Digital Business Ecosystem

Contract n^{o} 507953

Workpackage 6

Self organisation

Deliverable 6.4

Intelligence, Learning and Neural Networks in Distributed Agent Systems

Information Society Technologies

Project funded by the European Community under the "Information Society Technology" Programme

Contract Number: 507953
Project Acronym: DBE
Title: Digital Business Ecosystem

Deliverable No: D6.4
Due date: 31/08/2005
Delivery date: 31/08/2005

Short Description:
This report summarises our first three months research into the possible approaches for applying a Neural Networks (NNs) based intelligence to the Multi-Agent System (MAS) model of the Evolutionary Environment (EvE). The ultimate aim of which is to create a *distributed intelligence* capable of optimising the EvE. The approaches considered include applying NNs at the level of evolving populations, individual services, and even the replacement of the evolutionary optimisation with a NNs-based process. The work presented is preliminary, and therefore, the character of the report is partly speculative.

Partners owning: ICL
Partners contributing: ICL
Made available to: Consortium and EC

Versioning		
Version	**Date**	**Author, organisation**
1	31/07/05	Gerard Briscoe, ICL
1	31/07/05	Philippe De Wilde, ICL

Quality Check
1st Internal Reviewer: Paolo Dini - LSE
2nd Internal Reviewer: John Kennedy - INTEL

Intelligence, Learning and Neural Networks in the Distributed Agent Model of the Evolutionary Environment

Gerard Briscoe
Philippe De Wilde

Intelligent Systems and Networks Group
Department of Electrical and Electronic Engineering
Imperial College London

July 2005

Acronyms

BML	Business Modelling Language
DBE	Digital Business Ecosystem
DEC	Distributed Evolutionary Computing
DIS	Distributed Intelligence System
EvE	Evolutionary Environment
ExE	Execution Environment
ICL	Imperial College London
INTEL	Intel Ireland
ISUFI	Istituto Superiore Universitario di Formazione Interdisciplinare
LSE	London School of Economics
MAS	Multi-Agent System
NN	Neural Network
SBVR	Semantics of Business Vocabulary and Business Rules
SDL	Service Description Language
SF	Service Factory
SME	Small & Medium Enterprise
SOA	Service Oriented Architecture
SOLUTA	SOLUTA.NET
STU	Salzburg Technical University
SUN	Sun Microsystems
UBHAM	University of Birmingham

Contents

Acronyms **3**

Executive Summary **5**

1 Introduction **6**

2 Evolutionary Environment Digital Ecosystem **8**
- 2.1 Digital Ecosystem Model . . . 9

3 Approaches for the *Distributed Intelligence* **11**
- 3.1 Accelerate the Evolutionary Optimisation . . . 11
  - 3.1.1 Habitat - Crossover Catalyst . . . 12
  - 3.1.2 Agent - Individual Intelligence . . . 14
  - 3.1.3 Summary . . . 15
- 3.2 Replace the Evolutionary Optimisation . . . 15
  - 3.2.1 Advantages . . . 16
  - 3.2.2 Disadvantages . . . 16
  - 3.2.3 Summary . . . 17
- 3.3 Complement the Evolutionary Optimisation . . . 17
  - 3.3.1 Advantages . . . 18
  - 3.3.2 Disadvantages . . . 19
  - 3.3.3 Summary . . . 19
- 3.4 Directed-Migration of Agents . . . 20
  - 3.4.1 Advantages . . . 21
  - 3.4.2 Disadvantages . . . 22
  - 3.4.3 Summary . . . 22

4 Conclusion **23**
- 4.1 Recommendation . . . 23
- 4.2 Relation to Project Activities, WPs and Tasks . . . 23
- 4.3 Progress and Future Work . . . 23
- 4.4 Summary . . . 25

References **26**

Executive Summary

This report summarises our first three months research into the possible approaches for applying a NNs-based intelligence to the MAS model of the EvE. The ultimate aim of which is to create a *distributed intelligence* capable of optimising the EvE. The approaches considered included applying NNs at the level of evolving populations, individual services, and even the replacement of the evolutionary optimisation with a NNs-based process. The work presented is preliminary, and therefore, the character of the report is partly speculative.

The final model of the EvE is summarised, from deliverables D6.2 [4] and D6.3 [3], to define the context in which the *distributed intelligence* will function and the system that it should optimise. This is necessary as the Distributed Intelligence System (DIS) will be augmentative to the EvE, and will be architecturally specified in task C42 by ICL, and designed and implemented in task C53 by INTEL.

Regarding the objectives to 'investigate the feasibility of neural networks for achieving population clustering' [8] and of whether 'intelligence can optimise the evolutionary process' [8], the *acceleration* approach (of allowing individuals in evolving populations to choose their partners for crossover) and the implications of the results from deliverable D6.2 [4] were considered. It was concluded that an intelligence technique to directly optimise the evolving populations of the EvE would have limited applicability, because the evolving populations of the EvE are currently limited in diversity and complexity relative to biological ones. This has value in defining what approaches should not be used in the creation of a *distributed intelligence*, and therefore the resulting architectural requirements of the DIS for task C42.

Consideration was then given to the *replacement* approach, in which the evolutionary optimisation process would be replaced with a NNs-based process, and the *complementation* approach, in which the evolutionary optimisation would be complemented by agents recombining themselves. Finally, the *directed-migration* approach was considered, in which the agent migration is modified by agents sharing migration information. Both of the later approaches involve embedding NNs-based *individual intelligence* components within the agents to achieve the desired behaviour.

The *replacement* and *complementation* approaches are both viable from a theoretical perspective, but we expect that both will struggle with the NN processing of Semantics of Business Vocabulary and Business Rules (SBVR) information. Also, the *replacement* approach would significantly weaken the ecosystem concept within the EvE, and we consider it essential to maintain the ecosystem concept to meet the wider challenges faced by the EvE and the DBE project. The *directed-migration* approach has the most potential to optimise the EvE, primarily because it can handle the variability of the agent SBVR descriptions constructively. Also, it will not be replacing one technique for another, but will constructively interact with the ecosystem dynamics. So, the objective of 'how software components that are part of genetic selection can be intelligent' [8], will be met by creating a *basic intelligence* through investigating the application of the recommended *directed-migration* approach. This will create a *distributed intelligence* within the EvE which we will investigate to create value for the DBE by providing the core architectural requirements of the DIS for deliverable D6.6 'A high-level design specification of the

distributed intelligence System', required in task C42 at month 26 [8]. This will make it feasible for INTEL to implement the DIS in task C53, and integrate it with the EvE implementation and DBE core architecture with STU, SUN and SOLUTA [7]

The objective of 'how does this *distributed intelligence* interact with the ecosystem dynamics' [8], will be met in a future report in deliverable D6.5 'Report on *distributed intelligence* in evolutionary dynamics' [8], but is briefly introduced in the Conclusion. Developing a unified understanding of intelligence, self-organisation, and complexity, which we believe are intrinsically linked within the DBE, will allow us to investigate their wider implications for the DBE as a whole. This **unified intelligence** will create value for the DBE by providing a greater understanding of intelligence, emergence, and other critical processes operating within the DBE. It will also further scientific enquiries into the structures, requirements, and functions of a Digital Ecosystem.

1 Introduction

This report summarises our first three months research into the possible approaches for applying a NNs-based intelligence to the MAS model of the EvE. The ultimate aim of which is to create a *distributed intelligence* capable of optimising the DBE. NNs provide a learning-based intelligence, to cluster and distinguish different information input, which gives a wide applicability making them suitable for many but not all tasks.

The DBE architecture is made of three distinct environments: the Service Factory (SF), the Execution Environment (ExE), and the EvE [7]. The SF is devoted to service definition and development, including their Business Modelling Language (BML) descriptions [7]. The main opportunity for integrating a form of NNs-based intelligence within the SF comes from the BML SBVR service descriptions. It would be possible to add an intelligence component to each SF instantiation to assist with the knowledge representation process, specifically the writing of the SBVR descriptions of the services. Similar SBVR descriptions could be brought to the user's attention for comparison purposes, to assist in finalising their own SBVR service descriptions. However, this assistance would be limited by the highly structured nature of the SBVR information [12], which makes it difficult to process. It is more suited to being processed by a rule-based system than a NNs-based system. Also, the envisaged form of intelligence would be local and therefore lacking an appreciation of the variability of user-defined SBVR descriptions due to their distribution within the DBE.

The ExE supports the deployment and execution of the infrastructural and user services, implemented as distributed components [7]. The transaction workflow management, for the runtime execution of these services, is the main opportunity for the application of a NNs-based intelligence within the ExE. The transaction workflow management is a very sophisticated problem, with scope for intelligence-based management and optimisation, but its specialised nature requires a specialised solution. The main advantage of a NN-based intelligence is the ability to learn and adapt, but this would be of limited benefit for a transaction flow management system, which requires more of an expert style system with pre-defined transaction handling rules.

Consideration has been given to the possible applications of a NN-based intelligence within the SF and ExE, and limited applicability has been found. This is not unexpected, because both the SF and ExE are architecturally complete, whereas the EvE is considered a platform to provide additional mechanisms for a network-based economy [3], a few such mechanisms were suggested in deliverable D6.3 'How Software Development in the DBE Differs From Normal Business Software Development' [3]. So the EvE by design has more potential for the application of additional mechanisms, and the EvE model consists of a MAS with Distributed Evolutionary Computing (DEC). Therefore, we will investigate the possible approaches to integrate a form of NNs-based intelligence with the MAS model, to achieve a *distributed intelligence* capable of optimising the EvE, using the SBVR service descriptions from the core architecture, service migration (usage) data from the EvE [2], and possibly the SBVR user requests.

The final model of the EvE is summarised from deliverables D6.2 [4] and D6.3 [3] in **Section 2** of this document. The EvE Digital Ecosystem model is based on a MAS with DEC as an optimisation technique. An evolutionary optimisation process, at each Agent Station, will find the optimal combination of agents (services) to meet user-defined requests [2].

Section 3 describes the alternative *distributed intelligence* approaches for augmenting the EvE. Firstly, the possibility of accelerating the evolutionary optimisation process will be considered, by allowing individuals of the evolving populations to choose their partners for crossover. This could be achieved by creating a population-wide NN, termed a Crossover Catalyst, or by integrating a NN within each individual agent, termed an *individual intelligence*. In the *individual intelligence* technique the NNs would be embedded within the agents, so would migrate within the agents to different Habitats.

Considered next is the possibility of replacing the evolutionary optimisation process with a NNs-based process. The modular nature of the EvE makes this possible, and the agent (service) migration mechanisms would remain unchanged.

Considered next is the possibility of complementing the evolutionary optimisation by applying a NNs-based process to the Agent Pool (Service Pool) at each Habitat, to combine agents (services) into potentially useful solutions (applications) outside the evolutionary optimisation process. This would allow the creation of potentially useful applications or partial applications which could bootstrap the evolutionary optimisation processes instantiated to find applications to user requests.

Finally, consideration was given to the possibility of modifying the service migration, so complementing the evolutionary optimisation indirectly. The approach would allow the agents to direct their migration to specific Habitats, rather than just the generally directed migration from the inter-Habitat migration probabilities. This will help to optimise the subset of agents found at the Habitats, which are used in the evolutionary optimisation processes to find applications (combinations of agents) to user requests.

The conclusions are presented and discussed in **Section 4**, including the approach recommended to be investigated further. This includes the value of the contributions both scientifically and to the DBE. Consideration is also given to the relation of this work to the other activities within the project.

2 Evolutionary Environment Digital Ecosystem

The possible approaches for applying a NNs-based intelligence to the EvE will be considered, after the introduction of the EvE which is summarised in this section from deliverables D6.2 [4] and D6.3 [3].

In an old market-based economy, made up of sellers and buyers, the parties exchange property. In a new network-based economy, made up of servers and clients, the parties share access to services and experiences. We view the EvE as a platform to provide mechanisms for a network-based economy by extending the existing **state-of-the-art** in business software development, which can be summarised as Service Oriented Architectures (SOAs). The SOA expresses a software architectural concept that incorporates modular reusable business services that have clearly defined and standardised interfaces.

In a SOA environment, nodes on a network make resources available to other participants in the network, as independent services that the participants access in a standardised way. Most definitions of SOAs identify the use of web services in their implementation. Unlike traditional object-oriented architectures, SOAs comprise loosely joined, highly interoperable application services, and because these services interoperate over different development technologies, the software components become very reusable.

SOAs promise to provide potentially huge numbers of services that programmers can combine, via the standardised interfaces, to create increasingly more sophisticated and distributed applications. The **EvE** will extend this concept by the automatic combining of available and applicable services in a scalable architecture, to meet business user requests for applications.

The services are modelled as agents within the MAS model of the EvE. Individuals within our Digital Ecosystem are ‘combinations of agents’ (applications), created using evolutionary optimisation in response to business user requests for applications. These individuals will migrate through the Digital Ecosystem and adapt to find ‘niches’ where they are useful in fulfilling other business user requests for applications. So the word ‘ecosystem’ will be more than just a metaphor in the EvE.

The creation of the EvE Digital Ecosystem model, which started with the EvE Discussion Paper [5], has been achieved through the interactions of ICL, STU, UBHAM, SUN, SOLUTA, ISUFI and LSE. The final model for the EvE was presented in deliverable D6.2 [4], and more technical information provided in the internal EvE Architecture Requirements document [2]. The EvE model is based on a MAS with DEC. Each business user will have a dedicated node, called a Habitat. At the business user’s Habitat an evolutionary optimisation process will find the optimal combination of agents, from those registered at the Habitat, to meet requests from the business users. Alternative optimisation methods have been examined in deliverable D8.1 [10], and it was concluded that evolutionary optimisation was the most effective in addressing the DBE problem.

2.1 Digital Ecosystem Model

The Digital Ecosystem model contains elements from mobile agent systems, DEC and ecosystem theory. The Digital Ecosystem will consist of interconnected habitats just as in a biological ecosystem. Each Habitat will provide facilities similar to an Agent Station, including an 'Agent Pool' which is the set of agents and 'groups of agents' (applications) registered at the Habitat. The 'Agent Pool' provides a 'gene pool' for Evolving Populations. An Evolving Population, similar to an 'Island' in the DEC Island Model, will be created to use Evolutionary Computing and evolve the fittest (optimal) solution(s) to a request defined by the user. The agents and applications can migrate through the interconnected Habitats combining with one another in the Evolving Populations to meet user requests.

A service in the real business world, such as selling books, is represented by a software service on a computer of the business user. The service is listed on the FADA network via a proxy reference. A Service Manifest describes a service using a combination of BML, Service Description Language (SDL) and BML data, and is stored in the distributed Semantic Registry of the Execution Environment. It also includes a reference to the proxy which activates the software service.

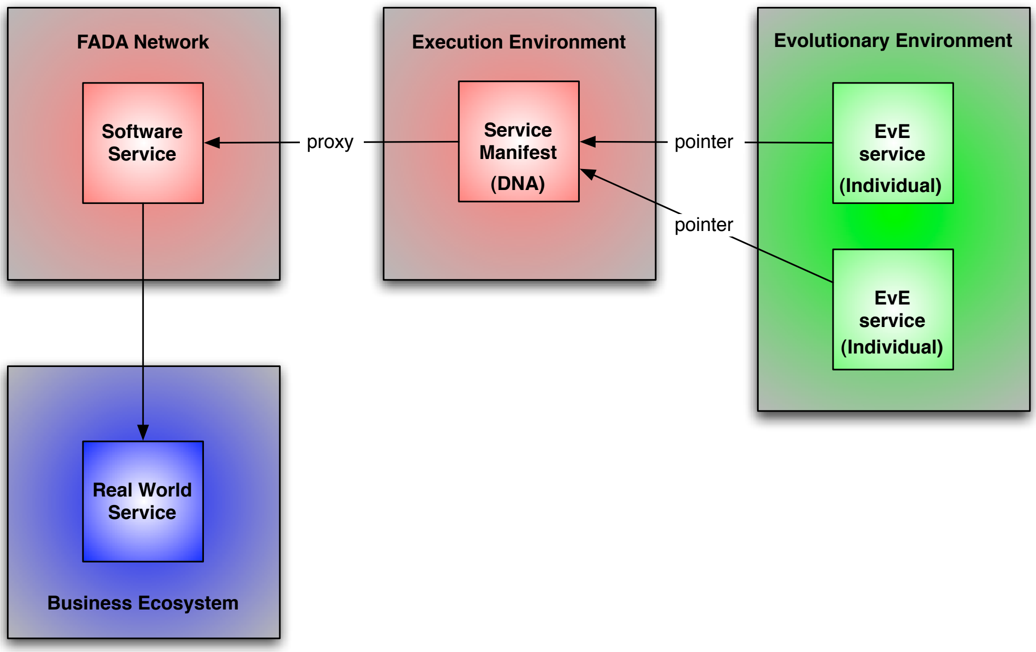

Figure 1: Relation of an EvE service to the DBE service it represents

The individuals (agents) within the Digital Ecosystem model of the EvE are lightweight entities consisting of a description (BML, SDL and BML data) and an executable reference (proxy) to the DBE service they represent. The description contained within each service acts as a guarantee of its functionality, and is the inheritable component from one generation to the next in the evolutionary optimisation.

Each Habitat is localised to the business user it was created for, and therefore the local collection of agents (Agent Pool) within it will have properties which match those of the business user, and so be of potential use to the business user. The temporary evolving

populations of applications (groups of agents), will search the agent combination space through evolutionary optimisation, to find the optimal solution(s) to meet business user requests for applications. These newly evolved applications are also stored within the Habitat's local collection of agents (Agent Pool), as the combinations of agents also have value.

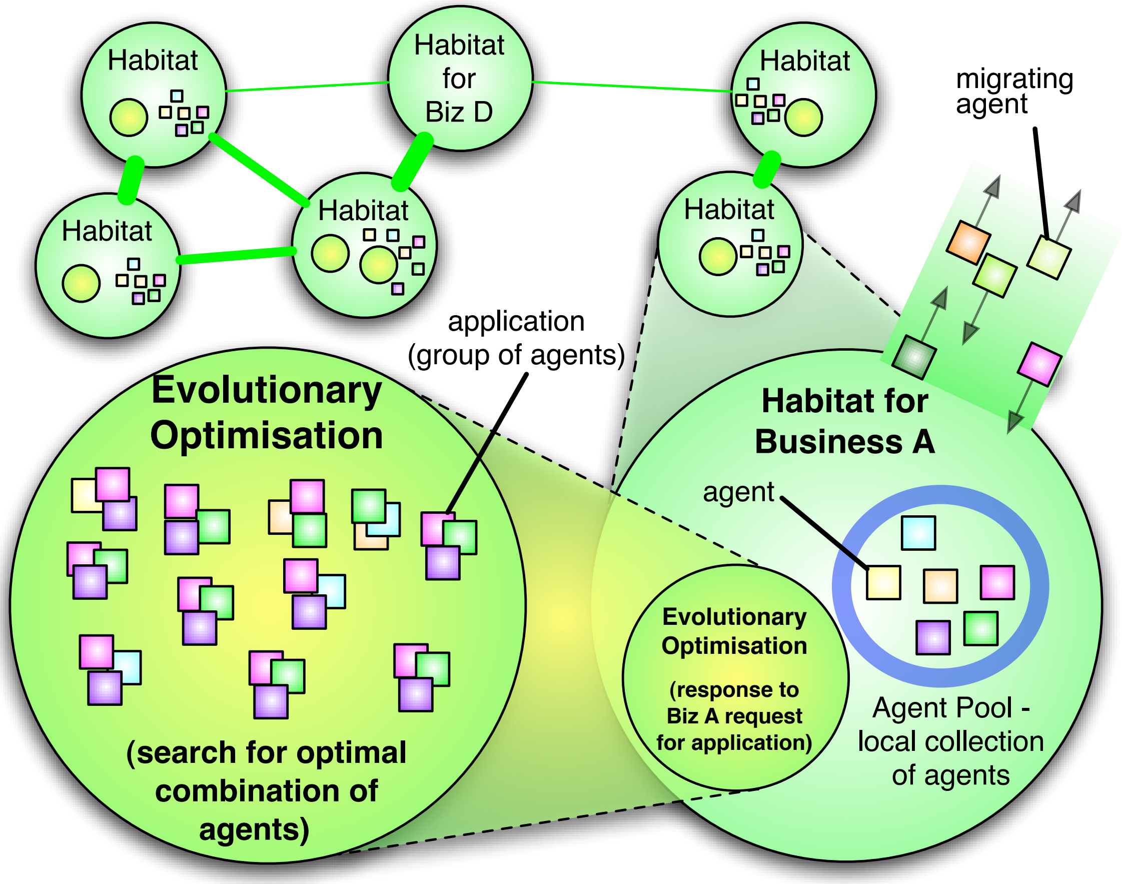

Figure 2: Digital Ecosystem Model

The Habitat also provides a place for agent migration to occur, as the Habitats are interconnected with one another. The migration of an agent within the EvE is initially triggered by deployment to its Home Habitat, for distribution to other business users who will potentially use the agent (service). The successful use of the migrated agent (service), in response to the business user requests for applications, will lead to further migration (distribution) and therefore availability of the agent (service) to other business users. The success of the migration, the **migration feedback**, leads to the reenforcing and creation of migration links between the Habitats, just as the failure of migration leads to the weakening and negating of migration links between the Habitats.

There will be at least hundreds of Habitats, as there is one for each business user, and potentially three or more times the number of populations at any one time. There will then be thousands of agents and applications (groups of agents) available to meet the application requests of the business users within the DBE. The union of the Habitats is the EvE Digital Ecosystem. The connectivity in the Habitats will be parallel to the 'opportunity spaces' (business sector and cross business sector interactions) that exist within the business user base.

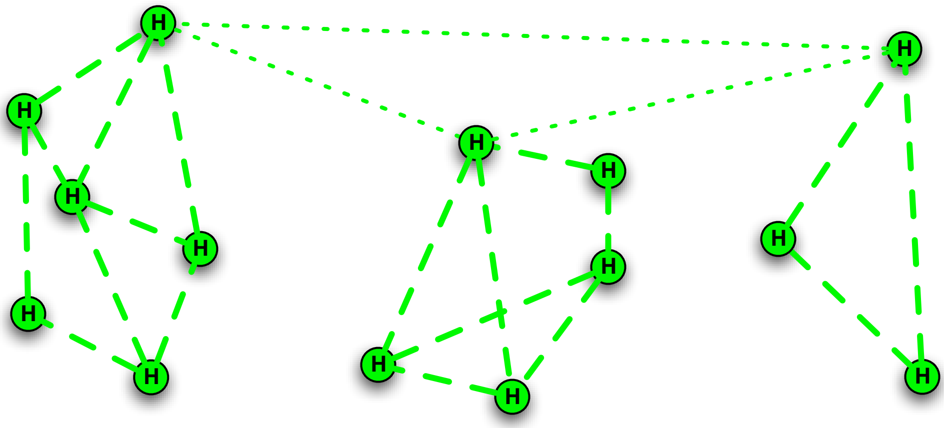

Figure 3: Habitat Network - Caveman Topology - Opportunity Spaces

The target topology for the EvE is a 'caveman' network in which there are 'caves' of high connectivity, equivalent to the strong interactions of opportunity spaces. This is based on the premise of sufficient 'critical mass' in the business user base of the DBE.

3 Approaches for the *Distributed Intelligence*

We intend to use NNs for the following reasons. NNs, which are based on the theory of cognitive science, simulate biological information processing through parallel, highly-interconnected processing elements (neurones) that cooperate to solve specific problems. Neural networks can be 'trained' by example. Generally, NNs are configured for a specific task via a learning process, such as data management. The strength of NNs lie in their ability to derive information from complicated or imprecise data. They can extract patterns and detect trends too complex for a human, or other computer techniques [11]. In addition, NNs can form their own representation of the information they process. Furthermore, tasks can be performed with incomplete or corrupted data, because NNs have a high fault-tolerance [1]. NNs have been used as a learning means for multi-agents information retrieval. In such cases, each agent learns its environment from users' relevance feedback using a neural network mechanism. [6]

The overall effect of the *distributed intelligence* on the EvE (Digital Ecosystem), in biological terms, will be improving the 'succession' process of increasing diversity and complexity. So the relatively stable 'climax community' would be reached within the Habitat clusters sooner.

3.1 Accelerate the Evolutionary Optimisation

This approach aims to accelerate the evolutionary process directly, so that the optimal solutions are found in fewer generations. As each evolutionary process within the EvE would be accelerated, the entire ecosystem would operate more efficiently. This would involve using a NNs-based intelligence approach, to determine the optimal partner for individuals, during crossover within evolving populations. Crossover occurs during the

replication stage of the evolutionary cycle, and involves combining the elements of two applications (groups of agents) into a new application (group of agents).

This approach has similarities to the *clustering*-based self-organisation which was investigated in deliverable D6.2 [4]. However, in D6.2 crossover was encouraged within clusters of the evolving population, where a cluster is a group of similar and identical individuals. It ultimately proved to be ineffective, the reasons for which lead to one of this approach's disadvantages and will be discussed later.

Using a NNs-based process to determine optimal partners for crossover can be done in one of two ways.

3.1.1 Habitat - Crossover Catalyst

The first version of this approach would use a population-wide NN-based component, which we will call a Crossover Catalyst. When an evolving population is instantiated to determine the optimal application (group of agents) to a user request, the Crossover Catalyst would be activated. It would determine, for the individuals of the evolving population, the optimal crossover pairings at each generation. A visualisation of this is shown in Figure 4.

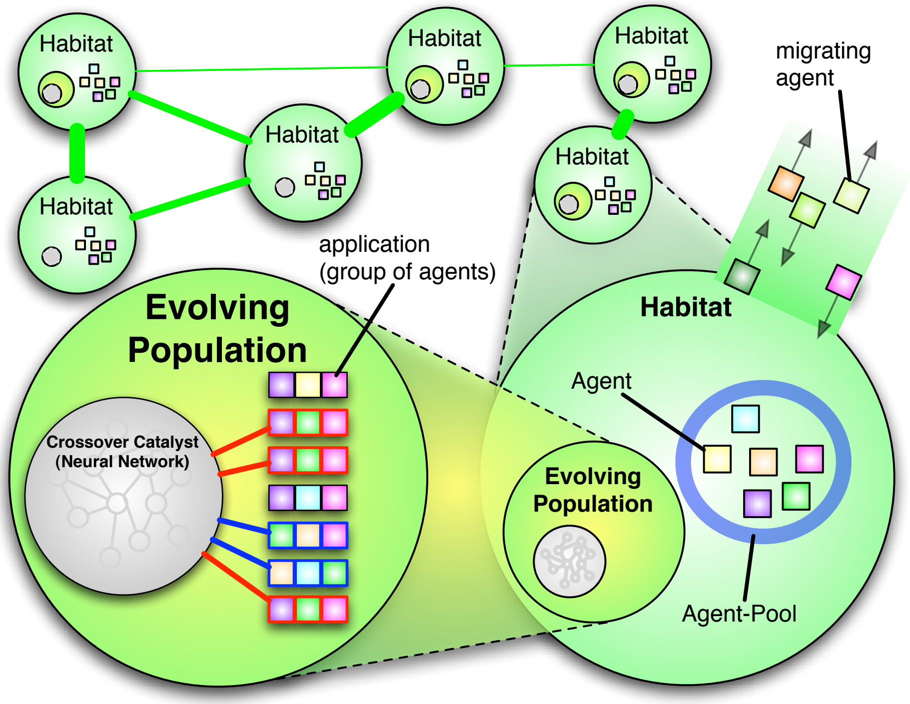

Figure 4: Acceleration: Crossover Catalyst

In this approach each Habitat would have a Crossover Catalyst available to the Evolving Populations instantiated to meet user requests. The NN of the Crossover Catalyst would have to process the entire population, and possibly the user request. Then return a pairing of the individuals (agent-sets) within the population. These individuals would then use this pairing for crossover during the replication stage of the evolutionary process. The property used for determining crossover may be similarity, or diversity. It would be determined by the training set provided to the NN within the Crossover Catalyst.

Advantages
This approach is currently the most developed. This is simply because it was the first approach conceived. The experimentation to integrate a NNs library into the existing simulations was done based upon this approach. Although there are no definitive results, the early impressions are not promising.

Disadvantages
The number of individuals within the population is variable throughout the evolution of the optimal solution, although it is less variable after the initial generations. An ordering of the agents within the individuals (agent-sets) would also be required. The more troublesome issue is the variable size of the agent-sets (individuals) and user requests. NNs require a predefined input layer. So, unless there is very reliable information available regarding the most common agent-set size and request length, then the effectiveness of the NNs would be reduced by significant variation in the size of the agent-set solutions and user requests. Also, an ordering of the properties within the SBVR descriptions would also be required, but that should be less troublesome.

Another concern is the difference in determining the optimal partners for crossover from one evolving population to the next. Potentially the context of evolving populations can be very different. For example determining the optimal application (groups of agents) for an inventory tracking system is very different to a computer aided manufacturing system. A single NNs-based Crossover Catalyst would struggle to effectively manage very different application requests at a single Habitat, significantly reducing its effectiveness.

This approach would struggle to support multiple simultaneous evolving populations, because of the way weightings of a NN are updated based on experience. This is not necessarily a current issue, but depending on the use and resource availability in the future, it could potentially become a limitation depending on whether there is a continuous training mechanism in place.

Another significant concern is the difficulty in achieving a truly *distributed intelligence*. There will be a Crossover Catalyst at each Habitat working independently to optimise the evolving populations instantiated to find applications (groups of agents) to user requests. Working almost completely independently is an extremely poor form of *distributed intelligence*. The only indirect interaction will be the migration of services from one Habitat to another, in which agents will be migrated sooner and alternative variations of agent groups would also be migrated. So the 'additional' interaction effect would be very negligible. Creating connections between the Crossover Catalysts would be akin to solving the problem of how neural clusters in the brain are interconnected, though they are responsible for different activities, such as vision, movement, etc. Although it is a very interesting problem, the challenge of answering it would be impractical within the timeframe of the project. We consider it critically important that there be significant, preferably direct interaction between the NNs-based components to achieve true *distributed intelligence* in more than just name, but it does not need to be interaction between the NNs themselves.

3.1.2 Agent - Individual Intelligence

This version of the approach would involve embedding a NNs-based *individual intelligence* component within each agent. This component would remain dormant within the agent upon migration and registry in the Agent Pool of a Habitat, but would become active during the agent's use in an evolving population to determine the optimal partner for crossover at each generation. This is visualised in Figure 6.

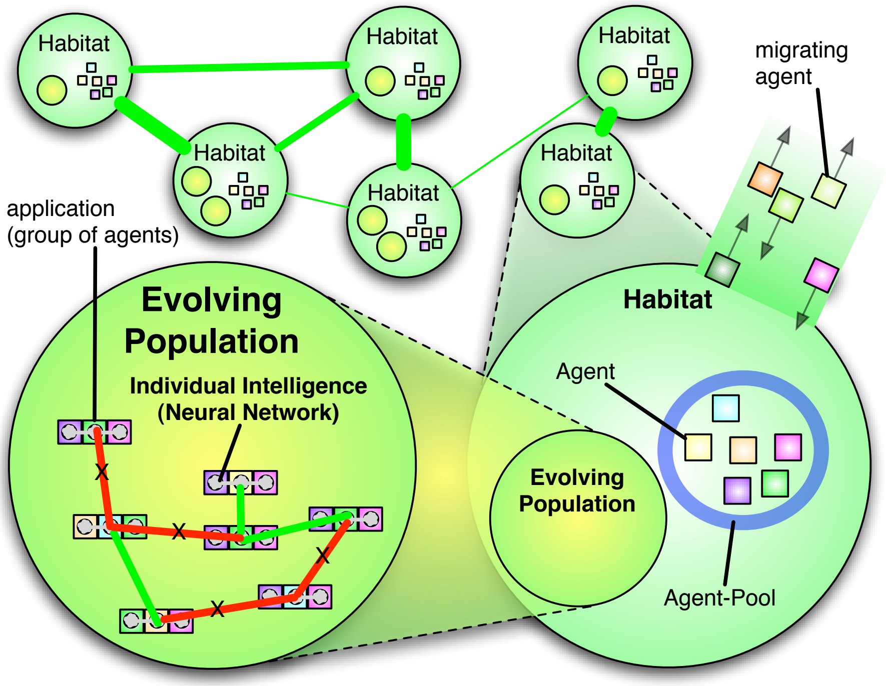

Figure 5: Acceleration: Individual Intelligence

When two agent-sets assess each other, to determine whether to perform crossover together, their agents embedded *individual intelligence* components will have to assess their counterparts SBVR descriptions. The simplest way to achieve this would be to pair the agents from the two agent-sets as much as possible. A full pairing may not be possible as the agent-sets may be of a different size. The property used for determining crossover would be determined by the training sets provided, like the previous approach.

Advantages
This version of the approach will be capable of providing a truly *distributed intelligence*, unlike the Crossover Catalyst version. The NNs-based *individual intelligence* components will be embedded within the interacting agents and so will migrate within the agents to interact with different agents at different Habitats.

This version of the approach would also lead to the phenomenon of interacting intelligent components, as the agents with the embedded intelligence must interact to function. This is considered valuable as it could potentially lead to emergent behaviour.

This approach will make the agents even more like software agents, by giving them a form of intelligence and direct control over their behaviour. This combined with the migration

will make the agents within the EvE similar to agents in a mobile agent system. A very weak generic definition of agent has been used thus far, albeit a fundamentally correct one. This approach would address this issue. However, admittedly the control provided is extremely limited to a single function within a specific process.

Disadvantages
A significant challenge will be to manage the interaction of the NNs-based *individual intelligence* components embedded within the agents, when the agents themselves are aggregated to form applications. In this approach the agents within an application (group of agents), each with an embedded NN, would have to collaborate with one another to determine their optimal partner for crossover. An ordering of the agents within the agent-set of individuals would be required, and an ordering of the properties within the SBVR descriptions would also be required,

Another challenge will be creating a scalable interaction between the agents to determine the optimal partners for crossover within an evolving population. Firstly, it will probably not be practical to interact with every other individual of the population. Secondly, there is no metric space to implement a neighbourhood model which is commonly used in determining the scope (range) of interaction within a population. So consideration will have to be given to determining suitable interaction rules for agents of an evolving population.

3.1.3 Summary

This approach, in biological terms, would be equivalent to a creating more specialised species within the ecosystem. The speed of the 'succession' process would be moderately increased, so the relatively stable 'climax communities' would be reached within the Habitat clusters sooner. There might be a minor loss in diversity, but this would be unlikely to affect the users.

This approach has similarity to the *clustering*-based optimisation from deliverable D6.2 [4]. Both involve altering the crossover operator, albeit very differently. Yet, the issue of insufficient population diversity that affected the *clustering* will also affect this approach. In deliverable D6.2 [4] it was surmised that individuals within the evolving population lacked sufficient complexity (relative to biological evolving populations) for the *clustering* mechanism to be effective. So although the Crossover Catalyst may determine the optimal crossovers to perform, it is doubtful it will show a significant difference to a random crossover operator. So, again, although the technique may be effective generally in evolutionary computing, it will not be effective in the specific case of the EvE.

We would expect that the envisaged NNs-based components would learn to be optimal crossover operators, which in our current simulations would be equivalent to random crossover operators. It would however provide an adaptable technique, always providing the optimal crossover operator in more sophisticated, future versions of the EvE.

3.2 Replace the Evolutionary Optimisation

In this approach the evolutionary optimisation process used to find the optimal applications (groups of agents) to user requests, would be replaced by a NNs-based

process. The modular nature of the EvE makes this possible. The agent migration between Habitats will remain, or else an alternative mechanism would be required to create the user-specific collections of applicable agents, the Agent Pool, from which the applications are constructed.

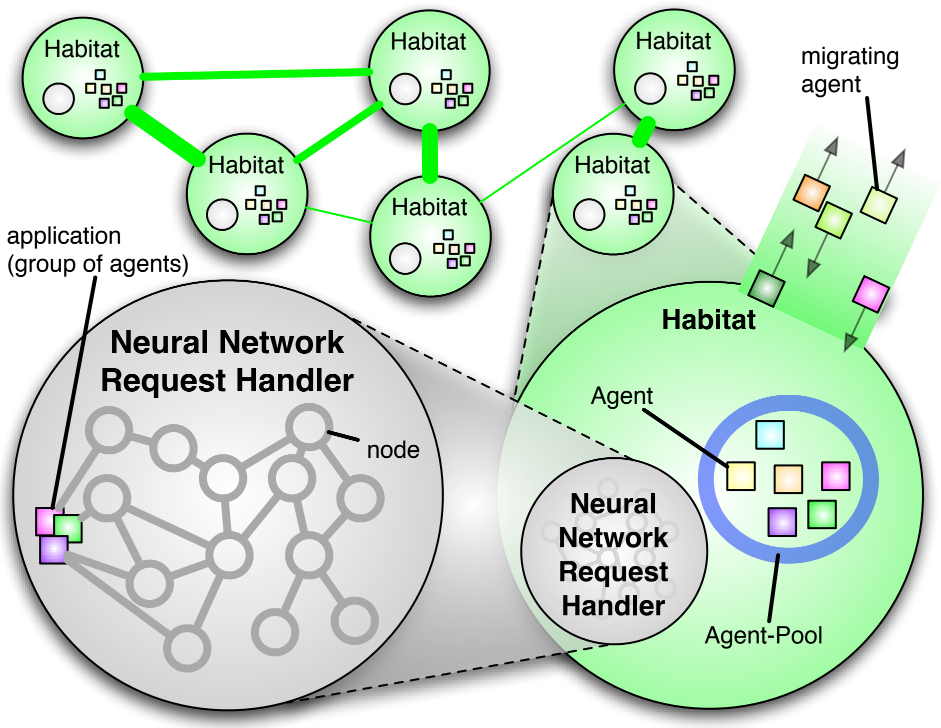

Figure 6: Replacement: Neural Network Request Handler

The NNs-based Request Handler will have to process a request and the agents within the Agent Pool, to determine the optimal subset of agents which most closely match the user request. An ordering of the agents within the Agent Pool will most likely be required, as well as an ordering of the properties within the business process units defined in the SBVR user request. This would be a very significant pre-processing step, and if mishandled might inadvertently prevent a solution from being found.

3.2.1 Advantages

This approach would provide an alternative mechanism for determining the optimal applications (groups of agents) to user requests, and its operational efficiency would increase over time as the NN learns, acquiring experience.

3.2.2 Disadvantages

This approach would also struggle to achieve a truly *distributed intelligence*. There will be a single NN Request Handler at each Habitat, working independently to find applications (groups of agents) to user requests. Working independently is an extremely poor form of *distributed intelligence*, and creating connections between the NN Request Handlers would represent a very significant challenge, impractical within the timeframe of the project.

This approach would also struggle to manage the variable size of user requests, the variable number of agents available in the Agent Pool, and the variable size of application (agent-set) responses. NNs require an input layer which is predefined. So, unless there is very reliable information available regarding the most common agent-set size, Agent Pool size, and request length, then the effectiveness of the NNs would be reduced. Also, an ordering of the properties within the SBVR descriptions would also be required.

It would not be possible to support simultaneous handling of multiple user requests, because of the way the weightings of a NN are updated based on experience. This would very much depend on whether there is a continuous training mechanism in place.

Replacing the evolutionary mechanism of the EvE with a NNs-based process, would weaken the Digital Ecosystem concept within the EvE. It makes the EvE much closer to a traditional MAS with mobile agent migration, and raises the issue of using a more traditional MAS approach with less biological inspiration. The current, biologically inspired, agent migration rules are more sophisticated than a traditional mobile agent system. We view replacing the current agent migration rules with an alternative technique for the creation of business user-specific collections of agents as extremely difficult and impractical at this stage of the project. Therefore, this approach would create inconsistencies in the EvE model.

3.2.3 Summary

We would expect that the NNs-based Request Handler could probably provide an alternative to the evolutionary optimisation. However, we would not be investigating how a *distributed intelligence* interacts with the ecosystem dynamics, but instead how it can replace a fundamental process of the ecosystem dynamics.

3.3 Complement the Evolutionary Optimisation

This approach involves endowing the agents with intelligence, using NNs-based *individual intelligence* components. However in this approach the agents use their intelligence to bind with one another outside the evolutionary optimisation, in the Agent Pool. The *individual intelligence* components have the same name and appearance (in the figures) as the *acceleration* approach, but function completely differently. This *distributed intelligence* would allow for the creation of potentially useful applications (groups of agents) or partially complete applications, inside the Agent Pool. This has the potential to bootstrap the evolutionary optimisation process, which responds to user requests for applications, because it is seeded from the Agent Pool.

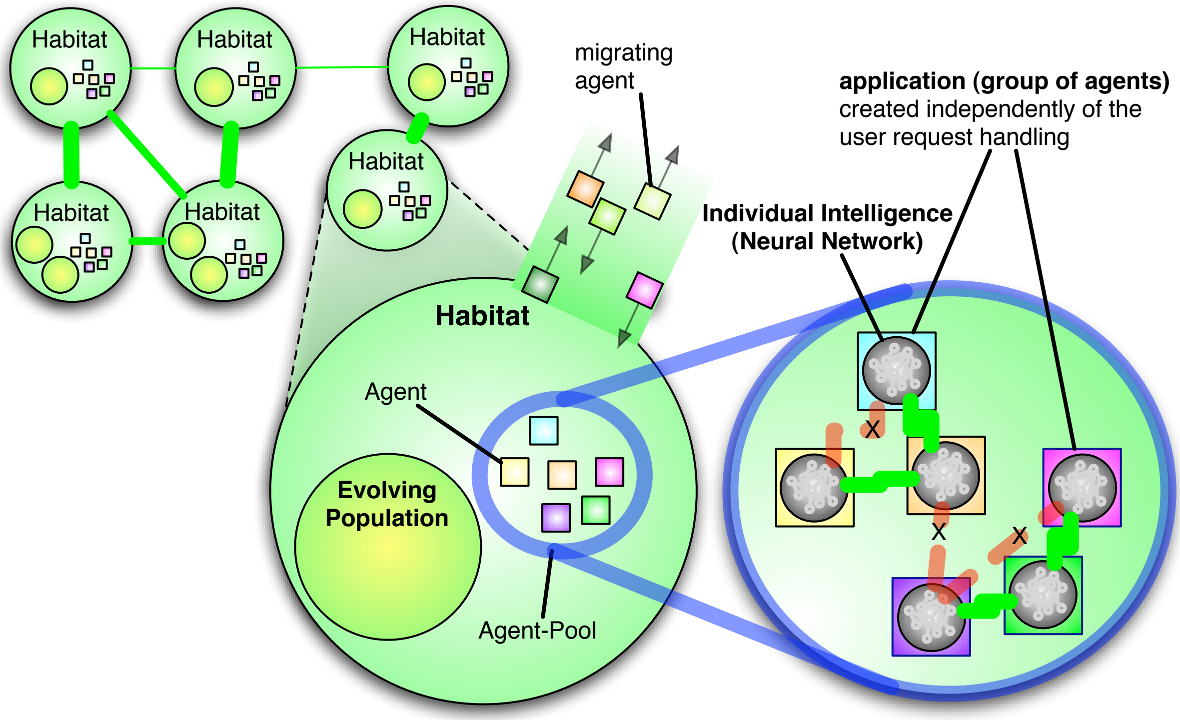

Figure 7: Complement: Individual Intelligence

If joining two agent-sets, then ideally each agent within the set needs to express a view about the other agent-set, specifically its agent's SBVR descriptions. For the agents to make an informed decision it must have an understanding of the context in which the new agent-set would operate, specifically past user requests. Only a simple response of join or not is required from each agent within the agent-sets, but the evaluation process would have to be quite sophisticated to assess other agent SBVR descriptions relative to their own, within the context of the current Habitat's user requests.

3.3.1 Advantages

This approach grants the agents at least some control over their behaviour, admittedly in a limited form, but still more than the previous approaches. The *acceleration* approach would also give the agents some self-control, but only for a single function within the evolutionary optimisation.

This approach, more so than the others, will achieve the interaction of a *distributed intelligence* with the ecosystem dynamics. The other approaches do not so much interact with the ecosystem dynamics, but are embedded subordinately or attempt partially to replace them.

This approach would also strengthen the 'agent' definition because of the NNs-based *individual intelligence* components embedded within the agents. It would also provide a truly *distributed intelligence*, because these agents would migrate to different Habitats. It would also create the phenomenon of interacting intelligent components, and therefore could potentially lead to emergent behaviour.

3.3.2 Disadvantages

The Agent Pool would cease to be a subset of the power-set of agents registered at the Habitat (Agent Station), because some duplication within the Agent Pool will be required. This is not considered problematic, because in the architectural specifications [2] and expected implementation the Agent Pool will consist of pointers to agents. So the increase in pointers will have a near inconsequential effect on the storage requirements of the EvE.

This approach will lead to an increase in the processing resources required. We will need to provide runtime to the agents within the Agent Pool in order for them to be able to collaborate. Managing the interaction of the NNs-based *individual intelligence* components embedded within the agents, when they are aggregated to form applications, would add to the ever increasing overhead that this approach would place upon the infrastructure. In this approach two agent-sets, each with their own embedded NNs, would have to collaborate with one another to determine whether to join or not. Some type of voting system would be required, else a dominant agent within the agent-set would have to be given decision making privileges. Both approaches have their own merits, and there are definitely others. The splitting of agent-sets would require a similar system and we believe all these sub-systems would place an excessive overhead upon the infrastructure. So, a significant increase in processing resources would be required, which may exceed the potential benefits.

The agents within an agent-set will have to interact when assessing potential partners (other agent-sets). So, an ordering of the agents within the agent-set and the properties within their SBVR descriptions would be required. Although the Agent Pool is nothing in size compared to an Evolving Population, it will probably not be practical for each individual to interact with every other individual in the Agent Pool, and there is no metric space to implement a neighbourhood model. So, a scalable strategy for intra-agent-set and inter-agent-set interaction will be required.

3.3.3 Summary

In biological terms, this approach would be equivalent to increasing the recombination that occurs within the ecosystem. Provided that excessive recombination does not occur, then the 'succession' process should moderately increase. So, the Habitat clusters would reach their 'climax communities' sooner, and they would probably be able to adapt to change faster and more easily.

This approach would create a *distributed intelligence* that interacts with the ecosystem dynamics of the EvE. It would also complete the MAS component of our Digital Ecosystem by giving the individuals (agents) some intelligence and control over their existence, but the problem of scalable interaction and variation in length of SBVR descriptions may prove insurmountable.

3.4 Directed-Migration of Agents

This approach will work to complement the evolutionary optimisation indirectly. As the EvE increases in size (number of users), the total number of agents (services) available globally will become increasingly large. For the evolutionary optimisation to continue to work efficiently as the EvE expands, optimal subsets are required at the Habitats of the super-set of agents available globally. The migration probabilities between the Habitats work to achieve this in a 'passive' manner. Although the mechanism is effective, the migration of agents will be slow and not strongly directed. It allows the agents, based primarily upon success at their current location, to spread in the correct general direction within the Habitat network. This approach will work in a more 'active' manner allowing the agents highly directed migration to specific Habitats, rather than generally directed migration. This will help to optimise the subset of agents found at the Habitats, which are then used in evolutionary optimisation processes to find applications (combinations of agents) to user requests.

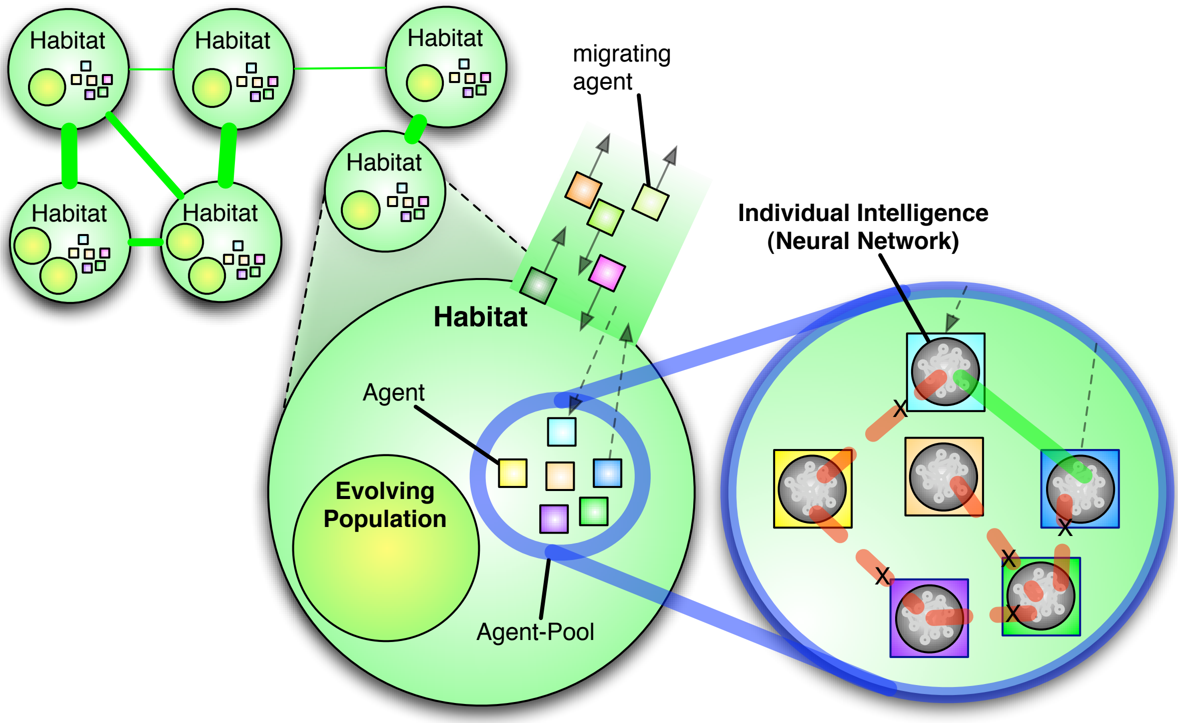

Figure 8: Complement: Service Migration

The agents can interact with one another, using their embedded NNs-based *individual intelligence* components, to determine if other agents are functionally similar based on their SBVR descriptions. This simpler approach, relative to the others, involves comparing agent SBVR descriptions to one another for similarity instead of compatibility, without the need of the current Habitat's context (user requests). If similar the agents can compare their 'migration and usage' histories [2] and determine Habitats where they could be valuable. This interaction would occur outside the evolutionary optimisation, in the Agent Pool.

None of the other approaches would have a significant effect on the agent life-cycle, because the recombination or migration stages basically remain in the same order. This approach

does significantly change the agent's life-cycle, shown in the lighter shade (blue) of Figure 9. Basically, where and when agent migration will occur will change. There will be more opportunities for agent migration, but more importantly these opportunities are for targeted migration.

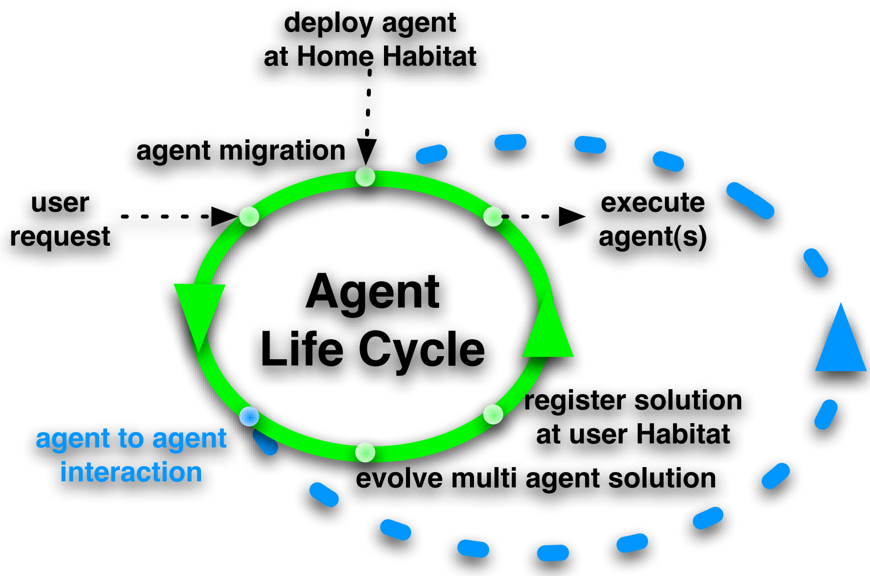

Figure 9: New Agent Life-Cycle

3.4.1 Advantages

We choose to place the intelligence within the agents to maintain the consistency of the MAS and Digital Ecosystem models of the EvE. There are other ways in which this behaviour could be achieved. For example, the Habitats could communicate more and share agent (service) usage information more directly, but this is not a behaviour found within a biological ecosystem or often within MASs. Whereas behaviour in which agents communicate and share information about suitable Habitats is much closer to biological ecosystems and known within MASs.

This approach would require less processing resources than the other approaches, because although there will be roughly the same number of inter-agent interactions per unit time, each interaction will process significantly less data. So, relative to the other approaches the NNs will have smaller input layers, and probably less hidden layers.

Automated training mechanisms should be possible and there are several to consider. This is a significant advantage over the other approaches, which have none. Firstly, as the agent needs to be trained to recognise similarity to itself, the first training set will consist of its own SBVR description. At the start of an agent's life-cycle its NN's pattern recognition capabilities would be limited, but still functioning. Agents could then train their NN based on experience. If visiting a Habitat based on an inter-agent interaction, whether it proves to be successful or not, it can be added to the training set. Additionally, untrained inexperienced agents could copy the weightings of a NN from a more experienced agent, when similarity is determined during an inter-agent interaction.

Variation in the size of user requests and agent-sets (responses) is not problematic for this approach. It will also have the least trouble with the processing of SBVR descriptions, and therefore has a substantial advantage over the previous approaches. NNs generally require strict limitations on the content and format of the information to be processed. Often, pre-processing is required to prepare the data for the input layer of a NN. Although we cannot ignore such concerns, we cannot impose such stringent requirements on the SBVR descriptions of the agents, as the EvE must cater to a large range of businesses and their services. The input layers of the NNs will be sized proportionally to the SBVR descriptions of the agents that the NNs represent. More importantly, the possible shortfalls in the size of the input layer when comparing other agents' SBVR descriptions is masked by the comparison for similarity. So, this approach will minimise the consequences of straying from a strict information coding regime, which is not possible with the agent SBVR descriptions.

This approach would also strengthen the 'agent' definition, because NNs-based *individual intelligence* components are used.

3.4.2 Disadvantages

A more sophisticated simulation will be required to validate this approach, compared to the other approaches, because it will lead to a global emergent effect over all the Habitats.

An ordering of the properties within the agent SBVR descriptions will be required. However, this is true of all the approaches.

Again, it will probably not be practical to allow each individual to interact with every other individual in the Agent Pool, and there is no metric space to implement a neighbourhood model. So, an alternative interaction strategy will be required.

The most significant challenge will be to manage the migration path of an agent-set, as each member agent will have their own optimal migration path. The differences are beneficial to the desired behaviour, because they will ensure a balanced migration path. However, the way in which conflicts in the migration paths are resolved will require finesse, to ensure optimal use of the information attained from the agent to agent interactions.

3.4.3 Summary

In biological terms, this approach would be equivalent to providing each individual with a relatively sophisticated social interaction mechanism. It should therefore significantly improve the 'succession' process by allowing niches to be fulfilled faster, so the Habitat clusters would reach their 'climax communities' sooner. Also, communities will be able to adapt faster to changing conditions.

This approach would create a *distributed intelligence* that interacts with the ecosystem dynamics of the EvE, and of all the approaches would have the greatest potential to optimise the EvE. It would also complete the MAS component of our Digital Ecosystem, by giving the individuals (agents) some intelligence and control over their existence.

4 Conclusion

4.1 Recommendation

The *acceleration* approach involves attempting to optimise the crossover operator similarly to deliverable D6.2 [4], and suffers from the same limitation of minimal effectiveness when applied specifically to the EvE.

The *replacement* and *complementation* approaches are both viable from a theoretical perspective, but from an implementation perspective we expect that both will struggle with their NN processing of SBVR information. The variability inherent within the SBVR descriptions is too high for straightforward NN processing. Furthermore, the *replacement* approach would significantly weaken the ecosystem concept within the EvE. We consider it essential to maintain the ecosystem concept to meet the wider challenges faced by the EvE and the DBE project.

The *directed-migration* approach has the greatest potential to optimise the EvE, primarily because it can constructively handle the variability of the agent SBVR descriptions. Also, it will not be replacing one technique for another, but will constructively interact with the ecosystem dynamics. So, we recommend adopting the *directed-migration* approach to pursue a *distributed intelligence* for the EvE.

4.2 Relation to Project Activities, WPs and Tasks

We will interact with task C42, High-Level Design of the DIS and EvE Architecture Integration, in which the desired behaviour and fundamental objects required for the DIS will be defined. This computing task is being done primarily by the lead author of this report, which will ensure synchronisation with the science and computing streams to make sure that the scientific outputs are channelled appropriately into the existing high-level design of the EvE. It will also discuss more practical issues, such as an automated training mechanism for the backpropagation of embedded feed-forward NNs. The SBVR distance function to be used within the EvE [9] could be adapted for initial training sessions, and/or a continuous training regime could be constructed based on the success of the decisions made by the NNs.

We will develop a *basic intelligence* based on this deliverable, so as to lead to an architecture specification, deliverable D6.6 'A high-level design specification of the *distributed intelligence* System', required in task C42 at month 26 [8]. This will make it feasible for INTEL to implement the DIS in task C53, and integrate it with the EvE implementation and DBE core architecture with SUN, STU and SOLUTA [7]. This process of interaction will be started shortly, similarly to the creation of the EvE, with the creation of discussion paper on the DIS.

4.3 Progress and Future Work

Regarding the objectives to 'investigate the feasibility of neural networks for achieving population clustering' [8] and of whether 'intelligence can optimise the evolutionary process' [8], the implications of the results from deliverable D6.2 [4] were considered. An intelligence technique to directly optimise biological evolution would have limited

applicability to the evolving populations of the EvE, because they are currently limited in diversity and complexity relative to biological evolving populations. So, such a technique would have limited applicability to the EvE.

The objective of 'how software components that are part of genetic selection can be intelligent' [8], will be met by investigating the application of the recommended *directed-migration* approach for the *basic intelligence*. Complementing the EvE, by the embedding of NNs-based *individual intelligence* components within the agents, will create a *distributed intelligence* within the EvE. We will investigate this *distributed intelligence* to meet this objective, which will include extending our existing simulations to validate the recommended approach. This will allow us to create value for the DBE in a framework for the architecture of a Distributed Intelligence System (DIS) to augment the EvE. The *distributed intelligence* will endow the agents with a degree of intelligence and control over their existence, with the aim of optimising the migration and distribution of agents within the EvE. The pattern recognition capabilities of NNs will be leveraged to allow agents to determine similarity based on their descriptions. We currently intend to use multilayer feed-forward NNs, using the backpropagation training technique to specify the required pattern recognition behaviour. The optimal NN structure, such as the neurone connectedness and the transfer function, will be determined through a combination of theoretical reasoning and explorative programming. The immediate future research will be conducted as follows:

- Complete the theoretical model to proceed with our simulations, leading to the primary requirements for the embedded NNs.
- Completion of the *distributed intelligence* model, excluding the embedded NNs. Access to the appropriate information and control mechanisms will be defined, etc.
- Construct a simulation, or extend our existing simulation, to verify experimentally our chosen *distributed intelligence* approach. Then we will experiment with varying the NN properties for the greatest optimisation of our EvE simulation.
- Construction of an architecture for the DIS, based on the research, but also including the consideration of more practical implementation issues.

The objective of 'how does this *distributed intelligence* interact with the ecosystem dynamics' [8], will be met by developing a unified understanding of intelligence, self-organisation, and complexity, which we believe are intrinsically linked within the DBE. We will aim to develop a greater understanding of the system, instead of producing algorithms for components of the DBE architecture. It is not yet clear how we will approach this challenge, as we are currently focussed on the *basic intelligence*, but we will briefly elucidate some of our initial thoughts:

- With the actual complex system of the DBE, will the local interactions from the critical mass of SME users and their services lead to the emergence of global intelligent behaviour, and if so what type of behaviour?
- For example, clustering-based organisation has proven to be important within the DBE at several different levels, as it achieves an organisation of the complexity generated within the ecosystem. Does enforcing clustering within the lower levels of the DBE have a global effect, and if so what type of effect?

- The wider indirect implications of how the *distributed intelligence* will interact with the ecosystem dynamics will be investigated, and will help to develop a greater understanding of the system. With the actual complex system of the DBE, will the local interactions from the users and their agents (services) lead to the emergence of global intelligent behaviour, and if so what type of behaviour? Does endowing the services with even a limited intelligence create unexpected chaotic global behaviour? And under what conditions would it occur?

- Currently the interaction between the *distributed intelligence* and the evolutionary dynamics is immune to the Baldwin effect [13]. If instead we made the individual intelligence components subject to the evolutionary optimisation, what behaviour would be observed? Although, considerable effort would be required to determine a viable mechanism, not only for crossing NNs, but potentially differently sized NNs.

We will investigate this **unified intelligence** to meet the objective, and create value for the DBE by providing a greater understanding, and the wider implications for the DBE as a whole, of intelligence, emergence, and other critical processes that will operate within the DBE. This work will also further scientific enquiries into the structures, requirements, and functions of a Digital Ecosystem.

4.4 Summary

Overall, there has been substantial progress in the first three months to meeting the objectives. Our recommended approach for the *distributed intelligence* of the EvE will endow the agents (services) with a degree of intelligence and control over their existence, with the aim of investigating how this can optimise the EvE. It will also be able to manage the variability in the SBVR descriptions of an agent, which is of critical importance if this research is to be transferable to the implementation of the DIS.

References

[1] A Becks, S Sklorz, and M Jarke. A modular approach for exploring the semantic structure of technical doument collections. In *Working Conference on Advanced Visual Interfaces*, May 2000.

[2] G Briscoe. Evolutionary Environment Architecture Requirements. *Internal*, April 2005. Available from: `http://www.iis.ee.ic.ac.uk/~g.briscoe/Eve/EveArchRequirements.pdf`.

[3] G Briscoe and P De Wilde. How software development in the DBE differs from normal business software development. *Digital Business Ecosystem*, Contract no 507953, 2004.

[4] G Briscoe and P De Wilde. Control of self-organisation and a performance measure. *Digital Business Ecosystem*, Contract no 507953, 2005.

[5] G Brisoce, J Rowe, and P Dini. Evolutionary Environment Discussion Paper. *Internal*, 2004.

[6] Y S Choi and Y S Yoo. multi-agent learning approach to WWW information retrieval using neural network. In *7th World Wide Web Conference*, Brisbane, Australia, April 1998.

[7] P Ferronato. D21.2 architecture scope document. *Internal*, 2005. Available from: `https://dbe.digital-ecosystem.net/files/documents/8/483/file_483.dat?filename=Del%5f%32%31%2e%32%5fDBE%20Core%20Architecture%20Scoping%20Document%5fRelease%5fA%2epdf`.

[8] M Giorgetti, P Dini, and A Nicolai. Deliverable D1.2, Detailed Work-Plan for the second phase. *Internal*, WP6 description, 2005.

[9] T Heistracher, C Masuch, T Kurz, and G Marcon. Towards a fitness function for matching sbvr with genetic algorithms optimization. *Internal Report*, 2005.

[10] Jonathan E. Rowe and Boris Mitavskiy. D8.1 report on evolution of high-level software components. *Internal*, 2005.

[11] N E Sondak and V K Sondak. Neural networks and artificial intelligence. In *Twentieth SIGCSE Technical Symposium on Computer Science Education 21, 1*, February 1998.

[12] Maurizio De Tommasi. Business modelling language 1.0. *Internal*, 2005. Available from: `https://www.digital-ecosystem.net/files/documents/8/561/file_561.dat?filename=Del%5f%31%35%2e%31%5fDBE%5fBusiness%20Modelling%20Language%20%31%2e%30%2epdf`.

[13] P Turney. Myths and legends of the baldwin effect. In *Proceedings of the 13th International Conference on Machine Learning*, pages 135–142, 1996.