

Enabling Virtual Organisation in Mobile Worlds

Peter Komisarczuk MIEE/MIEEE, Kris Bubendorfer, Kyle Chard

Mathematical and Computing Sciences, Victoria University of Wellington, PO Box 600, New Zealand
{Peter.Komisarczuk, Kris.Bubendorfer, Kyle.Chard} @mcs.vuw.ac.nz

Keywords: Resource management, NOMAD, market driven

Abstract

Future mobile network services could be provided around dynamically created virtual organisations (VO) utilising mobile code executing on a set of computing platforms (here called Depots). These VO's could be associated with enterprises, service providers, or mobile network operators whose code migrates amongst some set of computing platforms, some of which are closely associated with the mobile network operators infrastructure, e.g. machine-to-machine service gateways. These VO's are likely to require dynamic resource acquisition based on the location and mobility of their clients/service agents. In this paper we introduce NOMAD – Negotiated Object Migration Access Deployment, which is a software architecture that provides support for mobile, distributed applications, allowing such applications to locate and obtain resources where and when they are needed based on an economic resource management model. NOMAD enables virtual organisations to optimise their operations to gain competitive advantage, in an efficient market driven global infrastructure.

1 Introduction

It has been postulated that utility computer service providers [9] will be the basis for cost effective, easily extensible solutions for enterprises and service providers in general, including network service providers. One currently evolving model of utility computing is grid computing and this model may be considered a good basis for exploring utility computing in the context of future communication services. In fact the European Union GRASP [6] project is one such study exploring the grid paradigm based around the Application Service Provider model.

In the context of grid computing, a Virtual Organisation (VO) is created in order for a group of distributed computing resources to collaborate to achieve some task, such as a simulation [8]. In this paper we explore the utility computing model in the context of future communication services, with particular emphasis on a case study of a machine-to-machine (M2M) solution over a cellular network. Our emphasis is on enabling resource management and mobile object migration to allow VOs to exploit mobility and negotiate for network resources as well as computer resources. This paper's key

contribution is to extend the existing work of general market driven resource management.

Future mobile applications will cover both personal/user communication and machine-to-machine communications scenarios; although machine-to-machine communications is likely to be the dominant wireless network traffic and consumer of application services [4, 13]. Typically mobile devices are limited in their capabilities and require network based software entities to provide value added services or to collate, process and distribute data for user display or machine applications. The network based software entities are likely to utilise mobility and dynamically adapt to changes in their environment and to support heterogeneous computing and networking.

We can postulate that future computing platforms supporting wireless networks will need to provide support for the creation of virtual organisations (VO). A VO consists of one or more applications consisting of mobile code. The VO provides resources/services to their deployed customer base, whether user or machine, mobile or stationary. The VO can be associated with a single user but is most likely to be associated with a larger entity, e.g. a transport company using an M2M solution, or alternatively some Service Provider enabling the M2M infrastructure and applications for the transport company.

Within the NOMAD infrastructure VO's will enable the development of services and application layer network components that will incorporate the myriad of future user and machine interactions. These VO's are likely to be based on high performance computing clusters (or grid style of computing resource) where mobile agents and mobile software will cooperate in order to provide the services required for the different users/organisations. The service infrastructure is based on provision of computer functionality at the optimal location for the delivery of the services on the communication networks. The optimal location of a service (a piece of mobile software) may be close to the mobile device in order to minimise latency and so optimise response time, throughput and network costs.

Related Work

The virtual organisation and market driven component considered in this paper could be thought of as similar to the architecture that has been proposed by the GRASP project [6]

an EU IST (Information Society Technologies) project. GRASP is based on the ASP (Application Service Provider) model integrated with the Virtual Organisation (VO) model from grid computing. In the GRASP model services are made available by ASPs, which create service descriptions that are offered through a service locator [7]. However in the model presented here, the mechanisms of negotiating and providing resources differ from the GRASP model. However although the basis is similar – effectively enabling the outsourcing of resources. Our emphasis is on enabling market driven computer and network specific resource management combined with mobile object migration to enable VOs that can exploit mobility and use a wider variety of resources, such as mobile gateways. In order to provide this functionality appropriate middleware is required with the capability to describe, negotiate/contract and deliver such capabilities.

A prototype system that provides a distributed infrastructure to negotiate and optimise resource allocation, create resource contracts and exploit mobile objects has been developed at Victoria University of Wellington. The system is called NOMAD - Negotiated Object Migration Access and Deployment [2]. NOMAD identifies the requirements of the user, application or service, using a Resource Distribution Graph (RDG), based on locality, required connectivity and computational resources (CPU, memory and I/O capabilities, including network resources). The RDG is used within NOMAD as the basis for requesting resources, agreeing the contract and enforcing the contract within the NOMAD infrastructure. Resource allocation is enabled through a Vickery auction based process, which is discussed later.

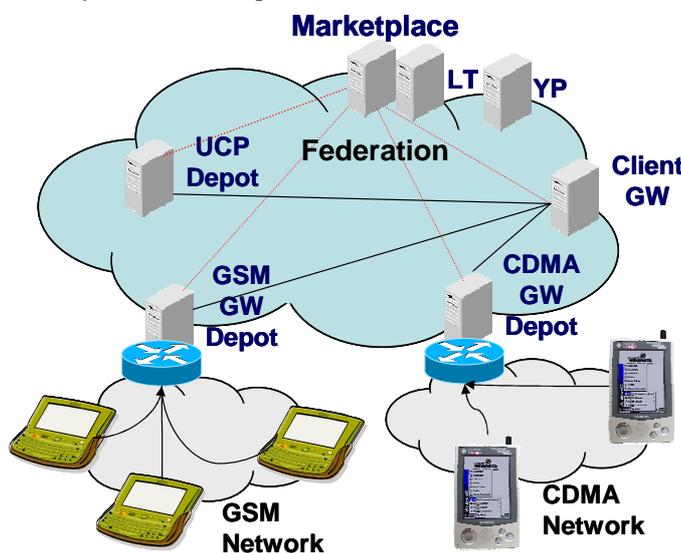


Figure 1 NOMAD federation, Utility Computing Depot, GSM and CDMA Depot and client gateway forming a VO.

The VO has been used within grid computing, but there it has a different emphasis. In grid computing the VO is associated with a set of participants that join to collaborate on a particular project. Here the VO consists of the negotiated resources that collaborate to form the entity or organisation.

The resources can be either dynamically or permanently allocated to an organisation, e.g. a gateway to the utility computing infrastructure. Grid computing has previously investigated a resource brokering architecture, which was introduced by NorduGrid [5]. Now with the introduction of the OGSA – Open Grid Services Architecture, the grid is now capable of accommodating the NOMAD resource negotiation mechanism. The grid has a resource management framework in which resources are assigned; however the overarching resource negotiation mechanism of NOMAD is potentially a valuable extension to these mechanisms.

A number of distributed systems have employed variants on market or computational economies. These have included Spawn (first full attempt at distributed computational economy, with bid escalation mechanism), Xenoservers (similar economic model to NOMAD) and D'Agents (auctions are initiated by servers for the excess capacity, access list controlled), see [2].

This paper outlines NOMAD in section 2, including a discussion of the RDG, the marketplace based on a Vickery auction mechanism. A brief design for a M2M over cellular network application is presented in section 3 and current work and future directions are discussed in section 4.

2 Introduction to NOMAD

NOMAD is a middleware architecture designed to support applications consisting of dynamic mobile objects. The high level infrastructure shortens time to prototype, reduces application complexity, and improves interoperability [2]. NOMAD consists of loosely coupled cooperating machines called Depots organized in federations (an administrative domain), where each Depot provides resources such as CPU cycles, memory, disk, such as the UCP (Utility Computing Provider of [7]) and additionally specific hardware and software components, such as the GSM gateway of Figure 1.

In NOMAD applications consisting of mobile code, negotiate for the resources provided by the network of Depots. The allocation of resources in the NOMAD infrastructure is based on an auction coordinated by a Marketplace, which consists of a number of Markets. An auction is defined as an “important stylised market mechanism, with well-defined rules, that allow selling in situations where a more conventional ‘market’ (set price) does not exist”. An auction is made of three major components: the vendor who has items to sell, the bidders who place bids for items, and the auctioneer who receives the bids and determines the winner according to the auction protocol. The auction protocol is responsible for a number of important aspects of the auction, including determining the rules of the auction, how the auction is carried out, how bidders communicate, how to determine the winner, and how much the winning bidder pays.

There are many variations of auction protocols; the one chosen for NOMAD is the Vickery auction mechanism, in which a sealed bid (i.e. a closed forum) is presented to the

auctioneer. At some predefined point in time the received bids are opened and the highest bid wins the auction. The winning bidder then pays the valuation of the second highest bid. It has been shown through the “revenue equivalence theorem” that the major auction protocols, including Vickery, yield the same expected returns in “private value” auctions (i.e. an auction in which the value to the bidder is presented irrespective of other bidders values for the item).

In the case of the NOMAD system the best price to an application (virtual organization) is the lowest price for which they can obtain the combination of resources that they require. NOMAD uses an approximation of the general Vickery auction (GVA), which leads to the correct allocation of multiple resources with minimal overhead [1,2]. Without the GVA approximation the auction solution is intractable (it is NP-complete) in auctions which consist of combinations of resources, which are the primary concern in NOMAD. The winner in NOMAD pays their bid less a discount, where the discount is calculated from the impact of a bidders participation in the auction.

In NOMAD the Depots play the role of bidders and the Market plays the role of the auctioneer. The Market is introduced as an independent auctioneer to deal with the problems of a lying or compromised auctioneer. For this reason it is implemented as a third party application operating at the NOMAD system level. The Market collects auctions from applications and distributes the auction data to suitable Depots via catalogues. The Market then collects bids from Depots, determines a winner, and finally returns a contract for the resources to the winning Depot and the initiating application.

The Vickery auction protocol is a single bid Pareto-optimal technique ideal for machine to machine negotiation [12]. From a computational perspective NOMAD provides an optimal GVA implementation by passing the description of the lot (which is a set of goods in an auction) to each Depot (or agent acting on behalf of a Depot).

The Application constructs a representation of the resources it requires in order to take part in an auction. Resource constraints are encoded in a Resource Description Graph (RDG), which describes the lot and any associated compromises that could be considered. The graph consists of edges representing resources and vertices representing accept states, which represents the task which the initiator wishes to take place. A path from the root to an accept state is called a sentence; the sentence represents the resources that the Depot is required to provide in order to fulfill the task.

The Depots (the bidders) partake in an auction to determine who will provide the resources and host the Application. The Market (the auctioneer) facilitates the auction, distributing the auction data to the Depots in the form of a catalogue. The Depot then processes the lots contained in the catalogue using policies to price their resources and returns the sealed bids to the Market. The bids received in an auction are then opened

at a predefined time, the winner is determined and finally a contract is returned to the initiating application and the winning Depot. This distributed auction mechanism avoids bottlenecks at the Market.

To complete our high level discussion of NOMAD we must mention that the Depot is composed of a Depot Manager and set of managed virtual hosts on which applications are hosted. Nomad also has support for mobility and lifecycle functions. Mobility is provided at two levels – the first is the yellow pages (YP) service which provides global service discovery and secondly through Location Tables which provide for mobile object binding. The location service is a distributed to optimize the distribution of location information within NOMAD [3]. Applications can customise location tables to meet their own functional requirements.

The Resource Description Graph

The RDG [2] is a rooted directed acyclic graph that describes and values resources in NOMAD. It can also be used to express resource allocation/usage, in the context of contract validation, or to enable dynamic resource management functionality. In NOMAD the initiator’s constraints represent resources that are required by the initiator and supplied by the bidders (the Depots). To represent a task the edges of the RDG are used to represent individual resources and the vertices indicate an accept state, for example the sequence of vertices ending in an accept state shown in Figure 2. By encoding resource constraints within the bid expression, NOMAD minimises the NP-complete Combinatorial Allocation Problem (CAP).

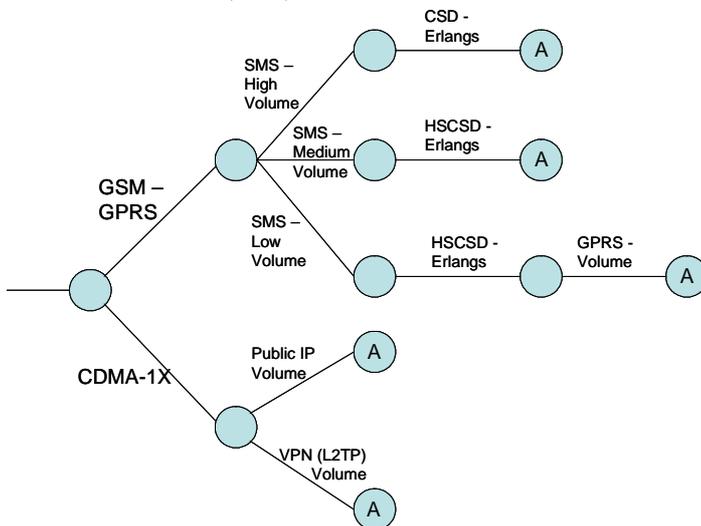


Figure 2 RDG example showing an M2M over cellular RDG (showing NZ network and bearers options to accept states)

Each sentence in the RDG has a valuation; valuations to the same accept state are normally different as they represent different sets of resources that can achieve the same outcome. Bids can only be made on accept states (which is also called a “lot”). This avoids partial bids being returned and so avoids the need for an application to combine contracts. NOMAD allows multiple accept states to be specified as a single bid,

called an “entirety bid” which can be made on the whole auction in an attempt to capture a degree of bidder synergy [2]. Multiple sentences to a single accept state allow NOMAD applications to specify alternate ways to satisfy a lot (through compromises), and target their needs for specific Depots. In the example shown the M2M gateway can choose to provide mobile client connectivity through SMS + CSD, or replacing CSD with HSCSD, or HSCSD + GPRS. The sentence with the lowest bid value will be taken by the Market.

The RDG approach has a number of advantages; firstly the acyclic graph prevents malicious graph loop attacks by applications. An RDG is created by a programmer in a VO/application to reflect policy; it is passed to the market and distributed to Depots via catalogues. It has a human readable form that can be stored, printed, and distributed safely (no code or active components reducing risk of attacks on market or Depots). The major advantage is the ease in which the graph can be broken up to represent small components of a problem. Modules can be created to quickly characterise a common problem, and may be combined with other modules (composition) to form complete graphs [2].

3 M2M over Cellular Case Study

As one example for testing the development of the RDG and NOMAD in telecommunication applications, we selected a machine-to-machine (M2M) application. The chosen M2M technology [10] utilises wireless CORBA (the Common Object Request Broker Architecture) located at a gateway on the GSM network connected to the Internet or a VPN. The gateway is called the access bridge and the generic gateway architecture is shown in Figure 3.

The Wireless CORBA specification [11] provides mechanisms to support physical (terminal) mobility in CORBA. Vanilla CORBA is not well suited to mobile communications due to the reliance on the TCP/IP protocol stack. Wireless networks add complications arising from sudden loss of connection, low bandwidth bearers and potentially high error rates which are dealt with through transport adaptation in the wireless gateway.

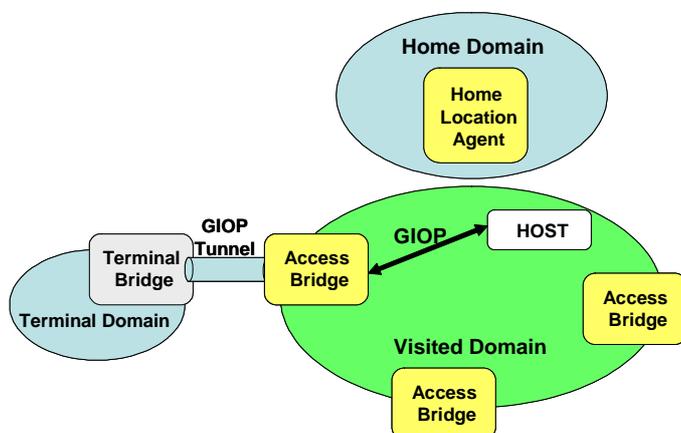


Figure 3 – M2M and wireless CORBA gateway architecture

The wireless CORBA framework defines three domains: the home domain, the visited domain, and the terminal domain and instead of specifying a mapping from GIOP (General Inter-ORB Protocol) onto the Internet Protocol, the specification defines a method to tunnel GIOP messages between the terminal and access bridges to optimise connectivity. Handover between access gateways is supported, e.g. through terminal mobility.

The M2M solution based around wireless CORBA utilises wireless networks to communicate between remote objects in the field and the corresponding application running at a service centre via an access bridge. In the Nokia solution GSM/GPRS bearers can be used dynamically according to the service policies, to perform the required client/server interactions. The bearers are SMS (Small Message Service), CSD (Circuit Switched data), HSCSD and GPRS (General Packet Radio Service). The data collected from the remote object is forwarded to the gateway before being routed through the Internet to a service centre where the corresponding server application is running.

M2M applications over cellular should allow the exploitation of geographical locality. Unlike other applications, simple M2M applications often do not need to specify stringent computational requirements as the mobile device is likely to have a small microcontroller and the upload/download of information is limited, e.g. GPS co-ordinates, oil temperature, speed etc. An example application that combines fixed and mobile assets is a simple vending machine stock application, where stock levels are uploaded to a server. The server processes the information and downloads restocking information to the closest or most appropriate stock truck

A New Zealand based case study is being implemented based around the outline in Figure 1. In this example an M2M service provider creates a virtual organisation to provide customers with access to their mobile devices through a web interface (deployed on the utility computing Depot), with devices connected to wireless CORBA gateways on the Vodaphone GSM/GPRS network. The Nokia wireless gateway product is not available for the TelecomNZ CDMA-1X network, so instead an alternative gateway is to be implemented. These cellular networks are to be extended to 3rd generation in the 2004/5 timescale, but the RDG in Figure 2 and options in Figure 4 only shows current network options.

The VO has a number of implementation options described in Figure 4. The VO can provide their own wireless gateways, in which case the VO needs to specify that it requires Depot’s with particular computing resources (CPU, memory and IO) plus choice of network, bearers and required capacity. Alternatively the VO can request a number of service provider provisioned wireless gateway with the network and bearer capabilities required to provide the M2M device connectivity. Part of the required RDG is shown in Figure 2. Currently the Nokia gateway does not support the mobility of itself, so initially only the Service Provider provisioned model is being explored.

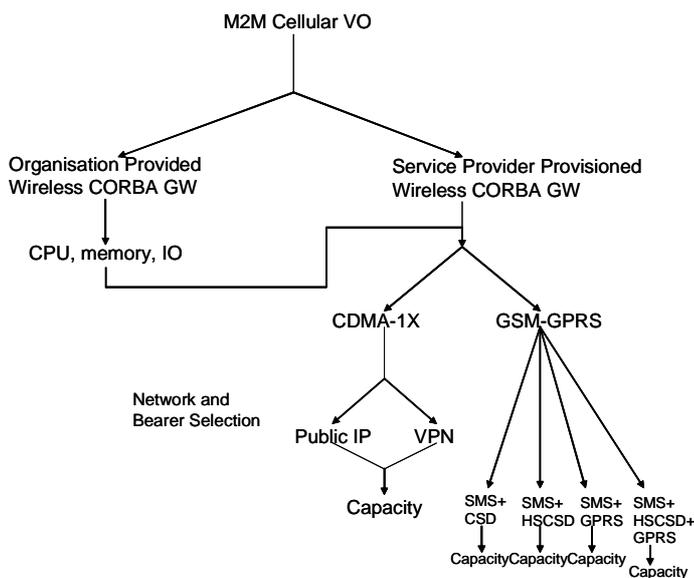


Figure 4 VO wireless CORBA solution choices

The virtual organisation can be further envisaged from Figure 5. The Service Provider has a generic framework containing gateway resources, management functions and event transmission services. These are employed to bind mobile devices to Client Applications. These channelled events are collated, processed and placed into long term storage and analysed events (e.g. correlated messages) are presented to the clients through the web interface. The web interface provides generic client access to the M2M client application.

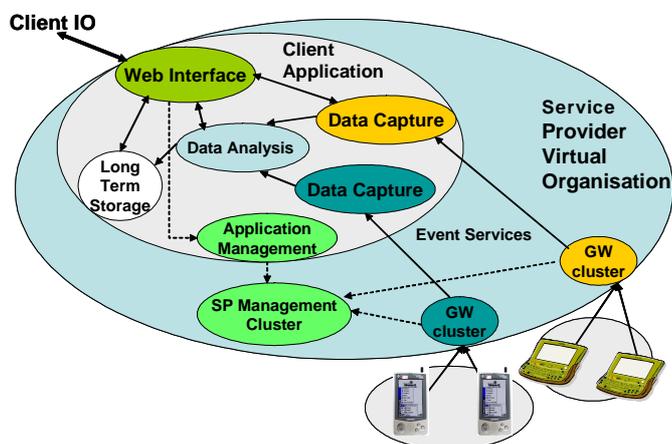


Figure 5 High level VO and application view

4 Conclusion

This paper has introduced NOMAD and its innovative resource allocation mechanism including resource description graph and provided details of a cellular service provider solution based on NOMAD. Extensions to the RDG for the description of communications environment are currently in progress, with some aspects of that work highlighted in the service example discussed in this paper. The NOMAD contract mechanism is currently being extended to use

SHARP style mechanisms and a payment infrastructure. Mobile device support has been added with the development of μ NOMAD for small memory devices, the first prototype runs in Java on the Compaq IPAQ and currently integration with the grid is being investigated.

References

- [1] S. Antlif, "Experimental verification of collective vickery auctions in NOMAD", Victoria University of Wellington, 2003. <http://www.mcs.vuw.ac.nz/comp/graduates/archives/honours/2003/antliff-simon/antliff-report.pdf> last access 24 July 2004.
- [2] K. Bubendorfer, "Nomad: Towards an Architecture for Mobility in Large Scale Distributed Systems", PhD Thesis, Victoria University of Wellington, December 2001, <http://www.mcs.vuw.ac.nz/research/dsrg/papers/KBthesis.pdf> last access 24 July 2004.
- [3] K. Bubendorfer, J. Hine, "NOMAD: Application Participation in a Global Location Service", *Lecture Notes in Computer Science*, **number 2574 in LNCS**, pages 294-306, Springer Verlag, January 2003. Proceedings of the 4th International Conference on Mobile Data Management, 21-24 January 2003, Melbourne, Australia.
- [4] Deloitte Research, "Mobilising the Machine", 2002.
- [5] P Eerola, B Konya, O Smirnova, T Ekelof, M Ellert, J R Hansen, J L Neilsen, A Waananen, A Konstantantinov, F Ould-Saada, "Building a production grid in Scandinavia", *IEEE Internet Computing*, **volume 7(4)** pages 27-35, July/August 2003.
- [6] Grid-based Application Service Provision (GRASP) <http://eu-grasp.net/english/> last access 24 July 2004
- [7] T. Dimitrakos, et al. "An Emerging Architecture Enabling Grid Based Application Service Provision", 7th IEEE International Enterprise Distributed Object Computing Conference EDOC 2003, Brisbane Australia, September 16-19, 2003.
- [8] I. Foster, C. Kesselman, S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", *International Journal of Supercomputer Applications*, **volume 15(3)**, 2001.
- [9] V. Machiraju, J. Rolia, A. van Moorsel, "Quality of Business Driven Service Composition and Utility Computing", Software Technology Laboratory, HP Laboratories Palo Alto, HPL-2002-66, March 15th, 2002
- [10] Nokia, "M2M white paper", version 6, 2002, <http://www.forum.nokia.com/main/1.6566.21.00.html>, last checked 2/4/04.
- [11] Object Management Group, "Wireless Access and Terminal Mobility in CORBA Specification", June 2001. Document dtc/01-06-02, updated to version 1.1, dtc/04-04-02.
- [12] E Rasmusen, "Games and Information – An Introduction to Game Theory", Blackwell Publishers, Oxford, 2nd edition, 1994.
- [13] V. Salomaa, "Opportunities in Wireless M2M Telematics", Institute of Strategy and International Business, Helsinki University of Technology, 29/3/2003.