

Proceedings of the 2017 Winter Simulation Conference

W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

EXPLORING PERSONAL DATA FUTURES TRADING WITH DESIGN FICTION BASED HYBRID SIMULATION

David Bell

Department of Computer Science
Brunel University London,
Uxbridge, UB8 3PH, UK.

ABSTRACT

Personal data underpins much of our digital lives with recommenders proposing products and services that themselves result from personal usage data analysis. The ownership and use of personal data is central to much current legislation in an ever changing commercial environment. Design fictions are utilized here to explore a future where consumers (termed ‘prosumers’) take ownership of their personal data and offer it to a marketplace of data buyers. Personal data trading offers a disruptive opportunity to empower the end user to realize the value of their own data using new electronic trading platforms that aggregate data in response to buyer requirements. An personal data exchange-based trading platform is described where data content classification determines a selling facade (or ‘persona’). Exploratory agent-based and system dynamics models emerge and are used to examine behavior, market dynamics and process flow within this fictional trading scenario.

1 INTRODUCTION

Understanding possible futures is naturally difficult when disruptive change is taking place. Fictions are able to motivate debate, with popular examples being science-fiction narrative or translation into films such as *Minority Report*. Bruce Sterling coined the term Design Fiction, more recently adding definition, stating “It’s the deliberate use of diegetic prototypes to suspend disbelief about change.” (Sterling 2012). Diegetic prototyping is an important distinction as such a proto design is focusing on potential objects and services as opposed to entire future worlds. Understanding fictional design in an artificial world is challenging. Effective solutions may not coincide with the best or optimal solution. Effective solutions may be explained as a symptom of satisficing (Simon 1996) - searching for satisfactory solutions, without explicitly specifying all possible solutions. Design Science Research with its focus on constructs, model, methods and instantiations (March & Smith 1995) is also able to ground such diegetic prototyping – providing props within a narrative vision. Simulation is then able to explore the dynamics of such a model. Understanding possible futures requires the designer to construct artefacts in order to better understand an artificial phenomenon and the journey required from current thinking. In this paper we go further and use a fictional design with developed artifacts as a basis for simulating a future in which they reside – utilizing a hybrid approach combining agent based trader behavior and marketplace system dynamics. Macal and North (2007) highlight the applicability of agent-based modeling in fictional trading and other market simulations. Fictional simulation can also be seen in game-based (Anderson et al. 2008) or role-play based contexts (Warren et al. 2005), but here we are interested in uncovering and exploring new constructs that underpin future settings. Artifacts, as identified by March and Smith (1995), can take the form of constructs, models, methods and instantiations. More specifically, IT artifacts, (Orlikowski and Iacomo. 2001) are bundles of material and cultural properties organized in some socially recognized form (examples being hardware and software). March and Smith (1995) synthesis of process and artifact is chosen as it enables the researcher to “do” and “observe” design as process steps - not only observing finalized instantiations. Focusing on specific artefacts enables activities to be more streamlined and focused in a particular area.

Furthermore, constructs may be a focus in a multi-paradigmatic community for the sole purpose of identifying relevant constructs for further investigation. Newell and Simon (1976) formalize this concept origination process when they state (as part of their description of physical symbol systems):

Two notions are central to this structure of expressions, symbols and objects: designation and interpretation... An expression designates an object if, given an expression, the system can either effect the object itself or behave in ways dependent on the object. The system can interpret an expression if the expression designates a process and if, given the expression, the system can carry out the process. (Newell et al. 1976)

Information Technology has been playing a pivotal role in reshaping trading marketplaces from early electronic platforms to current Web and mobile services. In particular, distributed electronic markets have typically required dedicated architectures able to provide the flexibility and robustness for high-frequency high-volume electronic trading (Fan et al. 1999). Early currency and equity trading platform constructs underpin much of this evolution, e.g. matching buyers and sellers, settlement and trade expiry. Simulation of marketplaces has always been a popular topic – including more recent service marketplaces (Fujita and Kase 2016).

Adopting a design-science, problem-solving paradigm, we introduce a number of innovative artefacts in order to design a novel approach to construct elicitation for design fictions – addressing a diversity of perspective in a future context. The paper begins with a description of a personal data trading fiction before designing a prototype system and associated simulation models. A trading platform is then presented that includes two cloud-based services (a data vault and trading exchange). Process flows and performance measures are then used to further understand design elements. Agent based modeling is used to explore trading behavior and system dynamics is then employed (using Vensim) to explore this behavior in a fictional trading marketplace. Importantly, both modelling approaches are used to uncover constructs for a personal data trading design fiction.

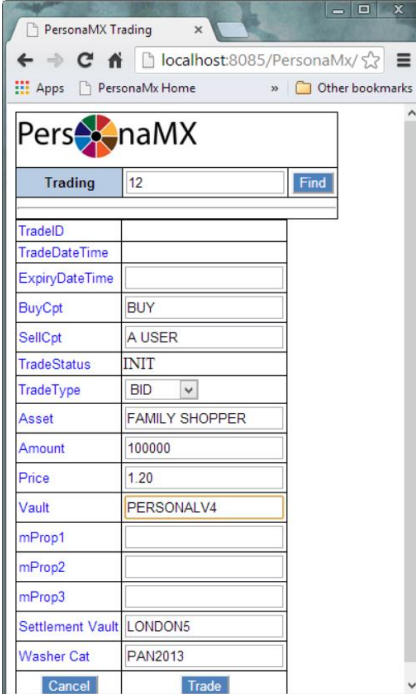
2 PERSONAL DATA TRADING

A design fiction proposes a near future where empowered consumers (‘prosumers’) can acquire their own personal data from service providers, including their cars, houses and smartphones. In contrast, product or service providers, and their marketing agents, require ever greater insight about current and future consumer, their habits and preferences. Trading exchanges are able to automatically aggregate offers from users (who provide data) with bids for that data by marketers. One approach to exploring this future is to build a diegetic prototype that uncovers the design elements, behavior and data flows.

Traditional trading systems rely on an underlying asset naming convention. When currency trading, we recognize USD or GBP. Equity trading use a mnemonic to represent the company (e.g. BARC for Barclays Bank PLC.) Data trading will also require a similar naming system to provide suitable instruments for trading. The approach adopted draws on equity and option trading convention. Data sets are described using two properties: 1) Data type and 2) Persona type. Persona, first described by Cooper (2004), is a user categorization, abstraction or archetype that details goals, fears, and typical activities that motivate and justify scenarios [8]. Clustering offers a more quantitative formation of persona properties (or characteristic). Persona characteristics are given a name for use in the traded instrument (e.g. WEEKLY SHOPPER) to easily identify and allocate prosumers. For example, “CELLPHONE”||”WEEKLY SHOPPER” describes data sets that include cellphone usage by a weekly shopper. A final indicator was then added to the naming convention to describe the date when datasets are delivered to the buyer. Option style mnemonics are added to the coding with the letters from A to L representing the delivery month. Therefore, the coding comprises:

```
<Data type> || <Persona Type> || <Delivery Month> || <Year>  
e.g. “CELLPHONE FAMILY-SHOPPER C 2020” where C represents March and matched trades  
would need to deliver data in March 2020.
```

This proto-design has a focus on three use cases. Firstly, on-boarding of users that offer data that then requires classification within persona groupings. Secondly, the trading of data futures where contracts for data supply are confirmed within an exchange platform. Lastly, a form of settlement where payment is exchanged for data sets. Unsurprising, the number of market participants is a key to success (often-termed liquidity) as connection between a buyer and many sellers of data will likely determine market viability.



The screenshot shows a web browser window titled "PersonaMX Trading" with the URL "localhost:8085/PersonaMx/". The page features the PersonaMX logo and a search bar with "Trading" and "12" entered, and a "Find" button. Below this is a form for entering a trading order. The form fields are as follows:

TradeID	
TradeDateTime	
ExpiryDateTime	
BuyCpt	BUY
SellCpt	A USER
TradeStatus	INIT
TradeType	BID
Asset	FAMILY SHOPPER
Amount	100000
Price	1.20
Vault	PERSONALV4
mProp1	
mProp2	
mProp3	
Settlement Vault	LONDON5
Washer Cat	PAN2013

At the bottom of the form are "Cancel" and "Trade" buttons.

Figure 1. Data Trading Order Entry.

In Figure 1, a data buyer is bidding for 100,000 prosumer data sets who have a “FAMILY SHOPPER” persona. They will be pay 1.20. Matching will only occur when 100,000 prosumers are prepared to sell their shopping data for 1.20 or less.

3 PERSONAMX PLATFORM

3.1 Trading Components

Several actors and components are required in our fictional trading world. Figure 2 presents a schematic outline of the process that leads initially to the generation of digital personas, utilizing shopping data for illustrative purposes. Prosumers generate personal data as part of their everyday life, e.g. associated with their supermarket shopping. They can then upload this data into a digital vault, which then triggers a data-mining engine, processing all user data to uncover common properties and subsequently enable the assignment of relevant personas to each prosumer. Prosumers may have multiple personas. If prosumers accept the assigned persona, they are then able to use the trading engine to offer their digital data sets. Buyers of uploaded data will bid for aggregated data sets for specific personas.

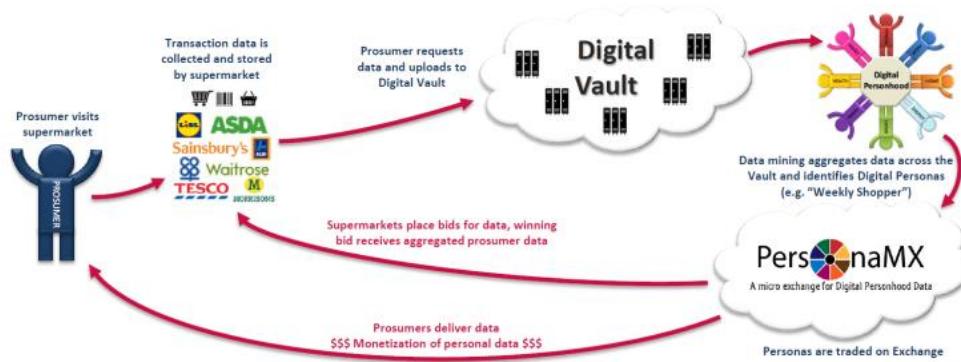


Figure 2. High level schematic of a personal data trading process.

A persona micro-exchange ('personamx') uses a traditional matching algorithm to match many data 'offers' to a single 'bid' for prosumer data. Bid and offer pricing determine trades to be matched. Importantly, each prosumer must have a persona in order to offer data in response to a specific bid.

3.2 Vault and Exchange Software Services

Data vaults are primarily concerned with storage of persona data. For example, it is envisaged that prosumers upload data or permission their devices to upload data (e.g. television viewing or heating controllers). A data vault is the main entry point for the users with regard to pre-trade and post-trade operations. Initially, pre-trade allows a prosumer to upload data for classification purposes, enabling the allocation of one or more persona to the prosumer. The vault is also used for the upload and download of data sets corresponding to matched trades – part of a settlement process where many prosumers upload data sets ready for aggregation and forwarding to the buyer. Payment is also envisaged at this point. Messages pass between the vault and exchange to: 1) Register new users, 2) allocate persona to users, 3) inform the vault of matched trades (data that needs to be uploaded in the future).

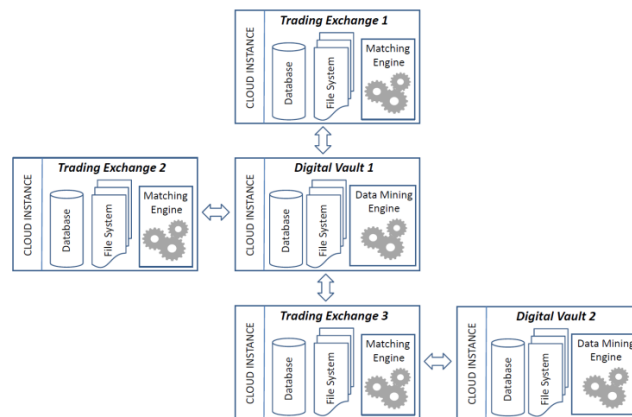


Figure 3. Cloud instances hosting trading exchanges and data vaults.

Data vault components (in Figure 3) are cloud instances with data storage and machine learning capability. Trading exchange components include matching software that appropriately matches JSON bid and offer records. Successfully matched trades have their status updated in order to indicate subsequent settlement. Cloud based instantiation of vault and trading components offer the possibility of segmenting processes based of load, data type or location. For example, a UK trading exchange could support trading activity for a specific persona. Performance of each component was examined from a technical perspective, running a number of trading scenario from 100 to 100,000 prosumers offering to sell data. With relatively

recent hardware performance (Single processor – Intel i5 5th Generation), our Java matching engine is able to support up to 20,000 offers. Consequently, if more offers are open within the matching period an additional cloud based trading service can be spawned. Two focus groups then used the software and discussed their propensity to sell their own data. A number of themes emerged from these discussions: 1) Worries about different types of data e.g. health, 2) changing motivation to sell data after successful outcomes (e.g. if data is not matched they may lose interest) and unsurprisingly 3) privacy. It was explained that all data is obfuscated and only data associated with a persona is sold (not an identifiable user). During the design, build and use of the trading platform, a number of key constructs (or objects) emerge. Some objects originate from traditional trading terminology whereas others result from this new model for trading personal data. For example, a digital data vault is a new addition to this trading context because much of the settlement process involves the aggregation and exchange of data. Objects are listed in Table 1 with a short description and a flag indicating novelty within current trading contexts.

Table 1: Fictional Constructs (or objects) emerging from instantiated system.

Object	Description	New
Trader	Buyers and sellers of data	
Instrument	Contract name that describes the trade	
Data Locker	Local personal data store that caches data before transmission to the Data Vault	Y
Data Vault	Cloud based personal data store that contains all user data (for persona identification and delivery or bought data)	Y
Persona	A user archetype traditionally used in user-centred design – adapted to represent an element of a digital personhood (see Figure 1)	Y
Prosumer	A consumer who generates and sells their own data or footprint	Y
Trades	A description of an offer to sell or buy data	
Matched Trades	Trades that result in a confirmed contract as a buyer and seller requests (open trades) are matched.	
Expired Trades	A trade that was unable to match within a defined period.	

3.3 Trading System Flows

In order to further detail the flow of data around the platform, Figure 4 depicts the interaction (and data flows) between agents and services. Implemented services uncover capabilities of services, whereas the process model informs how they can be collectively utilized by data buyers and prosumers.

After registration, data is uploaded and clustered in order to uncover personas and enable trading. Buyer bids are matched to prosumer data offers within the exchange (‘Matching Process’). Finally, matched trades are settled by delivering data and making payments.

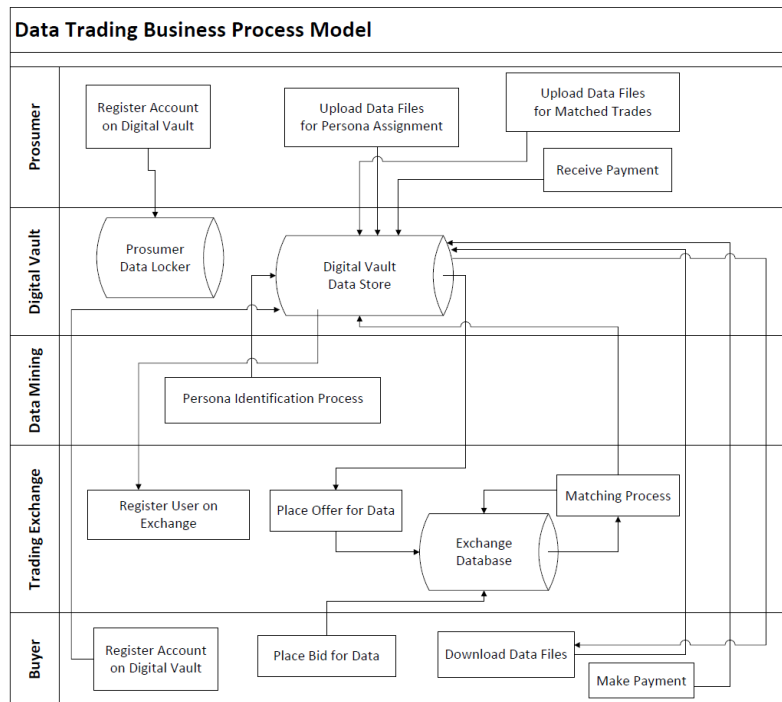


Figure 4. A process model detailing flows between stakeholders and services.

4 MODELING PROSUMER BEHAVIOUR

4.1 Simulation Platform and Agents

Future trading systems in a fictional context require exploration of prospective users. Prosumer behavior is implemented within an agent based model and built using our TEA-SIM platform. Classes of data and matching motivations were included in agent behavior. Agents were modelled using earlier constructs and themes from the focus groups. Data buyers and prosumers were modelled with motivation and data type concerns. Buyers and prosumers lose interest for a time if matching does not occur. Furthermore, buyers escalate their propensity to sell progressively more private data on successful sale of less private data. Some of this logic can be seen in Figure 5, the step functions for the persona agent. A persona is an element of personhood and additional trading personas are acquired as the system analyses prosumer data and they accept persona classification. JSON files are generated from a relational database models – with table names representing an agent type and each row an agent sub-type (with data items representing sub-type properties). For example, we have a table called Persona with rows for persona_tv, persona_familyshopping and persona_health). PHP configuration, agents and agent behavior can make use of constructs identified from system build and focus groups. We have developed a proof of concept model in this paper, but envisage that the same approach can be extended to apply the trading data generated from more extensive testing of the system. Our trading environment requires a number of agents to be implemented:

```

A persona_travel of type persona,
A persona_tv of type persona,
A persona_gas of type persona,
A persona_electricity of type persona,
A persona_familyshopping of type persona,
A persona_cellphone of type persona,
A persona_health of type persona,
A persona_buyer of type persona

```

Bell

Within the TEA-SIM platform, each agent is denoted by a specific id and set of variables. Agent variables are listed below:

```
$this->_supertype  
$this->_type  
$this->_id  
$this->_img  
$step
```

These variables are accessible to the step functions, including the pointer to a visual representation of the current agent's state (`_img`). Agent behavior is implemented in a number of available functions, such as:

```
getPosition()  
move($distance)  
anyNeighbour($distance[, $type[, $select[, $update]])  
allNeighbour($distance[, $type[, $select[, $update]])  
morph($type)
```

In order to set up the trading simulation, two configuration files are required for initialization and model execution - `init.json` and `model.json`. JSON files are easily extracted from the Tea-PoCT platform using a Web service call. The `init.json` file is where the initialization is carried out. Grid size, number of agent instances, the position of each agent and the simulation steps are defined in `init.json`. Agent attributes of class `person` are defined in `model.json` with behavior (step functions) in `steps.php`. Agent behavior specification is described in a module called `steps.php`. The stepping function, `step()` describes how each agent of a specific type progresses from one iteration to another, including state changes. In `steps.php`, the buyer or prosumer states or internal rules are being processed. Figure 5 shows the rules applied to `persona_buyer`, `persona_travel` and `persona_tv` of class `persona`.

```
class persona_buyer extends Agent {  
    function step($step) {  
        if ($this->buying) {  
            $this->seeking++;  
        }  
        else {  
            $this->sleeping++;  
            $this->move(1);  
        }  
  
        // When looking for X steps = stop buying for a while  
        if ($this->buying && $this->seeking>X)  
        {  
            $this->buying = false;  
            $this->_img = "img/notbuyer.png";  
            $this->seeking = 0;  
        }  
        // further code  
    }  
}  
  
class persona_travel extends Agent {  
    function step($step) {  
        if (mt_rand(0,1)){  
            $this->anyNeighbour(1, 1, array("buying" => true), array("buying" => false));  
            $this->morph(3);  
        }  
        $this->move(1);  
    }  
}
```

Bell

```
}  
  
class persona_tv extends Agent {.....}
```

Figure 5. Agent behavior code snippet.

Buyer and prosumer behavior can be seen. The buyer loses interest after X steps if no matching has occurred. The buyer image then changes to represent sleeping for Y number of steps. The prosumer acquires a persona (e.g. persona_travel) and once travel data is sold then morphs into another persona (e.g. persona_tv). Morphing occurs through all persona types. JSON is used to transfer data between systems (and models). For example, properties of the developed trading system can be added to the agent model (model.json). The same JSON file can then be used by further simulations - see Figure 6. Persona is a type of agent within the model and a number of subtypes with properties and initial values defined.

```
{  
  "persona": [  
    {  
      "id":1,  
      "name":"buyer",  
      "attributes": [  
        {  
          "name":"buying",  
          "value": true  
        },  
        {  
          "name":"seeking",  
          "value": 0  
        },  
        {  
          "name":"sleeping",  
          "value": 0  
        },  
        {  
          "name": "_img",  
          "value": "img/buyer.png"  
        }  
      ]  
    },  
    {  
      "id":2,  
      "name":"travel",  
      "attributes": [  
        {  
          "name":"selling",  
          "value": true  
        },  
        {  
          "name": "_img",  
          "value": "img/TravelLargeIcon.png"  
        }  
      ]  
    },  
    {  
      "id":3,  
      "name":"tv",  
      "attributes" // further definition  
    }  
  ]  
}
```

Figure 6. Agent properties.

4.2 Simulating Market Evolution

A simple 10x10 simulation is presented in this paper to demonstrate prosumer behavior. Prosumers are actively selling different data sets (for a specific persona such as FAMILY_SHOPPER – represented as a shopping cart). Buyers have green currency in their hand if active. When a buyer is next to a prosumer it will attempt to buy their data. In this example they have a 50% chance of matching. Once data is sold, the prosumer will morph into another persona and attempt to sell that data set.

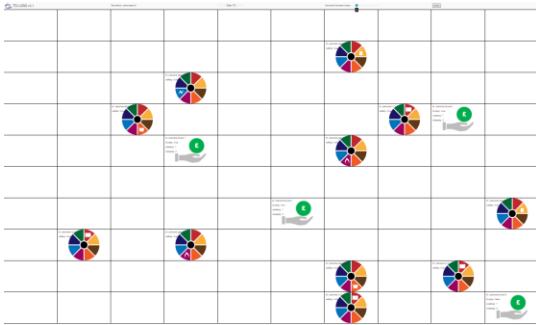


Figure 7. Agents at startup.

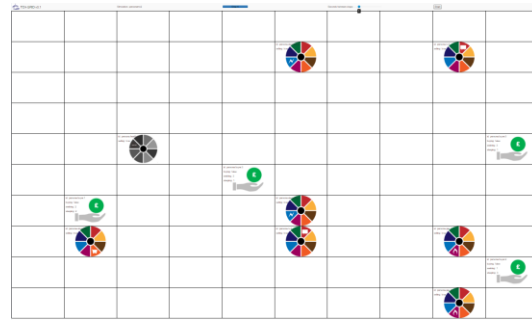


Figure 8: A prosumer not selling (all data sold).

Figure 8 depicts one of the prosumers leaving the market as they have sold all of their data for this trading period. In an ideal market, new participants would enter the marketplace. Figure 9 presents a market where no new participants are entering the market and buyers are losing interest. The topology of market participants is not sustainable and will then fail to operate. A number of simulations were executed to explore and extend persona behavior.

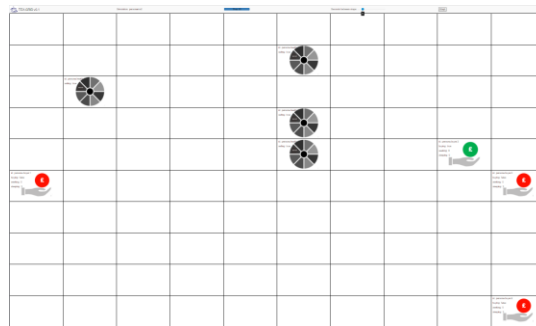


Figure 9. Buyers losing interest as no prosumers are selling.

Emerging from this exercise were a number of new constructs and these are detailed in Table 2.

Table 2: Fictional Constructs (or objects) emerging from agent simulation.

Object	Description	New
Data liquidity	Sustainable collection of data buyers and sellers	Y
Data trading propensity	The motivation to trade specific data types	Y
Market Entrants	New buyers of sellers entering the market	
Dissatisfied Traders	Buyers or sellers who are not happy with the market (e.g. time to buy or sell, pricing etc.). Seeking deals when happy or sleeping when not.	

5 MODELING A DATA TRADING MARKETPLACE

A first system dynamics model was then constructed to explore prosumer populations and their impact on the marketplace – transferring objects from the diegetic prototype and agent model behavior into a wider system model. This hierarchical format of hybridization enables agent behavior to flow into more strategic marketplace models (Chahal and Eldabi 2008), including sensitivity analysis.

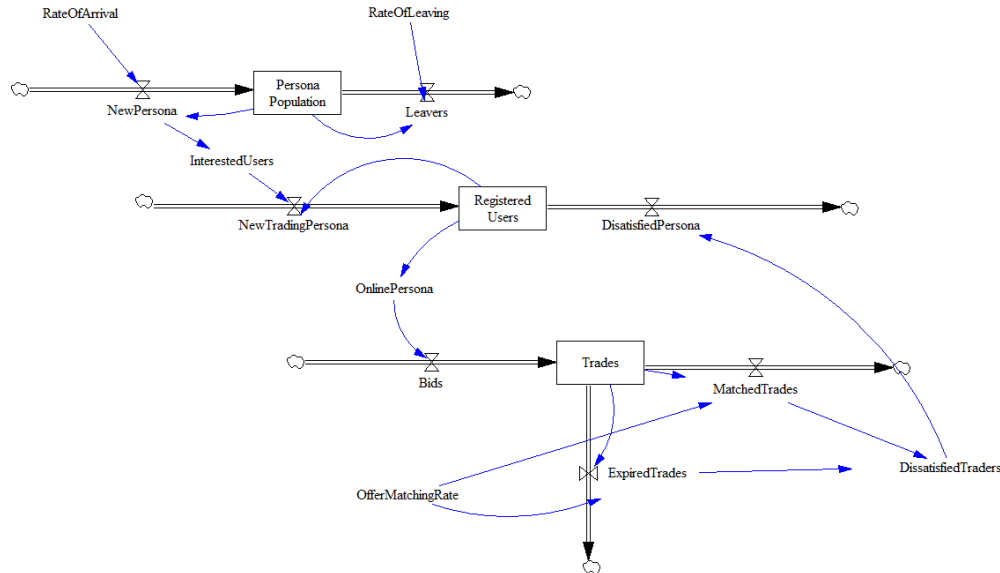


Figure 10. Personal Data Trading model (for a single persona).

System dynamics is able to provide a range of possible alternate futures in which to test the design. Key flows are identified from Figures 2 and 3 – user registration (with associated data uploading), trade entry and matching processes. Without real data from which to validate the model, our purpose is one of exploration where variables can be easily flexed and implications surfaced. A single persona model is presented in Figure 10. Key persona-specific behavior examined is that of leaving the marketplace when offers to trade that are not matched (i.e. expired). Noting fictional constructs (or objects) and flows between fictional services are used to construct this simulation. Properties from the specific persona behavior can be used in a persona specific system dynamics model, transferred using model.json. The SD model itself allows the designer to explore the population dynamics of specific persona using earlier uncovered trading behavior and constructs. Prosumers, with a specific persona, register with the platforms and then start to trade data. Successful matching is assumed to motivate continued use of the exchange and this is also explored. When used together with performance metrics, it is possible to estimate the horizontal scaling of cloud services as trading and data volumes increase.

Table 3: Fictional Constructs (or objects) emerging from the system dynamic model.

Object	Description	New
Market Size	Buyers and sellers of data (participants)	
Market Growth	Growth in marker participants	
Data Trader Cascading	Growth in new buyers or sellers over time	Y
Data Market Observers	Buyers or sellers who observer the market with interest before entry	Y

6 CONSTRUCT ELICITATION FRAMEWORK

Simulation can be used to provide a more robust and effective approach to design fictions. It is unsurprising that new constructs are uncovered as new artefacts are designed. Software, human or societal perspectives can be synthesized using hybrid simulation for construct elicitation. Construct elicitation can often be regarded as a conceptual exercise, but here we are using a JSON model property file to record constructs, properties and values for wider utilization and concept re-use. Figure 11 attempts to synthesize this hybrid approach to construct elicitation by combining modeling perspectives with model properties. Importantly, sensitivity analysis is used to explore construct dynamics within the larger SD model, directing further agent analysis.

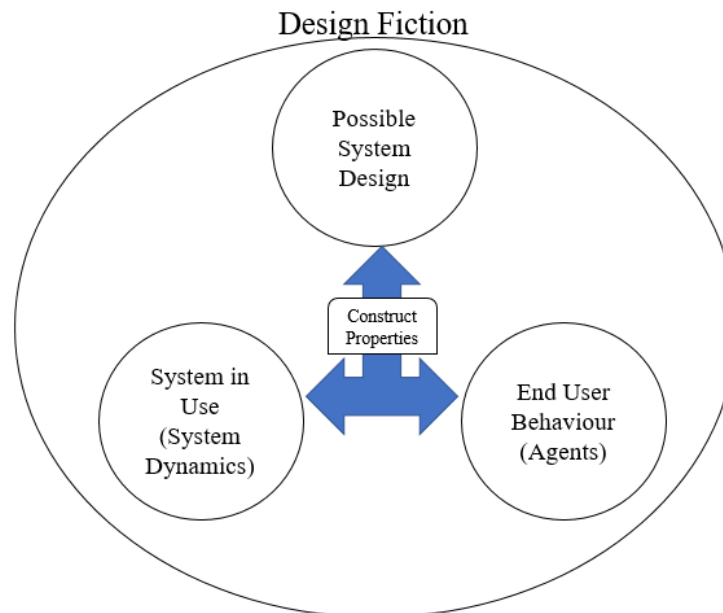


Figure 10. Design Fiction Construct Elicitation.

7 CONCLUSION

Design fictions are speculative futures that motivate discussion and direct further analysis. This paper presents an approach that draws on design science research methods to more fully explore the objects and artefacts within a fictional world. Simulation (Agent-based and System Dynamics in this case) provide a means to explore this fictional world helping to uncover constructs for consideration when designing systems. A personal data trading world is proposed where data futures are traded. Software services are built and tested to further uncover design elements and subsequently situating them within simulations. Agent based modeling is able to uncover individual trading behavior whereas system dynamics is able to demonstrate the mechanics of the trading ecosystem, including marketplace behavior that drives market viability and more practically service scaling. The constructs, models, methods and instantiation are able to support a fictional trading narrative and underpin the simulation models.

A construct elicitation framework emerges from this research that attempts to synthesize software artefacts with simulation models in order to uncover a richer set of future system constructs.

ACKNOWLEDGMENTS

This work was funded in part by the EPSRC Digital Personhood: Digital Prosumer -- Establishing a 'Futures Market' for Digital Personhood Data project (EP/L005859/1). Data trading ideas detailed were formulated by our project team – including Prof. Panos Louvieris, Dr. Natalie Clewley, Mr. Gareth White, Mr. Florian Gamper, Dr. Audrey Guinchard, Prof. Nikos Nomikos and Dr. Federico Colecchia.

REFERENCES

- Anderson, E. F., S. Engel, P. Comminos, and L. McLoughlin. 2008. “The Case for Research in Game Engine Architecture”. In *Proceedings of the 2008 Conference on Future Play Research, Play, Share - Future Play '08*, New York, USA, 228-31.
- Sterling, B. on Design Fictions. Accessed February 9 2017.
http://www.slate.com/blogs/future_tense/2012/03/02/bruce_sterling_on_design_fictions_.html
- Chahal, K. and T. Eldabi. 2008. “Applicability of Hybrid Simulation to Different Modes of Governance in UK Healthcare”. In *Proceedings of 2008 Winter Simulation Conference*, edited by S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler, 1469-1477. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Cooper, A. 2004. *The Inmates are Running the Asylum: Why High-tech Products Drive Us Crazy and How to Restore the Sanity*. 2nd Ed. Indianapolis, Indiana, SAMS.
- Fan, M., J. Stallaert, and A.B. Whinston. 1999. “A Web-based Financial Trading System”. *Computer*. 32(4): 64–70.
- Fujita, S. and Y. Kase. 2016. “Service Market Simulation Based on Service Dominant Logic”. In *Proceeding of IEEE International Conference on Agents (ICA)*, 31–36. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Macal, C.M. and M.J. North. 2009. “Agent-based Modeling and Simulation”. In *Proceedings of the 2007 Winter Simulation Conference*, edited by M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, 86-98. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- March, S.T. and G.F. Smith. 1995. “Design and Natural Science Research on Information Technology”. *Decision Support Systems*. 15(4):251–266.
- Newell, A. and H.A. Simon. 1976. “Computer Science as Empirical Inquiry: Symbols and search”. *Communications of the ACM*, 19(3):113-126.
- Orlikowski, W.J. and C.S. Iacono. 2001. “Research Commentary: Desperately Seeking the “IT” in IT research—A call to theorizing the IT artifact”. *Information Systems Research*. 12(2):121-134.
- Pruitt, J. and J. Grundin. 2003. “Personas : Practice and Theory”. In *Proceedings of the 2003 Conference on Designing for User Experiences*. 1–15. Aarhus, Denmark, The Interaction Design Foundation.
- Simon, H.A. 1996. *The Sciences of the Artificial*. 3rd ed. Rogers Street, Cambridge, Massachusetts, MIT Press.
- Warren, R., D.E. Diller, A. Leung, W. Ferguson, and J.L. Sutton. 2005. “Simulating Scenarios for Research on Culture and Cognition using a Commercial Role-play Game”. In *Proceedings of the 2005 conference on Winter Simulation Conference*, edited by M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, 1109–1117. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

DAVID BELL is a Senior Lecturer in Pervasive Informatics at Brunel University London. He also holds a PhD in Computer Science from Brunel University London. His research interests lie at the intersection of data-driven M&S, human-centred design and Web semantics. His email address is david.bell@brunel.ac.uk.