Digital Business Ecosystem

Contract n° 507953

WP4: Science Coordination

D4.2: Summary of DBE Science Achievements

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 N°: D4.1

Due date: 31/01/2007

Delivery Date: 9/03/2007

Short Description:

This report summarises the results of the Science Domain research relative to their contribution to the overall project's objectives. The development of the Evolutionary Environment is seen as the central achievement, even if its functionality was only partially completed.

Partners owning: LSE (Paolo Dini)

Partners contributed: STU (Thomas Heistracher, Thomas Kurz), HWU (Philippe De Wilde, Gerard Briscoe), ICL (Maria Petrou), UBham (Jon Rowe)

Made available to: unlimited distribution

Quality check

Internal Review: John Kennedy(INTEL), Elmar Husmann (IBM)

Summary of DBE Science Achievements

Table of Contents

Greater role of Social Science 3
Integration into the DBE Computing infrastructure 4
Summary of achievements by individual partners 5
ICL 5
UBHAM 6
HWU 7
STU 8
LSE 9
Assessment of stated Science objectives 10

Greater role of Social Science

It is interesting at the end of the project to look back at how the project's research areas were conceptualised at the beginning of the project. Fig. 1 shows that Social Science became increasingly important during the course of the project, culminating with the Delphi discussion on project governance.

The other arrows still seem largely relevant, but have acquired a much more concrete meaning. Also, where the project fell short is more evident.

Science – Memory – Computing. This arrow was realised through the Distributed Intelligent System (HWU). The other interpretation of memory through DNA, however, has not been realised yet (UBHAM, STU, ISUFI). During the course of the project the potential contribution of Science to Computing through gene expression has slowly come into focus, but it has not been realised yet either (LSE).

Science – Evolution – DBE. This arrow has been realised by half. The Habitat service was implemented and integrated with the ExE (STU, SUN) and the Studio (UCE) and the DIS was integrated into the EvE (HWU, INTEL), but the genetic algorithms acting on the SBVR models has not been realised (UBHAM, STU, ISUFI). This work is being addressed in the OPAALS project.

Science – Networks – Business. This arrow has been partially realised through the development of several simulations (STU, ICL, HWU).

Science – Social Networks – Social Science. This new arrow represents a fruitful collaboration mainly between the STU partner and the Social Science partners (LSE, CENSIS) through the EvE Simulator framework, which allows the visualisation of social networks of SMEs, among other things.

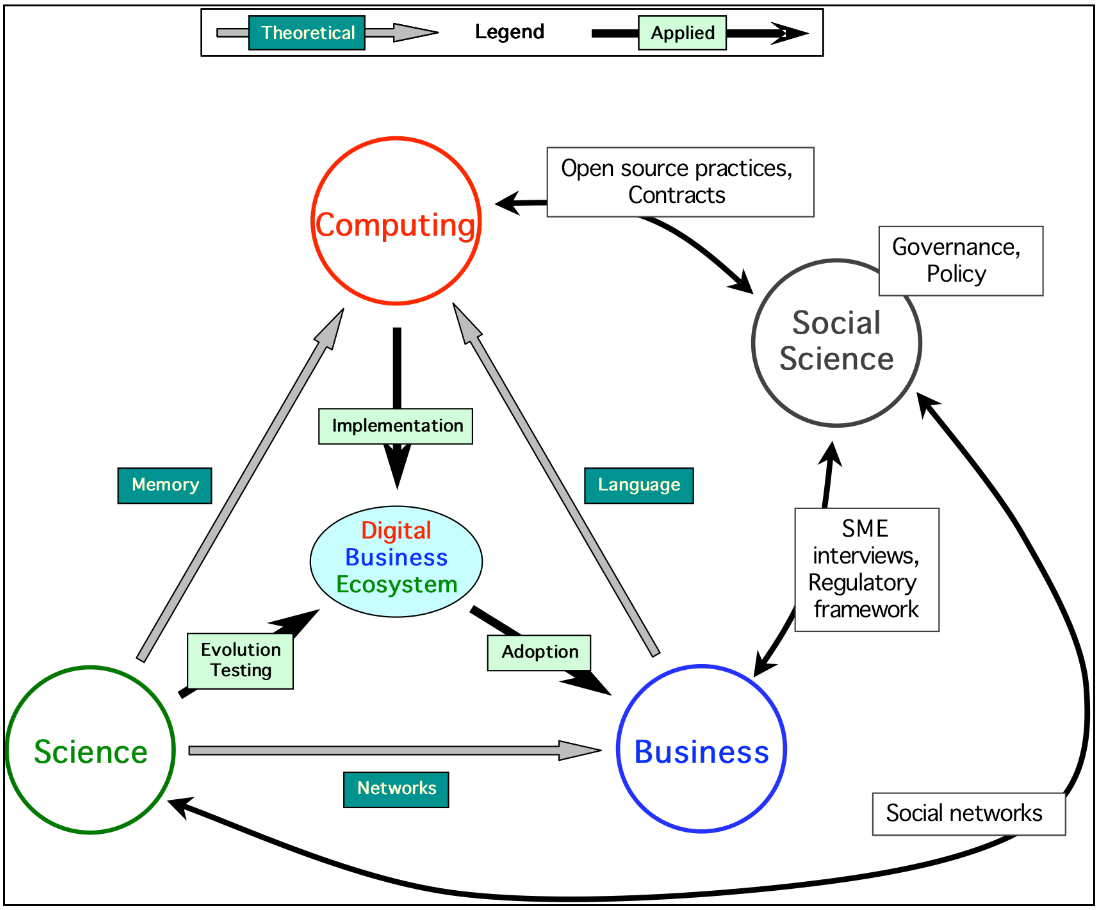

Fig. 1 Adding Social Science to the original Knowledge Flows triangle

Integration into the DBE Computing infrastructure

Fig. 2 shows a bird's eye view of which of the Science activities led to outputs that were integrated into the DBE Computing infrastructure, and give a qualitative measure of the degree of such integration. The figure is fairly self-explanatory and depicts an outcome that is quite positive, since about half of the Science outputs were integrated to different levels. Weak integration indicates also influence during development, such as the influence that research on network topology (UBHAM, HWU) had on the work of SUN to implement the ServENT P2P network.

Gene expression (LSE) stands out as the least integrated of all the Science activities. This has partly been caused by a shift in outlook on the most appropriate interface between biology and computing. In the DBE project proposal it was postulated that a fruitful collaboration could be established between LSE and SOLUTA on developing a UML profile that would incorporate biological concepts and models, to facilitate the automatic generation of code. This objective may still be obtainable, but during the course of the project it became increasingly clear that a theoretical connection between biology and computer science had to be found first. Some progress has been made in this direction, but the main problem of code execution through run-time interaction has not been solved yet. The objective remains eventually to provide a facility to specify software functionality at a high level, for the automatic generation of behaviour and/or code. However, it is not clear if UML, SBVR or some other language will be involved. As discussed in D18.4, before the problem of semantic specification can be addressed a common mathematical framework between biology and computer science needs to be developed. In parallel, a stronger connection between language, SBVR, software engineering, and social science will be sought. The integration of these two threads is expected to occur toward the end of the OPAALS project (see D18.6).

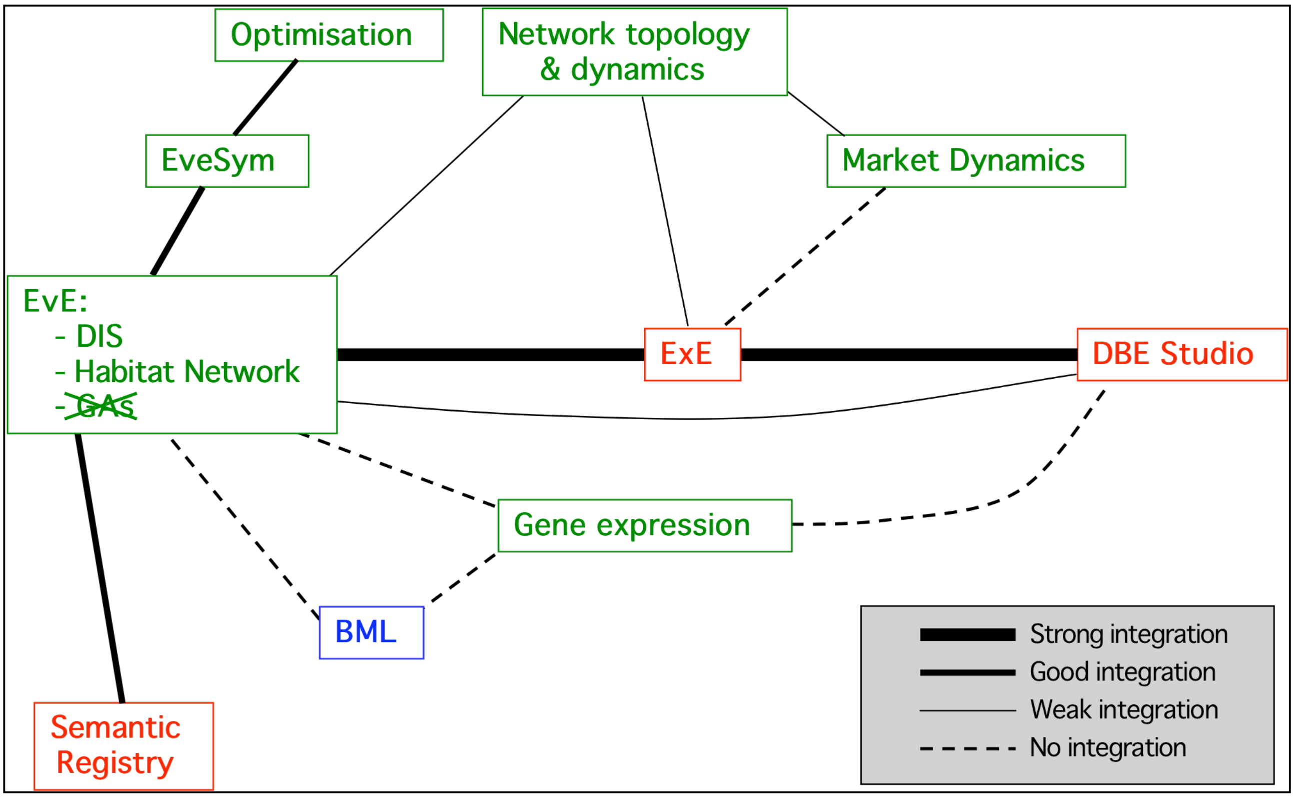

Fig. 2 Integration of Science outputs into DBE Computing infrastructure

Summary of achievements by individual partners

ICL

* Developed a versatile software package to produce models of a DBE that share the same statistical properties as a real DBE. Such a model allows one to produce realistic data with which to test various algorithms in relation to a DBE. The input parameters to such a simulator are observed statistics of a real DBE.(D12.3 and in final form D12.4)
* Proposed a way to model and express mathematically the concepts of customer preference and trust using psychophysical laws that model human feelings in response to external stimuli. (D12.3)
* Proposed an algorithm for product chain creation that is based on dynamic programming and performs better than a greedy algorithm, in the sense that it creates batter quality chains.(D12.4)
* Proposed methodology to model the interactions between companies based on buyer-seller relations.(D12.2, D12.3)
* Developed methodology for constrained satisfaction of the needs of individuals in a closed system of finite resources, with application to computer science, for example. (This was not included in any deliverable because it is based on D12.1 which the assessors rejected.)
* Developed software to predict the fate of the companies in a DBE in a static world. (D12.3)
* Developed software to predict the evolution of a DBE when the world is dynamic either by new companies entering the DBE while some go bankrupt, or when companies change their product portfolios.(D12.5)

UBHAM

UBHAM have made considerable progress in the study of evolutionary algorithms and their underlying principles. The main focus has been on the use of Markov Chain theory to establish connections between the behaviour of finite populations, and the corresponding theoretical infinite population equations. For example, the classical Geiringer's Theorem describes the action of crossover in the infinite population limit, and for fixed length chromosomes. We have proved the corresponding theorems for finite populations and for variable sized structures. We have also undertaken a review of current theory of genetic algorithm population dynamics and extended it, again making considerable use of Markov Chain theory.

The application of evolutionary processes to the DBE can be abstracted in two ways. Firstly, the solutions provided by the system might be highly-structured and variable in size and shape. The extension of genetic algorithm theory to such cases is extremely difficult, although some progress has been made along these lines. Secondly, at a higher-level, the DBE system may be thought of as supplying collections of (possibly overlapping) partial solutions, and the problem is to assemble to most cost-effective combination of such solutions which meet the user's requirements. In this way, the problem may be described as a dynamic set cover problem. Empirical studies and theoretical considerations have led us to conclude that an evolutionary approach is indeed superior to other optimisation strategies on such problems.

In addition to pursuing theoretical research, we have also worked closely with the other Science partners (notably STU and HWU) in providing assistance and advice in constructing simulations of the DBE system. In particular, we have provided STU with direct practical assistance in constructing the architecture of the DBE evolutionary environment.

The second major area of research for UBHAM is in the study of network dynamics and the application of such theory to the organisation of the DBE considered as both a social and physical network. The primary contribution here was the solution to the problem of how long a message would take to be copied across such a network, assuming transmissions are noisy. The solution is general and can be applied to arbitrary network architectures. It can be concluded that the "scale free" architecture which is likely to evolve as users join and leave the DBE is in fact fairly efficient in this regard. However, it may be that many users become isolated over time, and so we considered a self-adaptive algorithm in which users can change their connections (e.g. guided by a wish to be associated with users with similar profile). Preliminary studies show that such a system can achieve an efficient configuration in reasonable time. We have been in communication with SUN on this issue, but unfortunately time did not permit a direct collaboration.

A third area of research encompasses both the evolutionary systems and the network dynamics work. This is the problem of how to reduce a dynamical system with many degrees of freedom into one with far fewer, in such a way that the dynamics are preserved. We have found necessary and sufficient conditions for this to be done. In the simplest case, we consider a system with linear dynamics (e.g. a Markov chain) and the aggregation of states by equivalence classes. This work was extended to consider non-linear differentiable dynamical systems and arbitrary non-linear changes of variables. The resulting theorems can be applied directly, for example, to considering what kinds of coarse-grained models exist for particular evolutionary (or network) dynamics.

Publications

Mitavskiy, B. and Rowe, J.E. (2005). A schema-based version of Geiringer's Theorem for nonlinear Genetic Programming with homologous crossover. *Foundations of Genetic Algorithms*, Vol. 8. LNCS 3469. pages 156-175. Springer-Verlag.

Rowe, J.E. (2005). Population dynamics of genetic algorithms. In *Foundations of Learning Classifier Systems*, Bull, L. and Kovacs, T. (eds). Springer.

Rowe, J.E., Vose, M.D. and Wright, A.H. (2005). State aggregation and population dynamics in linear systems. *Artificial Life*, Vol 11, no. 4, pages 473-492.

Mitavskiy, B. and Rowe, J.E. (2006). An extension of Geiringer's Theorem for a wide class of evolutionary search algorithms. *Evolutionary Computation*, Vol 14, no. 1, pages 87-118.

Mitavskiy, B. and Rowe, J.E. (2006). Some results about the Markov chains associated to GPs and general EAs. *Theoretical Computer Science*, Vol 361, pages 72-110.

Mitavskiy, B., Rowe, J.E., Wright, A.H. and Schmitt, L.M. (2006) Exploiting Quotients of Markov Chains to Derive Properties of the Stationary Distribution of the Markov Chain Associated to an Evolutionary Algorithm. In *Simulated Evolution and Learning*. Lecture Notes in Computer Science, Vol. 4247, pages 726-733. Springer.

Rowe, J.E., Vose, M.D. and Wright, A.H. (2006). Differentiable coarse graining. *Theoretical Computer Science,* Vol. 361, pages 111-129.

HWU

D6.4 summarised 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 was preliminary, and therefore, the character of the report was partly speculative. This was necessary, provided early at month 21, to ensure the milestone at month 24 of a check that the outline design is compatible with the research conducted in the science stream.

D6.6 assimilates the distributed intelligence research into a suitable format for the engineering of the Distributed Intelligence System (DIS). As the DIS is augmentative to the EvE, the DIS architecture requirements are supplementary to the EvE architecture requirements, as defined in the 'Evolutionary Environment Architecture Requirements' internal project document. So, this report also includes the EvE architecture requirements, which recently have been updated to include some modifications and improvements from our extended simulations and the current implementation of the EvE. This report will supplement our existing informal communication of the DIS architecture requirements to INTEL for its implementation, ensuring that the distributed intelligence capabilities, studied by us at Heriot-Watt University, are integrated into the architecture of the DIS. This provided the milestone of a high-level design specification of the DIS, in a format suitable for its implementation, as an augmentative system to the EvE. Also, the objective of an EvE high-level design that would be integrated optimally and faithfully into the DBE Core Architecture, upon which the DIS is dependent.

D6.5 was a paper on our simulations of the effect of distributed intelligence in evolutionary dynamics, specifically the effect of the DIS on the EvE. The evolutionary dynamics was seen to be enhanced significantly with the application of the distributed intelligence, and marginally more so when using Support Vector Machines (SVM) than NNs. This provided our conclusions to the milestone regarding a 'change in the evolutionary dynamics with distributed intelligence applied'.

D6.7 outlines the task for the implementation of the DIS. The DIS provides the implementation of distributed intelligence for the EvE and Habitat network within the DBE project. From the 'High-Level Design Specification of the Distributed Intelligence System' deliverable, D6.6, it was derived that the DIS is required to perform two major tasks, service matching and targeted migration. The design and implementation of the DIS are outlined in this document. This met the milestone of an implementation of the DIS compatible with the EvE.

- During a few meetings with Claudius Masuch, the DIS components were integrated with the Habitat service and other core EvE components. This integration required some minor reworking on both the EvE and DIS components.
- The deliverable, D6.7, was written, and following an internal review, was submitted in mid-December 06.
- A DIS test simulator was implemented to enable the testing of the DIS components. This simulator is a Java application that uses a set of Service Manifest files to demonstrate NN matching and targeted migration.
- Some work was done for the preparation of an integrated EvE demo. This included providing advice with the development of an EvE query tool. Also it was investigated to see if the EvE components could be included within the main computing demo for the forthcoming review.

All of our activities, deliverables and millstones have worked to achieve the objective of the 'research, design and implementation of a DIS based on intelligence based on self-organisation to optimise and enhance the EvE.

The architecture of the DIS is very different to how it was seen at the start of the project. It was originally seen as a subsystem integrated with the Distributed Storage System (DSS), but separate to what is now called the EvE, providing fast responses to user requests. It was expected that the DIS would be more effective than the EvE in the short term, but less effective in the long term. The DIS developed differently for two main reasons. Firstly, INTEL made quick progress with the DSS before the start of the DIS research by HWU (whose researchers were originally based at Imperial College London (ICL) at the beginning of the project). Secondly, the direction of the EvE research, specifically the creation of the EvE model, showed that the 'ecosystem' concept would be fundamental in meeting the very significant and diverse goals to which the project aspired.

Also, from our research-orientated point of view the EvE is conceptually fundamental, and without it there is no Digital Ecosystem, or DBE. Deliverable 'D2.3 Software Roadmap'[1] states this succinctly, 'The concept (and the following implementation) of the Evolutionary Environment, is what makes the difference between the DBE project and any other project'. Although the architecture of the DIS has radically changed from its original inception, it will still help to provide better solutions to users in the shorter term than the EvE could alone. The targeted-migration approach chosen for the DIS will directly achieve this by targeting migration based on Migration and Usage Histories (MUHs), in response to similarity between EveServices determined by comparing their descriptions.

STU

In task S12 – Mathematical Model of Fitness Landscape – the objective of achieving distributed fitness for sharing of services was performed by suggesting a model of distributed optimisation, which is an extension of the fitness landscape simulator outlined in DBE deliverable D9.1. The resulting deliverable D9.2 – Report on evolutionary and distributed fitness Environment – discusses the abstract encoding of SBVR as well as first simulation results on "Critical Mass Assessment", "Clustering", and "Usage-based clustering" to better estimate metrics for the dynamic behaviour of the DBE system. In this task, communication and cooperation took mainly place with the partners Censis, ICL, ISUFI, LSE, and UBham.

A **collaborative tool for cross-domain interaction** was developed in context of task C37 – Simulator Evolutionary Environment – with the goal of applying the fitness calculus in a distributed evolutionary configuration of typical business environments to be able – amongst other aspects – to explore fitness options for SMEs. This tool facilitates investigation of boot-strapping and the systematic comparison of methods and mathematical approaches (e.g. investigation of long-term stability). Deliverable D9.5 - EvE Simulator Implementation – describes the three main tasks of the EvE Simulator framework which are (i) simulation of real-world networks and randomly generated large-scale networks, (ii) implementation of test bed for algorithm tuning, and (iii) configuration and visualization capabilities for non-technically experienced people. Stakeholders herein are researchers from natural as well as social sciences, business analysts, new regions joining the DBE, and the general public for dissemination of DBE concepts and 'philosophies'.

Moreover, D9.5 details two critical simulation cases: (i) critical mass effect, (ii) clustering, and (iii) usage-based clustering. Different optimization algorithms are implemented (GA – STU's own implementation to avoid the exposure to code contamination, UMDA, and GA-Beasley). An interface for import of data from the business and socio-economic partners is also implemented (CSV-import for regional networks) as well as an import/export interface in XML (together with partner ICL).

A valuable extension of the capabilities of the EvESim has been worked out together with partner CENSIS in the definition and implementation of so-called *social variables* that are most relevant regarding metrics of this distributed simulation environment (technical realisation in form of *service factories*). In addition,

STU's contribution to CENSIS's deliverable D27.3 – Social variables for simulations – are based on this work and cooperation.

Finally, the EvESimulator was published as a SourceForge project and is available as a single-click installation package (Java Web Start technology) from http://evesim.sourceforge.net.

The first implementation of one of the **core conceptual engines** in DBE, the EvE Service implementation, was performed via task C38 – EvE Service Implementation – and is documented in deliverables D9.3 – EvE Service Implementation Technical Documentation and D9.4 – EvE Service Implementation. The EvE Service Implementation provides node lookup, service pool exchange, and integration of the network of habitats based on the DBE Servent architecture as well as integration of habitat services into the Servent. This work was performed in collaboration with partners Intel and Techideas. In synchronisation with the activities related to the EvE Simulator, a SourceForge project for the EvE Network Service was initiated that can be accessed via http://evenet.sourceforge.net.

In task S24 – Adaptive Service Generator and Gene Expression, the possibilities of automatic service adaptation and gene expression were evaluated. Enhancements of the concept of Service Manifest (SM) for representation of services via service DNAs were developed to improve applicability of evolutionary and adaptive computing. The resulting deliverable D16.3 – Report on Adaptive Service Generator – gives a first abstract definition of SBVR model comparison and explains the concept of generation of UML class diagrams from SBVR models. A model of process-computing (PI-calculus) is compared with the function-oriented classical Lambda-calculus (cooperation with partner LSE); the bridge between both of them in the context of SOA is detailed. Moreover, the issues of *set covering* (including variants of genetic algorithms for solving the SCP; cooperation with partner UBham) and *automated theorem proving* (including an initial first proof-of-concept in OCaml) are explained in the context of the still open SBVR challenges.

LSE

A good conceptual understanding of self-organisation was reached through Deliverable D18.1, "Report on DBE-Specific Use Cases". Better clarity on the interaction of biology with computer science and social science with software engineering was achieved.

Understanding of the theoretical and mathematical nature of self-organisation from a dynamical systems point of view was not reached to a satisfactory level because it was deemed more important/urgent to transpose the problem of self-organisation from continuous dynamical systems to discrete systems and in particular finite state machines. This work led to the identification of interactive computing as an approch potentially capable of representing in software the non-linear coupling that is observed in complex natural and physical systems. Process algebras/concurrency theory (such as Pi-Calculus) and chemical programming/multiset rewriting frameworks were identified as a promising starting point for such an approach to computing. Such an approach cannot be said to be computing a mathematical function, and yet it appears that it can lead to useful computation. The theoretical basis of this approach, therefore, suggests that the Turing Machine is not as universal a model of computation as the general accepted wisdom has come to accept. The Fraglets have been identified as an interesting starting point for exploring interactive computation ideas, but much work remains to be done. The study of the applicability of discrete groups was started with the intent of identifying symmetries (for instance in the transformation of Fraglets sets) that could act as constraints on the available state space. Finally, network coding was examined since it builds on the structure of discrete groups to develop error detection and correction codes based on finite Galois Fields. The motivation is to regard incomplete specifications, as can be expected in interactive computing scenarios characterised by a distributed algorithm, as codes with errors which can be "corrected" to fulfil a complete "phenotype" or run-time instance. The project ended before these hypotheses could be tested so this work is continuing in OPAALS.

A critical analysis of current debates on sustainable economic development and ICTs from different social science viewpoints, especially with regard to the viability of F/OS for SMEs was performed and reported in

deliverables D18.3 "Report on the socio-economics of Free/Open Source" and D18.7 "Final synthesis of Social Science contributions".

A conceptual model for the mechanism through which DNA codes for desired metabolic functions in different environments was not achieved. Overloading of the same set of genes to respond in different ways to different stimuli may require a mapping from path dependence to different algorithms. A very small initial step was made in identifying the discrete relative orientations at which enzymes interact with their substrates with the rotational symmetries of the regular solids. Because such symmetries form a finite group, and because they can all be expressed in terms of a binary alphabet, DFAs were constructed for 3 of the 5 regular solids, suggesting a way to map between spatial features and behaviour of biological molecules and a binary data stream. This is only a highly speculative conceptual link for the moment. This work is ongoing in OPAALS in collaboration with a team of molecular biologists.

The formal connections that exist between Pi-Calculus, Multiset rewriting, BPEL and SBVR started to be explored through a recent BPEL book that discusses explicitly the connection between BPEL and Pi-Calculus.[1] Discussions are underway in OPAALS with the partner responsible for the Distributed Transaction Model for P2P networks, with a view to integrate the Temporal Logic of Actions approach used for the transaction model with SBVR, using BPEL as a common syntactical underlying framework. TLA appears to be a good candidate for connecting logics to dynamic behaviour in reactive and (hopefully) emergent systems.[2]

Assessment of stated Science objectives

The table below contains a list of objectives that was agreed to by the Science partners during the second and third year of the project, as part of the Science Vision deliverable (D4.1). In this report we give a frank assessment of what we were able to achieve and what we were not able to achieve.

We should make the point that the structural context of the research has played a very important role, but that this role cannot be formalised using current reporting norms. As discussed in the Activity Report, the main impediments to achieving these objectives were:

1- the challenges of reaching a consensual view of the research objectives, even within the Science Domain
2- the "bootstrap" of a radically interdisciplinary research group within a very traditional social science institution (LSE), which has made it difficult to recruit physicists, biologists, and mathematicians to work on some of the WP18 tasks

[1] Havey, M (2005). *Essential Business Process Modelling*, O'Reilly.
[2] Kurki-Suonio, R (2005). *A practical Theory of Reactive Systems: Incremental Modelling of Dynamic Behaviour*, Springer.

In the table below "(OPAALS)" means that a particular objective is being pursued in the OPAALS project.

Objective	Description	Assessment
	Languages, DNA, automatic code generation	
1	To investigate the algorithmic properties of the simplest metabolic cycles of the DNA and to relate its interactions with other cell components to finite automata and formal languages, with reference in particular to the π-Calculus and to Fraglets	**Not reached, work only begun (OPAALS)**
2	To assess to what extent the framework developed in 1 is applicable to UML and BML (SBVR) and to offer suggestions for increased applicability	**Not reached (OPAALS)**
3	To investigate the connections between dynamical systems and formal languages, using the π-Calculus as a starting point	**Not reached, work only begun (OPAALS)**
4	To begin the development a theoretical framework based on 1-3 that addresses automatic code generation or "behaviour generation" from models	**Not reached (OPAALS)**
	Evolution and intelligence	
5	To reconcile the model-driven (ontogenic) and the evolutionary (phylogenic) views of the DBE through appropriate architecture recommendations to the Computing group	**Not reached (OPAALS)**
6	The define a high-level architecture for the Evolutionay Environment	**Reached**
7	To investigate and develop criteria for semantic matching in order to support the formation of service chains in the Population Object	**Not reached, Work begun (OPAALS)**
8	To develop a distributed intelligent system to accelerate the evolutionary optimisation algorithms	**Reached**
9	To develop a general and adaptive Fitness Function tailored to the DBE	**Half-reached (OPAALS)**
10	To investigate the properties of genetic algorithms and adapt them to DBE needs	**Work begun (OPAALS)**
11	To achieve long-term learning and adaptive behaviour of the DBE	**Partially reached in principle, not tested (OPAALS)**
	Networks	
12	To develop criteria to optimise information propagation in networks of heterogeneous dynamic topology, in order to support Habitat clustering and dynamic P2P links	**Partially reached (OPAALS)**
13	To support the FADA network architecture development through simulations	**Reached**
	Business ecosystem	
14	To investigate the applicability of global cost function optimisation of multi-agent systems and of game-theoretic approaches to the analysis of binary SME transactions	**Reached**
15	To work with Social Science to develop a theoretical framework underpinning sustainable regional development based on Digital Ecosystems	**Partially reached (OPAALS)**
16	To study through simulations how sharing of and competition for services affects the efficiency of the market	**Reached**
17	To support the integration of all the EvE and business ecosystem simulations in the EvESimulator framework	**Reached**