagation to other team members who were not initially involved in conflict development.

5. Discussion

The OR community has a history of research around conceptual modelling in order to ensure computational models are *fit-for-purpose*, however, we have identified a significant gap around conceptual modelling for ABMS. This paper, building on the work by Kotiadis & Robinson (2008) and Robinson (2008a,b), has addressed the gap, by using a hybrid conceptual modelling approach to develop a conceptual model and ABM of relationship conflict within a multi-partner enterprise system implementation. We hope that our work advances OR research by assisting OR researchers in designing and developing ABMs of complex organisational settings.

Our hybrid conceptual modelling approach has used SSM, SNA, and UML, which was subsequently developed into a simplified ABMS in order to analyse the development and propagation of relationship conflict within a large multi-partner enterprise system implementation. We found SSM to be a powerful tool for capturing and analysing the background context to the problem situation. In particular, we found the *rich picture* to be a powerful tool for defining the organisational environment and depicting the nature of interactions between resources that might lead to relationship conflict. Similarly, we found *Analysis One, Two and Three*, CATWOE and PQR to be very expressive and able to comprehensively define the problem situation. In addition, we found SNA to be a powerful tool for capturing and analysing the social network topology of the RM Programme, which can be considered a typical example of a large multi-partner enterprise system implementation. It is noteworthy how the network map provides a reciprocal interpretation to the *rich picture* and Analysis One (Actors) and Two (Social System) from SSM. Furthermore, we found UML to be a very expressive language that provides a number of diagrammatic notations for developing complementary views of complex socio-technical systems.

Our simulation-based experimentation has begun to investigate the development and propagation of conflict throughout the social network of large multi-vendor software implementations, such as the RM Programme. Our initial investigation has focused on the role that CustHRPM had in the development of intra-group conflict (between project team members) within the HR Project, and how other PMs in the Project Team, once conflict infected, could act as *bridgers* to develop inter-group conflict (between different project teams). In particular, we found that relationship conflict, once developed, was able to propagate throughout the entire programme-wide social network if no project management interventions are performed. Experimentation around propagation probability for PMs and team members suggests a greater programme-wide effect from conflict propagating between team members than from PMs. Furthermore, our results indicate a *tipping point* where relationship conflict is guaranteed to propagate throughout the entire network if the probability of conflict spreading between team members is not controlled, with Figure 8 indicating this probability should be kept below 30%.

Although PMs have access to more information (as well as acting as *bridgers* in the social network), communication and interactions between team members has a more significant impact on conflict transmission rate. As a result, when conflict spreads, team members who don't have formal work-based relationships can be *infected* through a small chain of only one or two intermediary members. However, we believe that the PMs role is paramount after conflict has developed within their Project Team, due to both their professional role, but also their network characteristic of being a *bridger*. As such, PMs need to develop interventions that restrict the likelihood of propagation from their Project Team to another, which we conjecture could potentially be through implementing a kind of *quarantine* intervention by temporarily severing the link to other Project Teams, or through arbitration between *conflict infected* team members.

To conclude, we believe our hybrid conceptual model and resultant simplified ABM confirms that simple ABMs used for simulation-based experimentation can act as powerful tools to investigate how relationship conflict can develop within large multi-partner enterprise system implementations. Our simulations have also pointed us to a new exciting avenue of future research that could seek inspiration from the immune system, because these early findings suggest to us that relationship conflict might propagate throughout the programme-wide social network in a similar way to infections that propagate through social networks. As such, we believe that inspiration can be taken from the way the human immune system fights infection, and that analogous approaches can be developed by PMs in order to intervene when relationship conflict has *infected* their Project Team.

Acknowledgements

We would llike to convey our thanks to Mr Yuyang Tang for his stimulating discussions when an MSc Business Analytics student under our supervision. The actual content of Mr Yang's MSc Dissertation is soon to be published within a conference paper, but his work gave us renewed enthusiasm to continue our computational modelling and simulation activities into the three main types of conflict within large enterprise system implementations. We would also like to convey our gratitude to the two anonymous reviewers who provided thorough and thoughtful comments on the initial manuscript to ensure that it would be of interest to the OR/MS Community.

This work was in part funded by a Management and Business Development Fellowship awarded jointly by the Society for the Advancement of Management Studies, Economic and Social Research Council, and the United Kingdom Commission for Employment and Skills (SAMS-ESRC-UKCES) with Grant No. ES/L002612/1. The funders had no involvement in study design; development of the hybrid model; in the writing of this manuscript, or the decision to submit this manuscript for publication.

References

Alam, S. J., & Geller, A. (2012). *Agent-Based Models of Geographical Systems*, Heidelberg, Germany: Springer.

Axtell, R. L., & Epstein, J. L. (1994). Agent-based modelling: understanding our creations. *Bulletin of the Santa Fe Institute*, *Winter*, 28-32.

Balci, O. (1994). Validation, verification and testing techniques throughout the life cycle of a simulation study. *Annals of Operations Research*, *53*, 121-173.

Borgatti, S. P., Mehra, A., Brass, D. J., & Labianca, G. (2009). Network analysis in the social sciences, *Science*, *323*, 892-895.

Boudon, R. (1998). (1998). *Social Mechanisms: An Analytical Approach to Social Theory.* Cambridge, UK: Cambridge University Press.

Boulding, K. (1963). *Conflict and Defense*. New York, NY: Harper & Row.

Chaudron, M. R. V., Heijstek, W., & Nugroho, A. (2012). How effective is UML modelling? An empirical perspective on costs and benefits. *Software and Systems Modelling*, *12*, 571-580.

Checkland, P. (1981). *Soft systems methodology in action.* (1st ed.). Chichester, UK: John Wiley & Sons Ltd.

Checkland, P. (1999). *Systems Thinking, Systems Practice.* (2nd ed.). Chichester, UK: John Wiley & Sons Ltd.

Checkland, P., & Poulter, J. (2010). *Systems Approaches to Managing Change: A practical guide*, pp. 191-242, London, UK: Springer.

Coakley, S., Smallwood, R., & Holcombe, M. (2006). From molecules to insect communities - how formal agent-based computational modelling is uncovering new biological facts. *Scientiae Mathematicae Japonicae Online*, *e-2006*, 765-778.

Cox, S. A., & Kirkham, S. (2019). Understanding the contribution and challenges of using soft systems methodology to facilitate cultural change: A case study in the public sector. *Lecture Notes in Information Systems and Organizations (LNISO*, *28*, Cham, Switzerland: Springer.

Crossley, N. (2010). The social world of the network: Combining qualitative and quantitative elements in social network analysis. *Sociologica*, *4*, 1-34

Eilenberg, S. (1974). *Automata Languages and Machines.* Volume A. New York, NY: Academic Press

Epstein, J. (1999). Agent-based computational models and generative social science. *Complexity*, *4*, 41-60.

Holcombe, M. (1988). X-machines as a basis for dynamic system specification. *Software Engineering Journal*, 69-76.

Kehris, E., Eleftherakis, G., & Kefalas, P. (2000). Using X-machines to model and test discrete event simulation programs. *System and Control: Theory and Applications*, 163-168.

Kotiadis, K. (2007). Using soft systems methodology to determine the simulation study objectives. *Journal of Simulation*, *1*, 215-222.

Kotiadis, K., & Robinson, S. (2008). Conceptual modelling: Knowledge acquisition and model abstraction. In: S. J. Mason, R. R. Hill, L. Monch, O. Rose, T. Jefferson, & J. W. Fowler (Eds.), *Proceedings of the 2008 Winter Simulation Conference*, pp.951-958, Miami, FL: IEEE.

Macal, C. M. (2016). Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*, *10*, 144-156.

Macal, C. M., & North, M. J. (2010). Tuturial on agent-based modelling and simulation. *Journal of Simulation*, *4*, 151-162.

Mustafee, N., Harper, A., & Onggo, B. S. (2020). Hybrid modelling and simulation (M&S): driving innovation in the theory and practice of M&S. In. K. H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, & R. Thiesing (Eds.), *Proceedings of the 2020 Winter Simulation Conference*, pp.3140-3151, Virtual Conference: IEEE

Mustafee, N., & Powell, J. H. (2018). From hybrid simulation to hybrid systems modelling. In. M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, & B. Johansson (Eds.), *Proceedings of the 2018 Winter Simulation Conference*, pp.1430-1439, Gothenberg, Sweden: IEEE.

Powell, J. H., & Mustafee, N. (2017). Widening requirements capture with soft methods: an investigation of hybrid M&S studies in health care. *Journal of the Operational Research Society*, *68*, 1211-1222.

Robinson, S. (2008a). Conceptual modelling for simulation part i: Definition and requirements. *Journal of the Operational Research Society*, *59*, 278-290.

Robinson, S. (2008b). Conceptual modelling for simulation part ii: a framework for conceptual modelling. *Journal of the Operational Research Society*, *59*, 291-304.

Scott, J. (2013). *Social network analysis.* (3rd ed.). London, UK: Sage.

Stamatopoulou, I., Kefalas, P., & Gheorghe, M. (2007). Modelling the dynamics of biological state-based systems. *BioSystems*, *87*, 142-149.

Stepney, S., & Polack, F. A. C. (2018). *Engineering simulations as scientific instruments: a pattern language.* London, UK: Springer

Tako, A. A., Tsioptsias, N., & Robinson, S. (2020). Can we learn from simplified simulation models? An experimental study on user learning. *Journal of Simulation*, *14*, 130-144.

Taylor, S. J. E. (2014). *Agent-Based Modeling and Simulation*, The OR Essential Series. Basingstoke, UK : Palgrave Macmillan.

The Object Management Group. (2017). *OMG unified modeling language version 2.5.1.* Retrieved from https://www.omg.org/spec/UML/2.5.1/PDF

Tolk, A., Harper, A., & Mustafee, N. (2021). Hybrid models as transdisciplinary research enablers. *European Journal of Operational Research*, *291*, 1075-1090.

Vasilakis, C., Lecznarowicz, D., & Lee, C. (2009). Developing model requirements for patient flow simulation studies using the Unified Modelling Language (UML). *Journal of Simulation*, *3*, 141-149.

White, L. (2016). Behavioural operational research: Towards a framework for understanding behaviour in OR interventions. *European Journal of Operational Research*. *249*, 827-841.

Williams, R. A. (2018). Lessons learned on development and application of agent-based models of complex dynamical systems. *Simulation Modelling Practice and Theory*, *83*, 201-212.

Williams, R. A. (2019). Conflict propagation within large technology and software engineering programmes: a multi-partner enterprise system implementattion as case study. *IEEE Access*, *7*, 167696-167713.

Williams, R. A. (2020). Cybernetics of conflict within multi-partner technology and software engineering programmes. *IEEE Access*, *8*, 94994-95018.

Williams, R. A. (2021). User experiences using FLAME: a case study modelling conflict in large enterprise system implementations. *Simulation Modelling Practice and Theory*, *106*, 102196.

Yin, R. K. (2018). *Case study research and applications: design and methods.* (6th ed.). Los Angeles, CA: Sage.

Zornova, A., Ripoll, P., & Peiro, J. M. (2002). Conflict management in groups that work in two different communication contexts. *Small Group Research*, *33*, 481-508.

Tables

Component	**Definition**	**Mapping to RM Programme**
Customers	The beneficiaries or users that the system is imposed upon.	The Customer resources who will use the enterprise system following the RM Programme, so this does not relate solely to the RM Programme Customer team members, but the employee-base of 130,000.
Actors	Those who will do the Transformation Process.	The Customer, Software Vendor and 3 PSPs.
Transformation Process	The conversion of the input to output.	The design and implementation of the enterprise system and associated IT.
Weltanschauung	The worldview, which makes the Transformation Process meaningful in context.	Prior to the RM Programme, the Customer performed a thorough analysis of business benefits that would be achieved through an IT/IS enabled business transformation programme, with focus on operational savings and increased performance of administrative functions.
Ownership	Those who manage implementation of the system.	Senior Customer personnel within the HR, Financials and Payroll departments.

Table 1. CATWOE definition for the RM Programme.

Component	Definition	Mapping to RM Programme
P	What to do	Implement a new IT and IS architecture, which is centred around HR, Financials and Payroll ERP modules, and will be fully integrated to any remaining legacy systems.
Q	How to do it	Outsource the design, development and implementation to third-party providers, comprised of a single Software Vendor and Three PSPs.
R	Why do it	There is an urgent need to standardi~~z~~se back office business processes in order to facilitate significant efficiencies in administrative functions, leading to major operating cost reductions.

Table 2. PQR definition for the RM Programme.

Figures

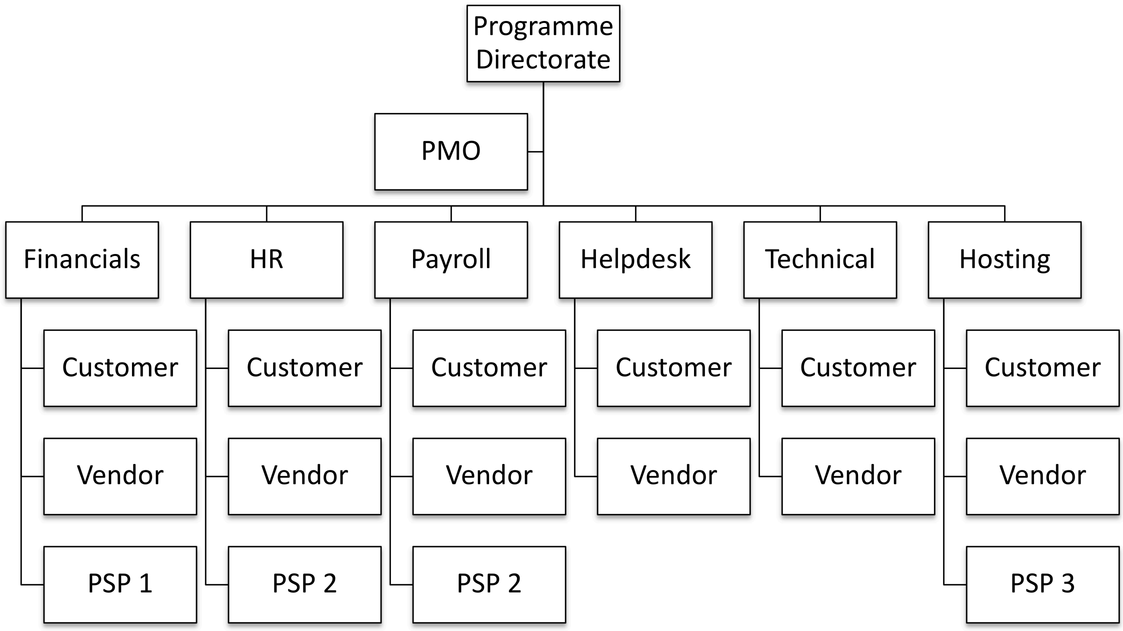

Figure 1.

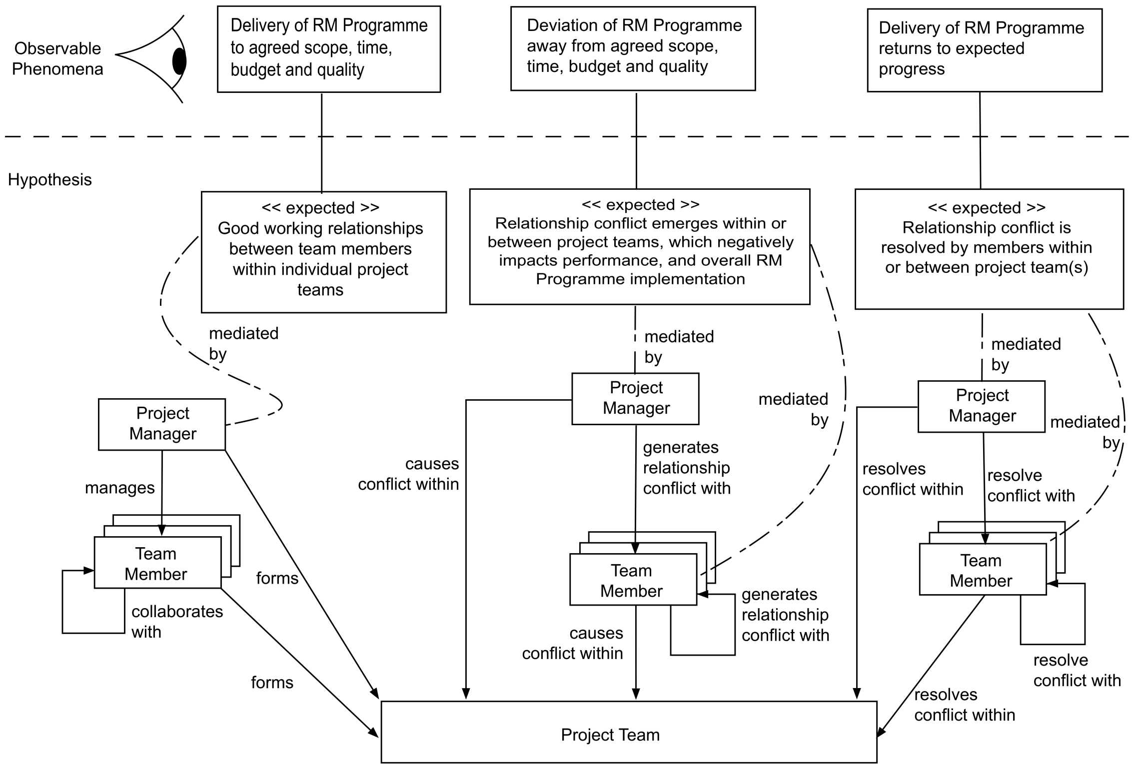

Figure 2.

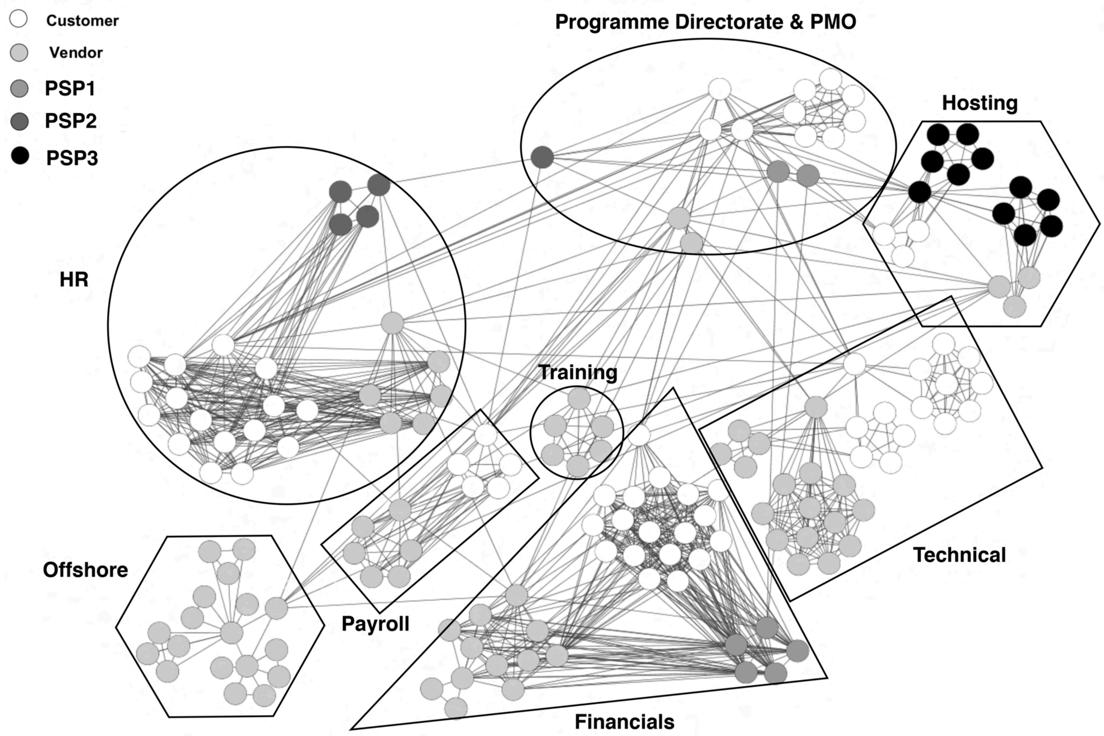

Figure 3.

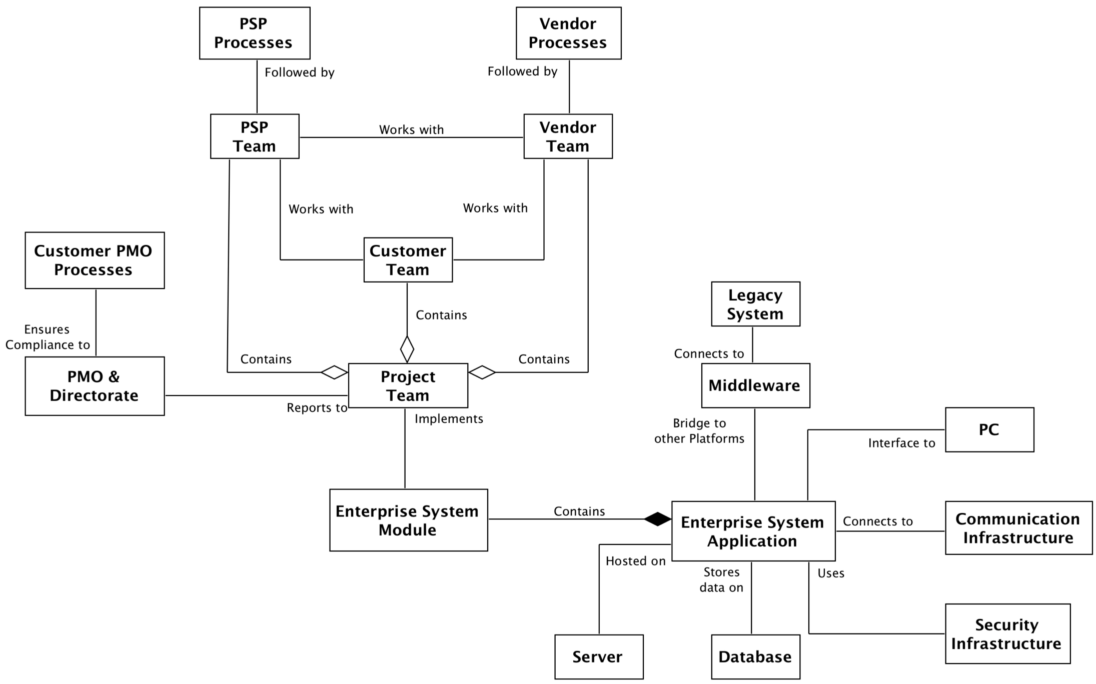

Figure 4.

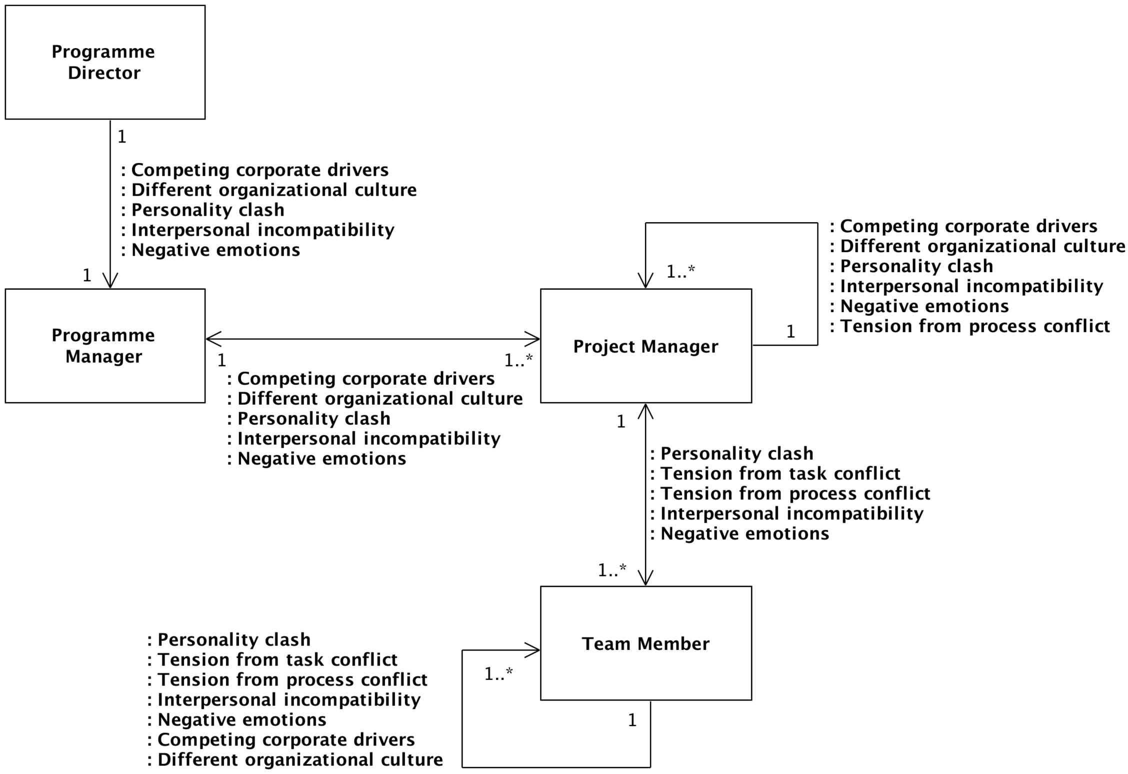

Figure 5.

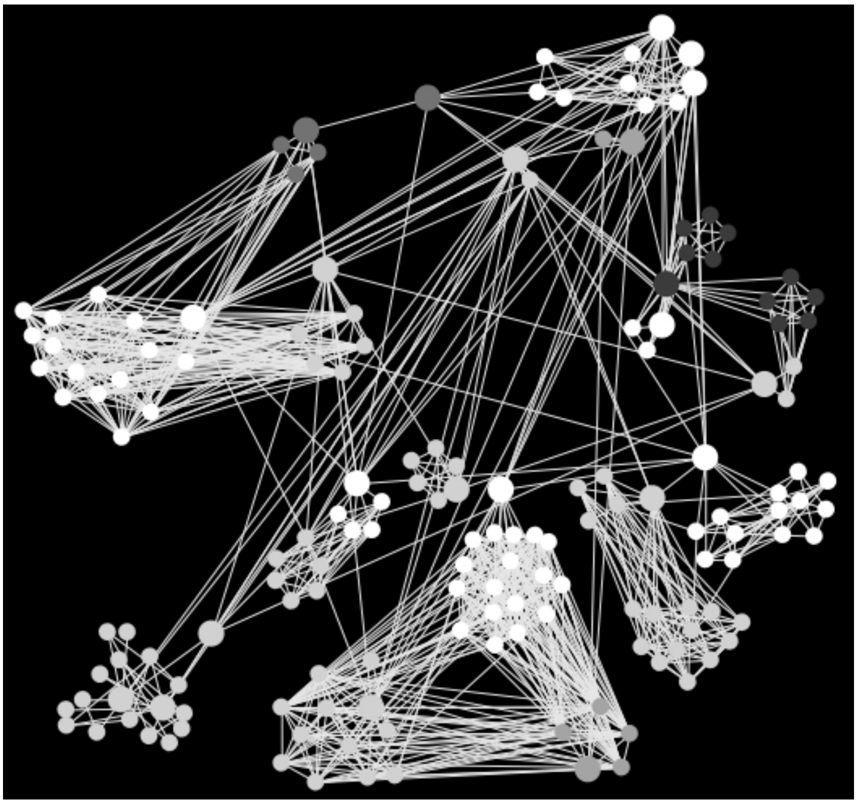

Figure 6.

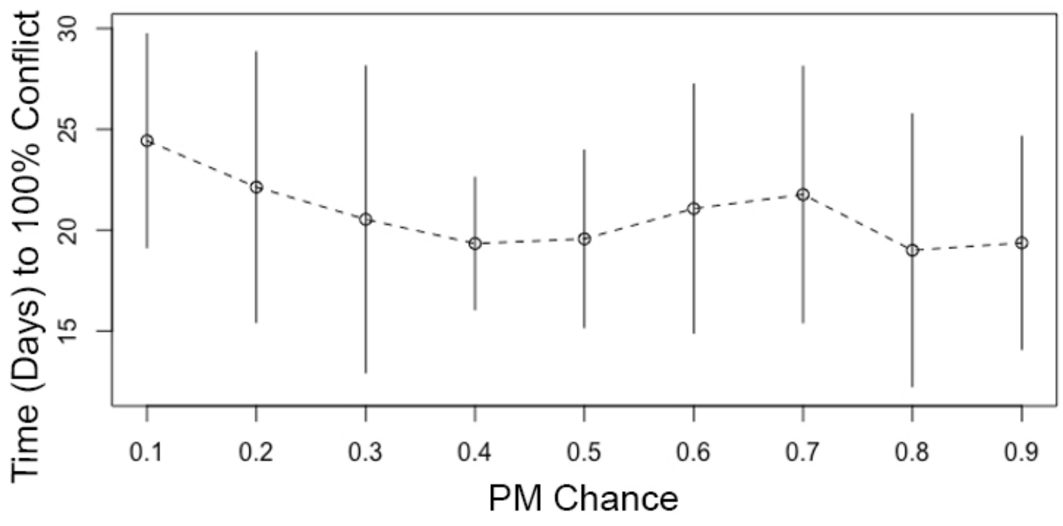

Figure 7.

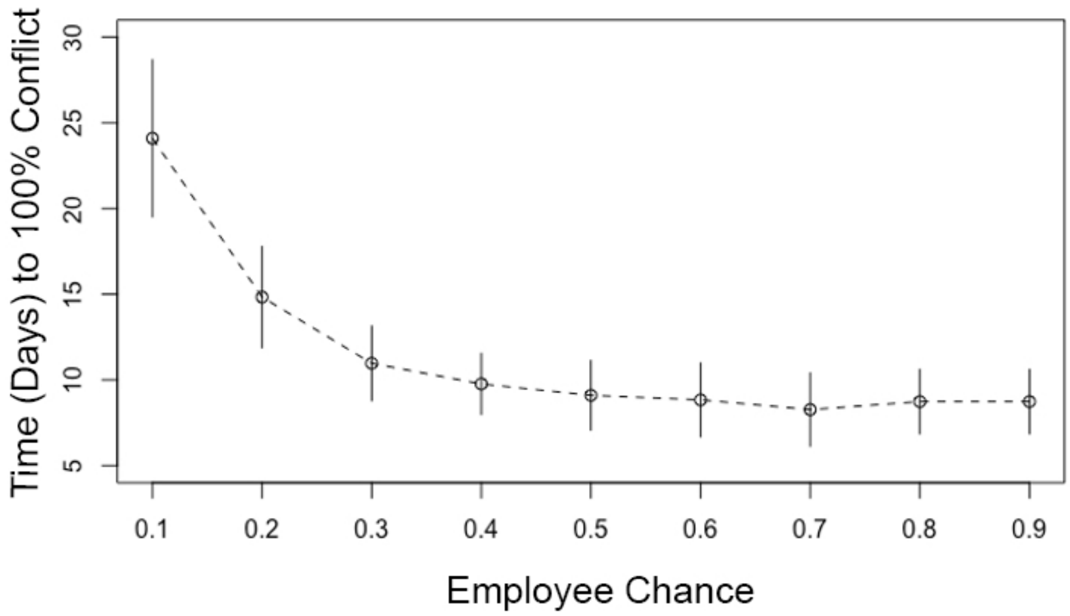

Figure 8.

Figure 1 Structure of the RM Programme with respect to project teams and their composition.

Figure 2 Rich Picture. Relationship conflict developed within the RM Programme through a number of mechanisms. We hypothesi~~z~~se that once formed, relationship conflict can propagate throughout the social network of the wider programme, which could impact delivery of the RM Programme against the agreed scope, time, budget and quality metrics.

Figure 3 Social Network Map for the RM Programme. This social network defines the *official* workplace relationships between the 159 individual resources. It is composed of sub-networks at the project team level (after Williams (2019)).

Figure 4 UML Class Association diagram for the RM Programme. This represents the associations between the highest-level definitions of components (i.e. agent Classes) within the RM Programme, such as Project Teams, Project Implementation Processes and the various IS and IT.

Figure 5 UML Class diagram for Relationship Conflict in the RM Programme.

Figure 6 Screenshot of FLAME model, showing the 159 agents and the workplace relationships between them.

Figure 7 Impact of PM probability of conflict propagation on the time (in days) required for complete programme-wide conflict.

Figure 8 Impact of team member probability of conflict propagation on the time (in days) required for complete programme-wide conflict.