A Model Based Realisation of Actor Model to Conceptualise an Aid for Complex Dynamic Decision-Making

Authors (Information ommitted for double-blind review)
emails

Keywords: Organisational decision making, Simulation, Actor Model of Computation, Actor Based Simulation.

Abstract: Effective decision-making of modern organisation requires deep understanding of various aspects of organisation such as its goals, structure, business-as-usual operational processes etc. The large size and complex structure of organisations, socio-technical characteristics, and fast business dynamics make this decision-making a challenging endeavour. The state-of-practice of decision-making that relies heavily on human experts is often reported as ineffective, imprecise and lacking in agility. This paper evaluates a set of candidate technologies and makes a case for using actor based simulation techniques as an aid for complex dynamic decision-making. The approach is justified by enumeration of basic requirements of complex dynamic decision-making and the conducting a suitability of analysis of state-of-the-art enterprise modelling techniques. The research contributes a conceptual meta-model that represents necessary aspects of organisation for complex dynamic decision-making together with a realisation in terms of a meta model that extends Actor model of computation. The proposed approach is illustrated using a real life case study from business process outsourcing industry.

1 INTRODUCTION

Modern organisations constantly attempt to meet organisational goals by adopting appropriate courses of action (Shapira, 2002). Evaluation of the possible courses of action and selection of the best option are key challenges faced by organisations. It calls for the precise understanding of various aspects such as goals, organisation structure, operational processes, historic data and the stakeholders (Daft, 2012). The socio-technical characteristics (McDermott et al., 2013), inherent uncertainty and non-linear causality in business interactions (Conrath, 1967), and high business dynamics (Sipp, 2012) further exacerbate the complex dynamic decision making (CDDM) endeavour.

The industrial practice of organisational decision-making heavily relies on human experts who typically use tools such as spreadsheets, word processors, and diagram editors. Though adequate for capturing and collating the required information, these tools offer limited analysis support if at all (Locke, 2011). As a result, CDDM tends to be a time-, effort- and intellectually-intensive endeavour. Furthermore, reports from leading consulting organisations such as McKinsey and Harvard Business Review (Kahneman et al., 2011, Meissner et al. 2015) often classify the current state of practice as biased, based on short-term emotion and imprecise for modern business context. This perceived poor state-of-practice of decision-making elicits a research question: *What kinds of technological aids are needed for decision makers to arrive at precise, unbiased and effective decisions?*

This paper argues that the success of decision-making largely depends on two key factors: (i) the ability to capture relevant information about organisation and (ii) the ability to perform *what-if* and *if-what* analyses of relevant information in a relatable form. The former ensures completeness of information and the latter ensures reduction of analysis burden on human experts.

A wide variety of Enterprise Modelling (EM) techniques have been proposed to capture the relevant information about organisation in a formal manner amenable to rigorous analysis (Authors Ref). However, they are found to be less effective and insufficient for a class of decision-making problems characterised by significant dynamism, inherent uncertainty, and emergent behaviour (Authors Ref). Being a socio-technical system, an organisation can be best viewed as a set of interacting units each having own goals and operating with the intent of achieving them. Thus, behaviour of the entire organisation is not known a priori (and hence never specified as such) but emerges through the interactions of units each having well-defined behaviour (Hewitt, 2010). Behaviour of an organisation unit can be specified in terms of the many actor languages and frameworks available e.g., Erlang (Armstrong, 1996), SALSA (Varela and Agha, 2001), AmbientTalk (Van Cutsem et al., 2007), and Kilim (Srinivasan and Mycroft, 2008), ActorFoundry (Astley, 1998), Scala Actors (Haller and Odersky, 2009), Akka (Allen, 2013) etc. Though capable of catering to the specification of autonomous, intentional, and emergent behaviour, these languages do not provide support for uncertainty and temporal behaviour.

In this paper, we present an approach that extends the actor model of computation with uncertainty and temporal behaviour to serve as an effective aid for CDDM. In particular, this paper makes two contributions: i) a conceptual meta-model that represents necessary aspects of the organisation along with the inherent characteristics of CDDM, and ii) a realisation meta-model that concretises conceptual model by extending the core concepts of actor model of computation. Also, we illustrate the proposed approach demonstrating its efficacy with a case study from Business Process Outsourcing (BPO) domain.

We envision an overarching research agenda[1] for developing a business facing decision-making framework to improve precision of decision-making, reduce personal biases while considering decisions, consider short term and long term effects before arriving at decisions, and reduce the excessive analysis burden on human experts in decision-making process. We argue the presented contributions form a basis of such a business facing decision-making framework.

The paper proceeds as follows: Section 2 presents research motivation by highlighting necessary tenets of CDDM and reporting brief overview of state-of-the-art of EM techniques and actor language/frameworks. Section concludes by highlighting notable gaps that limit the adoption of EM techniques and actor languages/frameworks for CDDM. Section 3 presents the conceptual meta-model that has a potential to address CDDM problems. A meta-model that realises conceptual model by extending actor model of computation with relevant concepts such as uncertainty, temporal behaviour is described in section 4. The illustration of

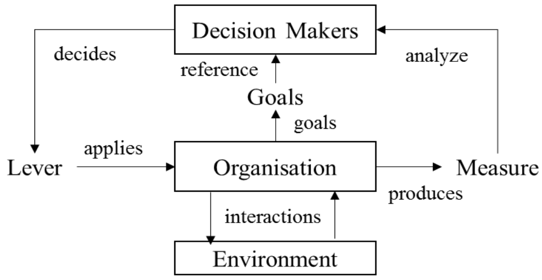


Figure 1 : Overview of decision making

the proposed approach using a case study from BPO is highlighted in section 5 and a brief evaluation of proposed approach is presented in section 6. The paper concludes with a brief summary research progress and future plan to realise the overarching research agenda.

2 MOTIVATION

An abstract representation of decision-making is presented in Fig. 1. As shown in the figure, an *Organisation* interacts with its *Environment* to achieve its *Goals*. The Goals are typically assessed by evaluating the key performance indicators or *Measures*. The decision makers evaluate/predict Measures with respect to Goals and decide appropriate courses of action or *Levers*. In this formulation, the decision-making is finding best possible Levers for stated Goals in the context of Organisation and Environment. Essentially it is an iterative and refinement based method to explore available Levers, evaluate them with respect to Goals, Organisation and Environment, and finally decide most effective options. We argue the efficacy of such exploration depends on two key factors, i.e., ability to specify information about relevant aspects of the Organisation and ability to analyse available information in the context of the Environment where it operates.

The management literature advocates multiple methods such as Incremental method (Mintzberg et al., 1976) and Carnegie Method (Cyert et al., 1963) for guided exploration of Levers for stated Goals. However, they do not prescribe or recommend any technological aid best suited for their proposed methods as their focus in not pertaining to any technological aspects. We conducted a series of literature reviews and experiments to understand - *What kinds of modelling abstractions and analysis techniques are available for specifying and analysing*

[1]Omitted for double-blind review.

Table 1: Requirements of CDDM

	Requirement	Description
Aspect	Why	Intention
	What	Structural Specification
	How	Behavioural Specification
	Who	Stakeholders and Human actors
Socio-technical Characteristics	Modular	Must encapsulate internal goal, structure and behaviour.
	Compositional	Multiple parts should be composed to a consistent whole.
	Reactive	Must respond appropriately to its environment
	Autonomous	Possible to produce output without any external stimulus.
	Intentional	Intent defines the behaviour
	Adaptable	Adapt itself based on context and situation
	Uncertain	Precise intention and behaviour are not known a-priori.
	Temporal	Indefinite time-delay between an action and its response
	Machine Interpretable	Models that are interpretable by machine (i.e., support for simulation/execution)

different aspects of an organisation? Are they capable of supporting expected characteristics of CDDM? What are the gaps?

Our experiments such as (Authors Ref) and literature review such as (Authors Ref) indicate inadequacy of state-of-the-art of relevant enterprise wide modelling and analysis techniques and tools. An overview of our explorations is presented in this section. We first describe the key tenets of CDDM that we use for evaluating the state-of-the-art specification and analysis techniques. Subsequently we discuss the findings and illustrate the gaps by examining the state-of-the-art specification and analysis techniques.

2.1 Tenets of CDDM

We argue that an Organisation can be understood well by analysing *what* an enterprise is, *how* it operates, *why* it is so, and *who* are the responsible stakeholders (Authors Ref). This hypothesis is principally aligned with the Zachman framework (Zachman et al., 1987) and industry prevalent enterprise modelling (EM) techniques such as ArchiMate (Iacob et al., 2012)

CDDM puts some special demands on specification in terms of desirable characteristics of organisation that include *reactive*, *adaptable*, *modular*, *autonomous*, *intentional*, *compositional*, *uncertainty* and *temporal*. Essentially these characteristics represent associated dynamism to interact with Environment. Furthermore, industry practice of decision-making desires precise *what-if* and *if-what* analysis for a-priori indication of a decision. As a result, a machine interpretable model forms the basis of analysis requirements. Table 1 enumerates specification and analysis requirements for CDDM.

2.1 Exploration of specification and analysis techniques

In (Authors Ref) we evaluated the suitability of EM techniques in the context of CDDM using Systematic Mapping Study methodology (Petersen et al., 2008). The evaluation concluded with a critical observation that the existing EM techniques are capable of satisfying the expected requirements of CDDM described in Table 1 in parts. In particular, we found the EM techniques that support necessary aspects of CDDM (such as Zachman Framework and ArchiMate) are not machine interpretable and thus not amenable for rigorous analyses. In contrast, specification approaches such as BPMN (OMG, 2011), i* (Yu, 2006) and Stock-n-Flow (SnF) (Meadows and Wright, 2008) are capable of sophisticated analyses and simulation. For example, the process aspect can be analysed and simulated using BPMN based tool, the high level goals and objectives can be evaluated using i*, and high level system dynamics can be simulated using Stock-and-Flow (SnF) tools such as iThink. However, they are not capable of representing all necessary aspects.

Detailed review synthesis led us to explore multi-modelling and co-simulation environments involving multiple EM techniques to address CDDM. The exploration was conducted using two activities: a) a literature review on multi-modelling and co-simulation environments such as DEVS (Camus et al., 2015), AA4MM (Siebert et al, 2010), AnyLogic (Borshchev, 2013), and b) an experiment on multi-modelling and co-simulation approach by combining i*, Stock-and-Flow and BPMN tools. The research finding, experimental setup and experiences are presented in (Authors Ref). Both the literature review and experiment on multi-modelling and co-simulation approach have produced evidence that indicate a multi-modelling and co-simulation based approach using multiple EM techniques are capable of representing necessary aspects and they collectively support the analyses needs. However, they are largely prone to intrinsic complexity (as discussed in (Authors Ref)) and accidental complexity (as discussed in (Authors Ref)). Moreover, they are not capable of expressing many

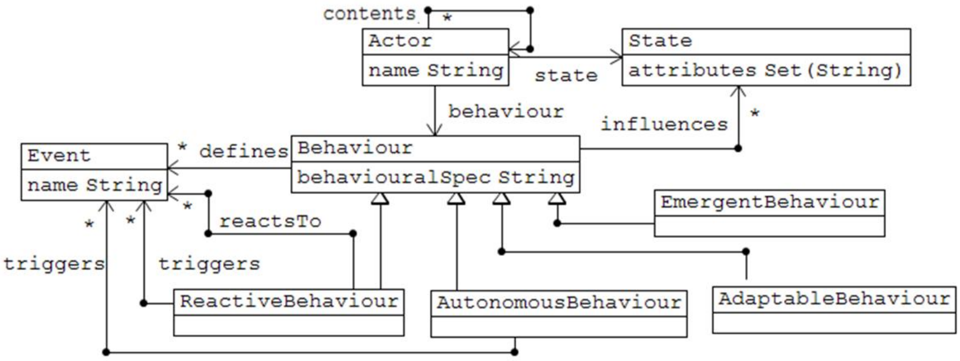


Figure 2 : Meta-model for representing Actor Language/Framework (AMModel)

socio-technical characteristics such as autonomy, uncertainty, temporal behaviour and adaptability.

The inadequate support for socio-technical characteristics in EM techniques (as in individual or within multi-modelling setup) opens up a scope for exploring the languages and frameworks that are based on the actor model of computation. A literature review on actor language and frameworks discloses their suitability in the context of CDDM. Essentially actor languages and frameworks are capable of specifying and analysing a range of socio-technical characteristics.

The key concepts and core capabilities of prevalent actor languages and frameworks (such as Scala Actor and Akka) are represented using a meta-model (termed as AMModel[2]) as depicted in Fig. 2. The concept *Actor*, a named entity, encapsulates *State*, *Behaviour* and internal Actors. A State can be specified using *attributes* and the Behaviour is defined using set of *Event* specifications. The *behaviouralSpec* of Behaviour influences State of an Actor and is capable of representing four kinds of behaviour namely *ReactiveBehaviour*, *AutonomousBehaviour*, *AdaptableBehaviour* and *EmergentBehaviour*. The ReactiveBehaviour reacts by responding to Events, AutonomousBehaviour triggers internal Events, AdaptableBehaviour describes the adaptability of an Actor using set of rules and EmergentBehaviour specify the *emergent* behaviour of an actor model (Agha, 1985).

The proposed AMModel is capable of representing the *what* aspect using the structure and composition, *how* aspect using behavioural specification and the *who* aspects using the Actor itself. It is also capable of representing characteristics such as *modular* using the notion of Actor, *compositional* using *contents* association (See Fig. 2), *reactive* using ReactiveBehaviour, *adaptable* using AdaptableBehavior and EmergentBehaviour and *autonomous* using AutonomousBehaviour. However, it is not capable of describing the *why* aspect and other characteristics such as *intentional*, *uncertainty* and *temporal* behaviour.

In the next section, we describe the necessary concepts of CDDM that satisfy the tenets described in table 1 in terms of a meta-model. Then we discuss an extended form of actor model that realises this meta-model.

3 CONCEPTUAL MODEL

We define a meta-model to represent the relevant aspects of an organisation along with the characteristics described in Table 1. The proposed meta-model, termed as CMModel, is depicted in Fig. 3. In the figure, an *OrgUnit* represents an *Organisation* which is an autonomous self-contained functional unit having high coherence and low external coupling. It has a set of *Goals* that represent its intention, *Measures* that describe the key performance indicators of OrgUnit, and *Levers* that represent possible courses of action with a potential to change an OrgUnit in terms of structure, behaviour and/or intentions. The goalSpec of Goal element uses Measure (and thus historical and current Data of an OrgUnit). The Measures are expression over Data and the leverSpec of Lever element describes the change specification or defines configuration parameters. An OrgUnit interacts with environment through a set of Events.

Internally an OrgUnit contains *Data*, *Behaviour*, *Structure* and *Participants*. Data represents the current and historical *States* of OrgUnit, i.e. current

[2] Meta-models are drawn using xModeler tool (http://www.eis.mdx.ac.uk/staffpages/tonyclark/Software/XModeler.html)

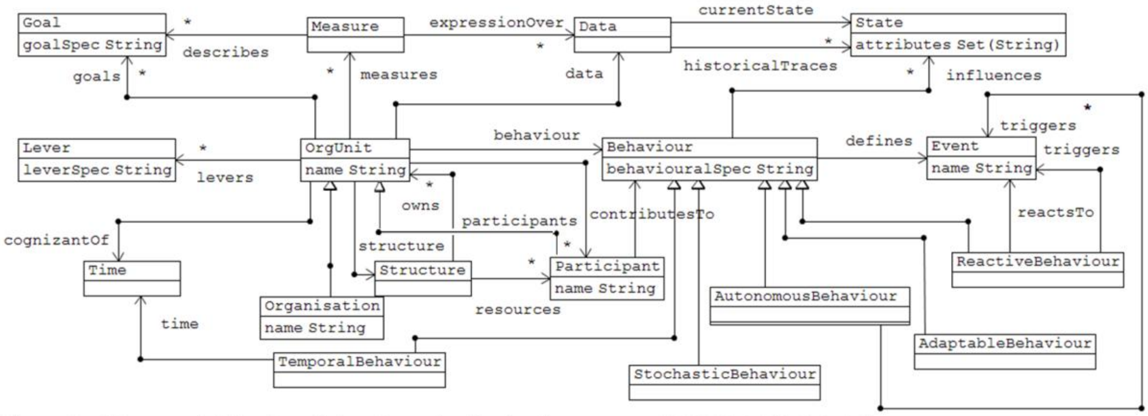


Figure 3 : Meta-model for describing Organisation in the context of CDDM (CMModel)

state and traces. The Structure of an OrgUnit is described using multiple contained OrgUnits and *Participants*. The contained units can interact with each other to delegate responsibilities or can participate in a hierarchical composition structure to accomplish higher level goals. The Participant, a specialised OrgUnit, represents the resources of the OrgUnit. Proposed meta-model advocates five kinds of *Behaviour* namely *ReactiveBehaviour*, *AutonomousBehaviour*, *StochasticBehaviour*, *TemporalBehaviour* and *AdaptiveBehaviour*. The ReactiveBehaviour represents external interactions using (external) Events and AutonomousBehaviour represents the internal behaviour using (internal) Events. The behaviouralSpec of StochasticBehaviour describes associated uncertainty of raising an Events and responding to an Event, and the behaviouralSpec of TemporalBehaviour describes the temporality of Event specification. The behaviouralSpec of AdaptableBehaviour describes adaptation rules. We introduce a concept 'Time' to represent time aspect of an OrgUnit. There could be a central 'Time' element for an Organisation or each OrgUnit may own a 'Time' element.

Conceptually, the elements OrgUnit, Event, Data, and nesting capability of OrgUnit specifies the *what* aspect, Goal specifies the *why* aspect, Behaviour specifies the *how* aspect and Participant specifies the *who* aspect of an Organisation or OrgUnit. Event helps to capture reactive nature, the intent is captured using Goal, the modularity is achieved through the concept of OrgUnit, autonomy is possible due to the concept of AutonomousBehaviour, Events and Time, and composition can be specified using nesting relation. Also, OrgUnit is adaptable as it can construct and reconstruct its structure using AdaptableBehaviour; modular as it encapsulates Data, Structure and Behaviour; intentional as it has its own goals; and compositional as it can be an assembly of OrgUnits. The StochasticBehaviour and TemporalBehaviour are capable of representing associated uncertainty and temporality.

Thus we argue this meta-model conceptually covers all the specification needs stated in table 1 and a machine interpretable specification realising this meta-model can serve the analysis need.

The proposed meta-model is grounded with a set of existing concepts. The modularisation and reactive unit hierarchy are taken from component model concepts. Goal-directed reactive and autonomous behaviour can be traced to actor behaviour (Agha, 1985). Defining states in terms of a type model is

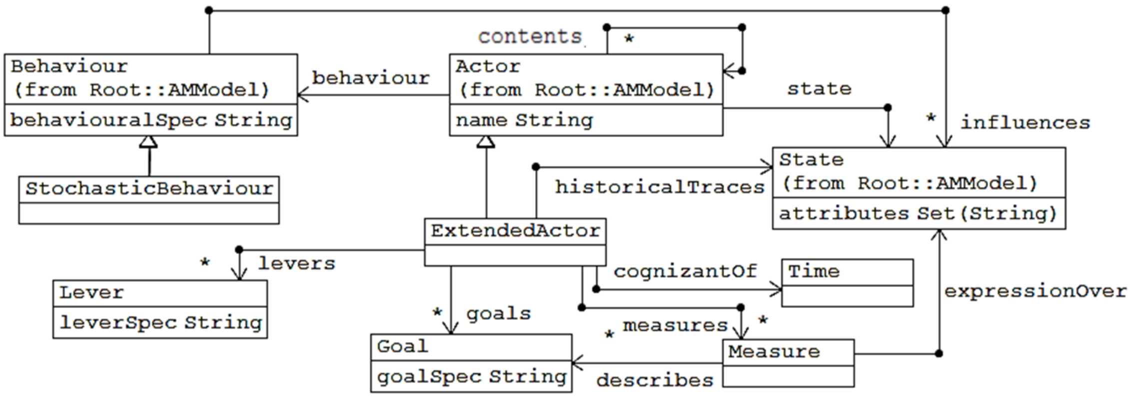


Figure 4 : Extended Actor meta-model (EAMModel)

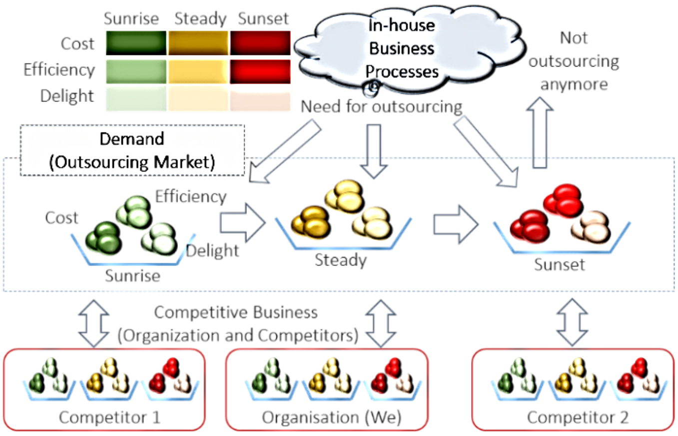


Figure 5 : Overview of Business Process Outsourcing Scenario

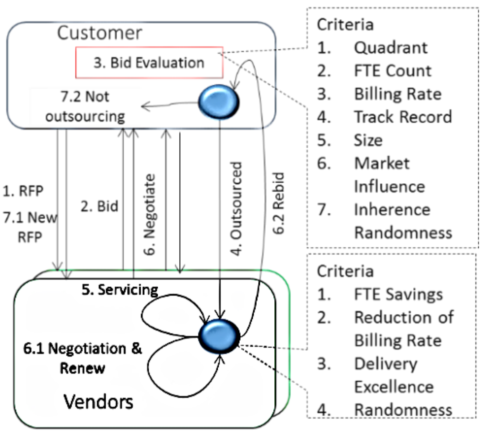


Figure 6 : Interactions and Behaviours

borrowed from UML. An event driven architecture is introduced for reactive behaviour and the concept of intentional modelling (Yu, 2006) is adopted to enable specification of goals.

4 REALISATION MODEL

In this section we propose extensions to the actor meta-model (AMModel) of Fig. 2 to realise the proposed conceptual model (CMModel) of Fig. 3. The extensions are presented using a meta-model (termed as EAMModel) in Fig. 4. As shown in the figure, the concept of Actor (described in AMModel) is primarily extended with the concepts of *Goal*, *Measure*, *Lever* and *Time*. The extended Actor is represented as ExtendedActor in EAMModel. The concepts *Goal*, *Measure*, *Lever* and *Time* associated with ExtendedActor of EAMModel conforms to *Goal*, *Measure*, *Lever* and *Time* concepts introduced in CMModel.

The *Behaviour* of an Actor in AMModel is extended with two additional behavioural types namely StochasticBehaviour and TemporalBehaviour wherein the StochasticBehaviour and TemporalBehaviour of EAMModel conform to the definitions of StochasticBehaviour and TemporalBehaviour of CMModel respectively.

We argue, the extended actor model (i.e., unified version of AMModel and EAMModel) is capable of realising the conceptual model represented in CMModel. The concept ExtendedActor of AMModel is capable of representing OrgUnit, Organisation and Participants as the concept ExtendedActor is an encapsulated, modular, autonomous, composable entity.

Actor of AMModel (and thus ExtendedActor of EAMModel) is capable of representing its current states using *attributes* of *State* entity. ExtendedActor definition is further capable of representing traces using *historicalTraces* association to *State* entity. Thus the unified meta-model (of AMModel and EMModel) is capable of representing *Data* of CMModel.

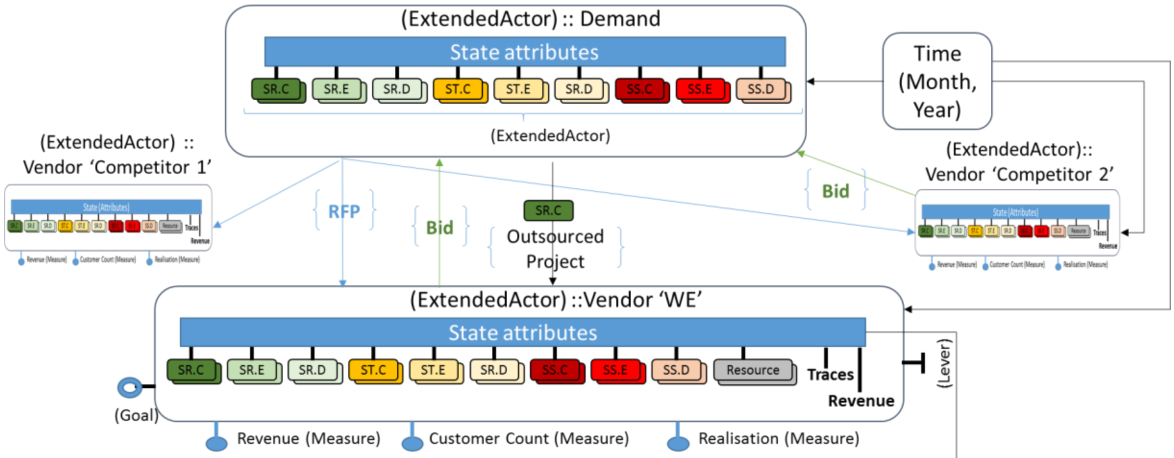
Similarly, the *Actor* described in AMModel is capable of representing reactive, autonomous and adaptable behaviour using ReactiveBehaviour, AutonomousBehaviour and AdaptableBehaviour. ExtendedActor in EAMModel further introduces the StochasticBehaviour and TemporalBehaviour. Thus we argue they collectively represent all necessary behavioural types described in CMModel.

Finally, the extensions Goal, Measure, Lever and Time to the conventional actor meta-model help in realising conceptual model that is necessary for CDDM.

We have conceptualised a language termed as Enterprise Simulation Language (ESL) that implements the concepts of conventional actor model as depicted in Fig. 2 along with the extension proposed in Fig. 4. We have developed a prototype implementation of ESL using DrRacket[3]. We have also developed a prototypical simulation engine to iterate over the "apply Lever – Observe Measure – Analyse the feasibility of Goals" loop as depicted in Fig. 1. The simulation machinery comprises of Spreadsheet, DrRacket based ESL execution engine and Python wherein Spreadsheet is used for specifying *Lever* configuration, ESL engine for simulation, and Python for visualising *Measures* and *Goals*.

We illustrate the proposed realisation model and simulation capability of its implementation using an

[3] https://racket-lang.org



	Quardant	Billing Rate ($/Hr per FTE)		FTE Productivity	Negotiation Levers after term completion			Delivery Excellence
		Min	Max	Less FTE (in %) in Bid Quotation	Proposed FTE Reduction (%)	Proposal of Billing Rate Reduction (%)	Influencer Relation	Ration of [Excellent,Good, Normal, Below Normal]
Cost	Leader	8	12	8	2	5	Excellent	[60, 30,10, 0]
Efficiency	Visionary	18	22	6	0	5	Good	[50, 40,10, 0]
Delight	Contender	110	130	4	5	0	Normal	[20, 60,20, 0]

Figure 7 : Realisation of BPO scenario using extended actor meta-model

industrial case from BPO domain. The next section presents the case study and observed results.

5 ILLUSTRATION

Consider the business process outsourcing (BPO) domain. *Customers* outsource *business processes* for a variety of reasons such as reducing *Cost* (C), increasing *Efficiency* (E), bringing about a major transformation, i.e., *Delight* (D). The outsourced processes can be classified into three buckets based on maturity of BPO verticals. For instance, Transcript Entry process of Healthcare vertical is one of the early adopter of BPO and has derived almost all potential benefits accruable from outsourcing (termed as *Sunset* or SS). On the other hand, IT Infrastructure Management process being a late adopter of BPO has a large unrealized potential to be tapped (termed as *Sunrise* or SR). And there are processes such as Help Desk, Account Opening, Monthly Alerts etc., that fall somewhere in between the two extremes as regards benefits accrued from BPO (i.e. *Steady* or ST). Thus, BPO demand space can be viewed as set of customers where each customer is characterised by one of the type described using a 3 x 3 matrix of Fig 5.

A customer invites bids from the *vendors* for a specific BPO outsourcing project. Typically, factors such as *Quadrant* (i.e. ranking as per independent agency such as analysts), *FTE Count Range* (i.e. full time employees to be deployed on the outsourced process), *Billing Rate Range* (i.e. per hour rate of full time employee), *Organisation Size*(the number of employee) and *Track Record* (i.e., familiarity with the processes being outsourced), influence who wins the bid. Other soft issues such as *Market Influence* (i.e. perception of the market as regards delivery certainty with acceptable quality), the rapport with the vendor etc., also play a part in bid evaluation. In addition to these known factors there could be some uncertainty in bid evaluation criteria (in other words, bid evaluation criteria can't be fully known a-priori).

It is common observation that BPO outsourcing projects come up for renewal after few years (typically 3 to 5 years). Customer may renew the contract with the existing vendor on modified terms (typically advantageous to the customer) or may opt for rebidding. Factors influencing the renewal decision are reduction offered in *FTE Count*, *Billing Rate*, number and degree of escalations, perception that the external agent has as regards ability to meet the project requirements, inherence uncertainty, etc. Contracts that fail to get renewed become candidates for later bidding. Fig 5 shows a high level schematic of BPO space. The events of interest illustrating the interactions between customer and vendor, and the state transition of outsourcing project are depicted in Fig 6.

The demand space exhibits temporal dynamism. For instance, new processes emerge as candidates for outsourcing and some of the existing processes no longer need to be outsourced as, say, technology advance eliminates the need for human intervention

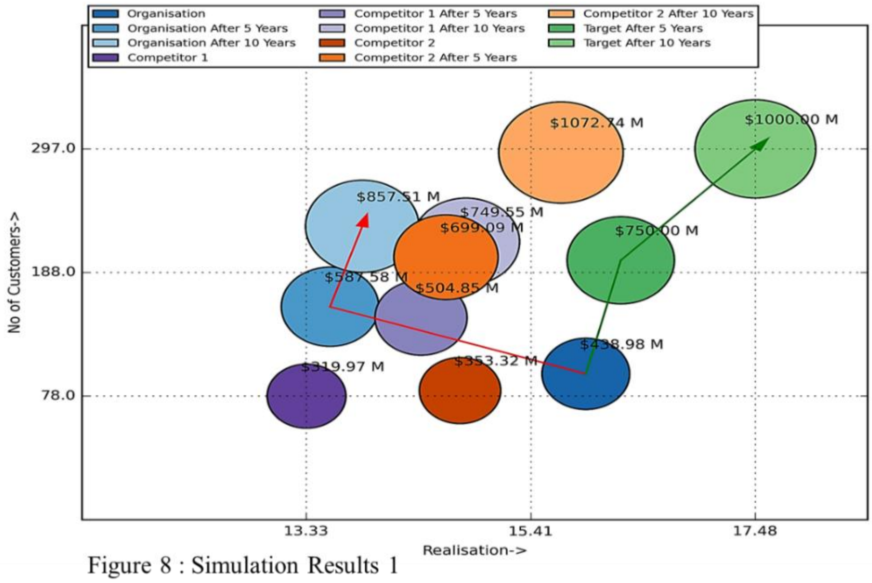


Figure 8 : Simulation Results 1

in the process thus making it straight-through. Thus the BPO space can be viewed as an event-driven system where events have a certain frequency and are stochastic in nature. The frequency and stochastic characteristics typically vary from process to process. While operating in this uncertain space, a BPO vendor needs to make decisions of the following kind: Will continuation with the current strategy keep me viable 'n' years hence? What alternative strategies are available? How effective will a given strategy be? By when will a given strategy start showing positive impact? Will I be growing at the expense of competition or vice versa? and so on.

Answers to the above questions are essentially linked to the evaluation of portfolio basket i.e., 3 x 3 matrix of Fig 5, of the organisation in terms of revenue accruable and expenses in terms of FTEs., The ability to predict the portfolio basket of the organisation and its competitors after a given time period becomes critical to support decision-making.

5.1 Realisation using Actor Model

We model BPO scenario using our extended actor model (i.e., the unified meta-model of AMModel depicted in Fig. 2 and the EAMModel depicted in Fig. 4). The key elements *demand*, *vendor* and *outsourced projects* are modelled as *ExtendedActor* as shown in Fig. 7. The demand ExtendedActor comprises of nine attributes where each attribute represents a bag of outsourced projects of specific type from the demand classification i.e., {SR, ST, SS} X {C, E, D}. Each outsourced project *ExtendedActor* represents a specific demand classification using its *State* attributes and implements the state-machine depicted in Fig. 6. The increase (or decrease) of specific type of outsourced project in demand space is specified by instantiating (or destructing) specific outsourced project *ExtendedActor*.

Vendors *ExtendedActors* has a set of *State* attributes to represent portfolio baskets (i.e., flattened out 3 x 3 matrix), *Resources*, *Traces* and other attributes such as *Revenue*. The portfolio basket represents the bags of outsourced project ExtendedActors, The *State* attribute *Resource* contains a bag of ExtendedActors that represent the FTEs and the State attribute *Traces* is historical data about the values of *State* attributes of vendor such as *Revenue* at specific time. The characteristics of a vendor such as *Quadrant*, *Billing Rate*, *FTE count*, *Market Influence* and *Delivery Excellence* are also represented using *State* attribute of vendor *ExtendedActor*. In this formulation, one vendor is marked as 'We' and rest are classified as competitors. This example considers two competitors – *Competitor 1* and *Competitor 2* as depicted in Fig. 7.

The table in Fig 7 shows the initial characteristics of 'We' ExtendedActor. We make these characteristics configurable to attenuate their values, thus these *State* attributes also act as *Lever* in this example. As shown in the figure, a vendor is equipped with a set of negotiation levers namely, the range of *Billing Rate* (employees billed against the outsourced process), range of *FTE Productivity* (percent reduction possible in number of full time employees), range of *FTE Reduction* (reduction possible during renewal of a contract), range of *Billing Rate Reduction* (reduction possible in billing rate during renewal of a contract), *Influence Relation* and *Delivery Excellence*. The Influence Relation is a qualitative characteristic that is quantified using four weighted labels namely 'Excellent', 'Good',

	Quardant	Billing Rate ($/Hr per FTE)		FTE Productivity	Negotiation Levers after term completion			Delivery Excellence
		Min	Max	Less FTE (in %) in Bid Quotation	Proposed FTE Reduction (%)	Proposal of Billing Rate Reduction (%)	Influencer Relation	Ration of [Excellent,Good, Normal, Below Normal]
Cost	Leader	12	14	10	5	1	Excellent	[90, 10,10, 0]
Efficiency	Visionary	25	30	10	5	1	Good	[80, 10,10, 0]
Delight	Contender	110	120	10	5	1	Excellent	[70, 30,0, 0]

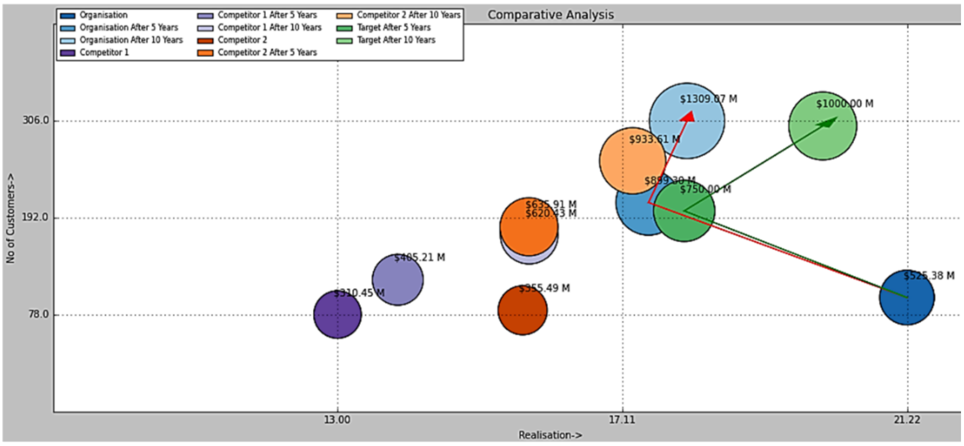


Figure 9 : Lever Specification and Simulation Results 2

'Normal' and 'Not Good'. Value of *Delivery Excellence* attribute is a probability distribution. For instance, 'We' ExtendedActor is confident of delivering 'Excellent' quality on 60% of Cost kind of BPO projects won. The values for 'Good', 'Normal' and 'Below Normal' quality for this kind of BPO projects are 30%, 10% and 0% respectively.

There are many *Measures* that are of interest to the stakeholders of 'We' vendor. Fig. 7 depicts three key *Measures* namely *Revenue* (i.e., the State attribute *Revenue*), *Customer Count* (i.e., total number of outsourced projects contain by 'We' vendor) and *Realisation* (i.e., average *Revenue* earned by each Resource per hour) for illustration purpose.

One can model different kinds of vendor by setting appropriate values to the initial setting. The 'Competitor' ExtendedActors are modelled on the same lines as 'We' ExtendedActor.

This example considers a central *Time* to synchronise the entire business-as-usual (i.e., *Behaviour*). Time triggers two *events* namely 'Month' and "Year" to the demand and vendor ExtendedActors. Demand ExtendedActor raises RFP events for outsourcing project. Each RFP event is characterized by the kind of process being outsourced (i.e. SR or ST or SS), the objective for outsourcing (i.e. C or E or D), size of the process in terms of FTE count, and the desired billing rate. Interested vendors respond to the RFP event by picking suitable values from their characteristics at random. Bid evaluation function is a weighted aggregate of the various elements of RFP response and a random value to capture effect of inherent uncertainty. The vendor with the best evaluated value wins the outsourcing project which gets executed as defined by the characteristics of the particular vendor. Essentially, an outsourcing project ExtendedActor moves from Demand ExtendedActor to a vendor ExtendedActor (i.e., from demand basket to vendor portfolio basket) as shown in Fig. 7. The existence of an outsourcing project within a vendor impact vendor's *State* attribute (and thus Measures) as outsourcing project consumes the resources and contribute the revenue, the customer count and other measures. It also impacts the track record and market influences over the time.

The decision to renew existing contract is modelled on similar lines but with a different set of characteristic attributes influencing the decision. Essentially the autonomous outsourcing project ExtendedActor raises *Renew* event after 3 to 5 years timeframe. Here too, the evaluation is cognizance of incomplete and uncertain knowledge renewability criteria.

5.2 Simulation and Results

We use the simulation environment developed using DrRacket and Python to run the system for 10 years. In the interest of space, two decision questions from many scenario playing are discussed in this paper. Decision questions are – i) *Will continuation with the current strategy keep me viable 'n' years hence with respect to the competition*? And ii) *What* *will be my position if we decide to change my characteristics?*

An overview results of simulation run is shown in Fig 8. As can be seen, the current revenue of ‘We’ (represented using shades of ‘blue’ disks) is 438.98 MUSD from 90 customers with a realization of nearly 15.5 USD per hour per FTE. Corresponding numbers for competitor 1 and competitor 2 respectively are < 319.97, 78, 13.33 > (depicted using shades of ‘violet’ disks) and < 352.32, 79, 15.1 > (depicted using shades of brown disks). In short, at present ‘We’ vendor is doing much better than competition.

‘We’ vendor set a goal to deliver <750, 200, 17> after 5 years and <1000, 290, 18> after 10 years (depicted using green disk). As can be seen, by continuing to operate the same way the ‘We’ vendor will be delivering <587.58, 160, 13.5> after 5 years and <857.51, 215, 14> after 10 years (as directed by red line in Fig. 10) thus missing both the targets by a considerable margin. More importantly, competitor 2 will be overtaking ‘We’ vendor after 5 years and both the competitors will be significantly ahead of ‘We’ vendor after 10 years.

Clearly, ‘We’ vendor cannot afford to continue with its current way of operation. Further detailed analysis, involving model elements not described in this paper for want of space, shows that much of current revenue of ‘We’ vendor is from sunset kinds of outsourced processes for cost reasons. Over time this market is going to shrink considerably with demand for steady as well as sunrise processes (for objectives other than pure cost reduction) increasing significantly. Thus ‘We’ vendor needs to bring about a change in its characteristics so as to be able to win more bids in this demand situation. Fig 9 shows the modified characteristics of ‘We’ vendor leading to the improved performance as shown in Fig 9. With the changed parameters, the ‘We’ vendor is able to beat both revenue and customer targets while failing to meet the realization target narrowly.

6 EVALUATION

For the kind of decision-making problem illustrated in this paper, industry practice relies extensively on Excel spreadsheets. Such an approach typically represents the influence of lever onto measures in terms of static algebraic equations. However, value of a lever and influence of a lever onto a set of measures can vary over time. This behaviour cannot be captured using excel sheet. There is no support for encoding stochastic behaviour either.

Stock-and-Flow models are also used for a class of decision-making. In this approach, the system behaviour is expected to be known a priori. Essentially the system is specified in terms of stocks, flows of stocks, and a fixed set of equations over system variables that control the flows. Value of a stock or a flow or a variable is a discrete number or a range or a distribution. The quantitative nature of Stock-and-Flow models and sophisticated simulation support enables decision-making through what-if scenario playing. It is possible for a stock or an individual variable to have a value that is a probability distribution, however, the structure of the stock-n-flow model must remain unchanged. Thus, systems dynamics modelling provides only a partial support for specifying and processing the inherent uncertainty within a system. Moreover, it is best suited for an aggregated and generalized view of a system where individual details get eliminated through averaging, and sequences of events are grouped as continuous flows. This generalized approach and ignorance of individual characteristics that significantly influence the system over time often leads to a model that is somewhat removed from reality. Though not designed to specify specialized behaviour, it can be done using systems dynamics modelling. But this is an effort-intensive endeavour, and more importantly leads to model size explosion (Authors Ref). For example, modelling of 4 competitors each having special characteristics leads to roughly 4 times increase in the size of systems dynamics model.

The proposed approach enables modelling of a system as a set of units each listening/responding/raising events of interest and interacting with other units by sending messages. A unit encapsulates state (i.e. a set of *State* attributes), trace (i.e. events it has responded to and raised till now) and behaviour (i.e. encoding of reactions). As the modelling abstraction supports ‘time’ concept, value of a variable and relationships between variables can change with respect to time. Consider the example of determining the impact of track record on bid win of organisation where the value of track-record variable changes over time thus affecting bid win factor. Since a process is an individual actor, simulator can determine the impact of successful contract completion, renewal with/without negotiation etc., for that specific process – systems dynamics model falls short here. A trace of events serves as a memory that can be queried to establish more complex relationships between levers. For example, successful completion of contract leads to improved track record as well as better rapport with the customer thus improving the bid win factor of future outsourcing bids everything else remaining the same. Thus, the abstraction provides primitives for creating models that closely mimic reality.

7 CONCLUSION

Effective decision-making is a challenge that all modern enterprises face. It requires deep understanding of aspects such as organisational goals, structure, operational processes. Large size, socio-technical characteristics, and increasingly fast business dynamics make this activity much more difficult task for decision makers. Inadequate support for representing necessary aspects of an organisation in a relatable form and inability to handle inherent uncertainty and temporal characteristics are the present lacuna in state-of-the-art technological aids that are used in decision-making.

This paper shows the gaps by evaluating technological aids with respect to the needs of complex dynamic decision-making. We began by outlining a conceptual model (i.e. CMModel) that has potential to mitigate the identified gaps between the available technical capabilities and expected characteristics. We then argued that an extended form of actor model (i.e., AMModel and EAMModel) can address these needs. We validated the hypothesis through an industry scale case study from BPO domain. We have shown how the case study can be modelled in terms of the proposed realisation model that is an extension of actor model of computation for complex dynamic decision making. We have shown how simulation of this model helped in identifying the most appropriate of the available alternatives at each decision point. Thus, it can be said that the proposed approach can be used to define purpose-specific strategy and/or evaluate the most appropriate from a set of candidate strategies.

We acknowledge this paper does not discuss the language constructs of ESL, but, principal objectives of paper were: establish the core concepts of CDDM, correlate the core concepts with actor model of computation, and propose the necessary extensions to actor model for supporting complex dynamic decision-making.

Our next step is to use the proposed extended actor meta-model and its implementation in the form of ESL for developing a business-facing decision-making framework that will improve the precision of decision-making, reduce personal biases while considering decisions, consider short term and long term effects before arriving at decisions, and reduce the excessive analysis burden on human experts in decision-making process.

REFERENCES

Agha, G.A., 1985. Actors: A model of concurrent computation in distributed systems. Tech. rep., DTIC Document

Allen, J., 2013. Effective akka. O'Reilly Media, Inc.

Armstrong, J., 1996. Erlang - a survey of the language and its industrial applications. In: Proc. INAP. vol. 96.

Astley, M., 1998. The actor foundry: A java-based actor programming environment. University of Illinois at Urbana-Champaign: Open Systems Laboratory

Borshchev, A., 2013. The big book of simulation modeling: multimethod modeling with AnyLogic 6. AnyLogic North America Chicago

Camus, B., Bourjot, C., Chevrier, V., 2015. Combining devs with multi-agent concepts to design and simulate multi-models of complex systems. In: Proceedings of the Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium. pp. 85-90

Conrath, D.W., 1967. Organizational decision making behavior under varying conditions of uncertainty. Management Science 13(8), B-487

Cyert, R.M., March, J.G., et al., 1963. A behavioral theory of the firm. Englewood Cliffs, NJ 2)

Daft, R., 2012. Organization theory and design. Nelson Education.

Haller, P., Odersky, M., 2009. Scala actors: Unifying thread-based and event-based programming. Theoretical Computer Science 410(2), 202-220

Hewitt, C., 2010. Actor model of computation: scalable robust information systems. arXiv:1008.1459.

Iacob, M., Jonkers, D.H., Lankhorst, M., Proper, E., Quartel, D.D., 2012. Archimate 2.0 specification: The open group, Van Haren Publishing

Kahneman, D., Lovallo, D., Sibony, O., 2011. Before you make that big decision. Harvard business review 89(6).

Locke, E., 2011. Handbook of principles of organizational behavior: Indispensable knowledge for evidence-based management. John Wiley & Sons

McDermott, T., Rouse, W., Goodman, S., Loper, M., 2013. Multi-level modeling of complex socio-technical systems. Procedia Computer Science 16, 1132-1141

Meadows, D.H., Wright, D., 2008. Thinking in systems: A primer. Chelsea Green Publishing

Meissner, P., Sibony, O., Wulf, T., 2016. Are you ready to decide? McKinsey Quarterly, April. 2016

Mintzberg, H., Raisinghani, D., Theoret, A., 1976. The structure of unstructured decision processes. Administrative science quarterly pp. 246-275.

OMG., 2011. Business Process Model and Notation (BPMN), Version 2.0 (January 2011)

Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M., 2008. Systematic mapping studies in software engineering. In: 12th international conference on evaluation and assessment in software engineering. vol. 17, pp. 1-10.

Shapira, Z., 2002. Organizational decision making. Cambridge University Press.

Siebert, J., Ciarletta, L., Chevrier, V., 2010. Agents and artefacts for multiple models co-evolution: building complex system simulation as a set of interacting models. In: 9th International Conference on Autonomous Agents and Multiagent Systems: 509-516.

Sipp, C.M., Elias, C., 2012. Real Options and Strategic Technology Venturing: A New Paradigm in Decision Making, vol. 31. Springer Science & Business Media.

Srinivasan, S., Mycroft, A., 2008. Kilim: Isolation-typed actors for java. In: European Conference on Object-Oriented Programming. pp. 104-128.

Varela, C., Agha, G., 2001 Programming dynamically reconfigurable open systems with salsa. ACM SIGPLAN Notices 36(12), 20-34.

Van Cutsem, T., Mostinckx, S., Boix, E.G., Dedecker, J., De Meuter, W., 2007. Ambienttalk: object-oriented event-driven programming in mobile ad hoc networks. In: Chilean Society of Computer Science, XXVI International Conference of the. pp. 3-12.

Yu, E., Strohmaier, M., Deng, X., 2006. Exploring intentional modeling and analysis for enterprise architecture. 10th IEEE International Enterprise Distributed Object Computing Conference Workshops.

Zachman, J., et al., 1987. A framework for information systems architecture. IBM systems journal 26(3), 276-292